

UNIVERSITATEA DE STAT DIN TIRASPOL

Cu titlu de manuscris

C.Z.U: 37.016:004(043.3)

GASNAȘ ALA

**METODOLOGIA IMPLEMENTĂRII SISTEMELOR DE
MANAGEMENT AL ÎNVĂȚĂRII ÎN PROCESUL DE STUDIU AL
PROGRAMĂRII ORIENTATE PE OBIECTE**

532.02 – DIDACTICA INFORMATICII

Teză de doctor în științe pedagogice

Conducători științifici:

**Braicov Andrei, doctor în
științe fizico-matematice,
conferențiar universitar**

**Cioban Mitrofan, doctor habilitat
în științe fizico-matematice,
profesor universitar, academician**

Autor:

CHIȘINĂU, 2018

© Gasnaş Ala, 2018

CUPRINS

ADNOTARE	5
LISTA ABREVIERILOR	8
INTRODUCERE	9
1. REPERE TEORETICE ALE SISTEMELOR DE MANAGEMENT AL ÎNVĂȚĂRII ÎN PROCESUL DE STUDIU AL PROGRAMĂRII ORIENTATE PE OBIECTE	17
1.1. Analiza comparativă a sistemelor de management al învățării	17
1.2. Studiu comparativ al caracteristicilor de bază ale limbajelor de programare orientate pe obiecte.....	27
1.3. Tendințe de aplicare a SMÎ în predarea-învățarea-evaluarea informaticii.....	39
1.4. Concluzii la capitolul 1	46
2. BAZELE METODOLOGICE DE FORMARE A COMPETENȚEI DE PROGRAMARE ORIENTATĂ PE OBIECTE PRIN UTILIZAREA SISTEMELOR DE MANAGEMENT AL ÎNVĂȚĂRII	48
2.1. Elaborarea modelului pedagogic de predare-învățare-evaluare a cursului Programarea Orientată pe Obiecte prin utilizarea SMÎ.....	48
2.2. Metodologia utilizării modelului elaborat.....	62
2.3. Concluzii la capitolul 2	105
3. DEMERSURI EXPERIMENTALE ÎN IMPLEMENTAREA SISTEMELOR DE MANAGEMENT AL ÎNVĂȚĂRII ÎN PROCESUL DE STUDIU AL PROGRAMĂRII ORIENTATE PE OBIECTE	107
3.1. Activități preexperimentale de utilizare ale modelului POO SMÎ.....	108
3.2. Valorificarea experimentală a modelului POO SMÎ.....	118
3.3. Validarea rezultatelor experimentale	122
3.4. Concluzii la capitolul 3	138
CONCLUZII GENERALE ȘI RECOMANDĂRI	138
BIBLIOGRAFIE	141
ANEXE.....	156
Anexa 1. Chestionar	156
Anexa 2. Test de evaluare inițială	159
Anexa 3. Teste de evaluare sumativă	162
Anexa 4. Teste de evaluare finală Teste de evaluare finală	172
Anexa 5. Teme pentru cercetare suplimentară	175
Anexa 6. Indicatorii statistici calculați cu aplicația SPSS pentru eșantionul de control, anul de studii 2015-2016, secția cu frecvență la zi	176
Anexa 7. Indicatorii statistici calculați cu aplicația SPSS pentru eșantionul de control, anul de studii 2016-2017, secția cu frecvență la zi	177

Anexa 8. Indicatorii statistici calculați cu aplicația SPSS pentru eșantionul de control, anul de studii 2016-2017, secția cu frecvență redusă.....	178
Anexa 9. Indicatorii statistici calculați cu aplicația SPSS pentru eșantionul experimental, anul de studii 2015-2016, secția cu frecvență la zi	179
Anexa 10. Indicatorii statistici calculați cu aplicația SPSS pentru eșantionul experimental, anul de studii 2016-2017, secția cu frecvență la zi	180
Anexa 11. Indicatorii statistici calculați cu aplicația SPSS pentru eșantionul experimental, anul de studii 2016-2017, secția cu frecvență redusă.....	181
Anexa 12. Histogramele eșantionului experimental 2016-2017, grupele cu frecvență redusă.....	182
Anexa 13. Histogramele eșantionului de control 2016-2017, grupele cu frecvență redusă.....	183
Anexa 14. Rezultatele experimentului pedagogic, anul de studii 2015-2016, secția cu frecvență la zi	184
Anexa 15. Rezultatele experimentului pedagogic, anul de studii 2016-2017, secția cu frecvență la zi	185
Anexa 16. Rezultatele experimentului pedagogic, anul de studii 2016-2017, secția cu frecvență redusă.....	186
DECLARAȚIA PRIVIND ASUMAREA RĂSPUNDERII	187
CURRICULUM VITAE	188

ADNOTARE

Gasnaș Ala. **Metodologia implementării Sistemelor de Management al Învățării în procesul de studiu al Programării Orientate pe Obiecte.** Teză de doctor în științe pedagogice. Chișinău, 2017.

Structura tezei: introducere, trei capitole, concluzii generale și recomandări, bibliografie din 176 titluri, 16 anexe, 140 pagini de text de bază, 33 figuri, 34 tabele. Rezultatele obținute sunt publicate în 10 lucrări științifice.

Cuvinte-cheie: proces didactic, programare orientată pe obiecte (POO), model pedagogic, Sistem de Management al Învățării (SMÎ), tehnologii informaționale, strategii interactive, instruire centrată pe student, competențe POO.

Domeniul de studiu: Științe pedagogice. Didactica informaticii.

Scopul cercetării: fundamentarea teoretică și elaborarea unui model pedagogic al procesului de studiu al Programării Orientate pe Obiecte prin intermediul Sistemelor de Management al Învățării.

Obiectivele cercetării: (1) Determinarea avantajelor oferite de Sistemele de Management al Învățării și argumentarea necesității implementării lor în procesul didactic universitar. (2) Precizarea Sistemelor de Management al Învățării cu caracteristici optime pentru sistemul educațional universitar. (3) Identificarea limbajului orientat pe obiecte adecvat studierii conceptului Programării Orientate pe Obiecte. (4) Elaborarea modelului pedagogic centrat pe utilizarea sistemelor de management al învățării în procesul de studiu al POO. (5) Argumentarea științifico-metodologică a eficienței utilizării modelului pedagogic în studierea *Programării Orientate pe Obiecte*. (6) Validarea prin experiment didactic a eficienței modelului pedagogic elaborat.

Noutatea și originalitatea științifică a lucrării constă în: a) identificarea instrumentelor SMÎ pentru predarea-învățarea-evaluarea *Informaticii*; b) elaborarea modelului pedagogic de studiu al *Programării Orientate pe Obiecte* prin intermediul SMÎ; c) elaborarea metodologiei de implementare al acestui model; d) proiectarea curriculumului la cursul *Programarea Orientată pe Obiecte*, adaptată la specificul utilizării SMÎ.

Problema științifică soluționată constă în fundamentarea teoretico-praxiologică a eficientizării procesului de studiere a *Programării Orientate pe Obiecte*, prin elaborarea unui model pedagogic de predare-învățare-evaluare în baza cursului universitar *Programarea Orientată pe Obiecte*, axat pe utilizarea Sistemelor de Management al Învățării și orientat spre formarea competenței de Programare Orientată pe Obiecte la viitorii specialiști în informatică (inclusiv în domeniul *Științe ale Educației*), având ca efect optimizarea procesului de formare a competenței *Programarea Orientată pe Obiecte*.

Valoarea aplicativă a lucrării rezultă din metodologia elaborată și posibilitatea implementării ei în cadrul cursului universitar POO prin aplicarea SMÎ, în scopul formării competenței POO la specialiștii în informatică și în domeniul ȘE.

Implementarea rezultatelor științifice: metodologia elaborată este utilizată în predarea cursului universitar POO în cadrul Facultății *Fizică, Matematică și Tehnologii Informaționale* a Universității de Stat din Tiraspol.

АННОТАЦИЯ

Гаснаш Ала. **Методология внедрения Систем Управления Обучением в процессе изучения Объектно-Ориентированного Программирования.** Диссертация доктора педагогических наук. Кишинэу, 2017.

Структура диссертации: введение, три главы, выводы, библиография из 176 наименований, 16 приложений, 140 страниц основного текста, 33 рисунка, 34 таблицы. Результаты исследования опубликованы в 10 научных работ.

Ключевые слова: образовательный процесс, объектно-ориентированное программирование (ООП), педагогическая модель, системы управления обучением (СУО), информационные технологии, интерактивные стратегии, навыки ООП.

Область исследования: Педагогика. Дидактика информатики.

Цель исследования: Теоретическое обоснование и разработка педагогической модели процесса обучения объектно-ориентированного программирования с помощью систем управления обучением.

Задачи исследования: 1) Определить преимущества СУО и обосновать необходимость их внедрения в высшей образовательной практике; 2) Выявить СУО с наилучшими характеристиками для последующего внедрения в систему высшего образования; 3) Определение наилучшего ОО языка для изучения объектно-ориентированной концепции. 4) Разработать педагогическую модель, основанную на внедрение СУО в процессе обучения дисциплины ООП; 5) Научно-методологически обосновать использование разработанной педагогической модели для изучения дисциплины ООП в ВУЗе; 6) Экспериментально апробировать эффективность разработанной педагогической модели в процессе педагогического эксперимента.

Научная новизна работы: а) определение инструментов СУО использованные для обучения информатики; б) разработка педагогической модели для изучения дисциплины ООП путем внедрения СУО; в) разработка методологии для реализации этой модели. г) разработка учебного плана для дисциплины ООП, адаптированной к специфике использования СУО.

Главная решенная проблема заключается в теоретически-праксиологическом обосновании эффективности процесса обучения объектно-ориентированному программированию путем разработки педагогической модели обучения, на основе университетского курса ООП, сосредоточенной на использование СУО и направленной на создание компетенции ООП для будущих программистов (в том числе для будущих специалистов в области образовательных наук (ОН)), с целью оптимизации процесса формирования компетенций ООП.

Теоретическая и практическая значимость исследования состоит из разработанной методологии и возможности внедрения в обучении дисциплины ООП с применением систем СУО, с целью создания компетенции ООП для будущих программистов и будущих специалистов в области ОН.

Внедрение результатов исследования: разработанная методология используется в преподавании дисциплины ООП на факультете Физики, Математики и Вычислительной Техники Тираспольского Государственного Университета (г. Кишинэу).

ANNOTATION

Gasnaş Ala. **Methodology of implementation of Learning Management Systems in the process of study of Object-Oriented Programming.** Doctoral thesis in pedagogical sciences. Chisinau, 2017.

Thesis structure: introduction, three chapters, general conclusions and recommendations, bibliography of 176 titles, 16 annexes, 140 pages of basic text, 33 figures, 34 tables. The results obtained are published in 10 scientific papers.

Key words: didactic process, object-oriented programming (OOP), pedagogical model, learning management systems (LMS), information technologies, interactive strategies, training priorities, OOP skills.

Field of study: Pedagogical Sciences. Didactics of Informatics.

Aim of the research: Theoretical foundation and development of pedagogical model for study process of the object-oriented programming through LMS.

Objectives of the research: (1) Determination of advantages deriving from the LMS and argumentation of the necessity for their implementation in the academic didactic process; (2) Specification of LMS with the most conforming characteristics of the university education system; (3) Identification of object-oriented language appropriate to study of the concept of object-oriented programming; (4) Development of the pedagogical model centered on the implementation of the LMS in the didactic process at the OOP discipline; (5) The scientific-methodological argumentation of the efficiency of using the pedagogical model in studying the OOP discipline; (6) Validation by pedagogical experiment of the efficiency of the elaborated pedagogical model.

The scientific novelty and originality of the research consists in: a) identifying the methods of using LMS in teaching-learning-evaluation of informatics; b) elaboration of the pedagogical model for studying the OOP university by implementing the LMS; c) elaboration of the methodology for the implementation of this model; d) elaboration of the curriculum in OOP course, adapted to the specifics of the use of LMS.

The solved scientific problem consists in the theoretical-praxiological substantiation of the efficiency of the process of study of the course Object Oriented Programming by developing a teaching-learning-evaluation pedagogical model based on the POO university course focused on the use of LMS, oriented towards the formation of the OOP's competence to future specialists in computer science (including in the field of Education Sciences (ES)), which effect is optimizing the Object-Oriented Programming competency.

The practical value of the research results from the elaborated methodology and the possibility of its integration within the OOP university course through the implementation of the LMS in order to train the OOP for IT specialists and ES.

Implementation of the scientific results: the elaborated methodology is used in the teaching of the OOP university course within the Faculty of Physics, Mathematics and Information Technologies at the State University of Tiraspol.

LISTA ABREVIERILOR

CPOO	– competență de programare orientată pe obiecte
DI	– design instrucțional
ICS	– instruire centrată pe student
ÎAD	– învățare autodirijată
ÎBÎ	– învățare bazată pe întrebări
ÎBP	– învățare bazată pe problemă
ÎBPr	– învățare bazată pe proiect
ÎD	– învățământ la distanță
ÎM	– învățământ mixt
MFTI	– matematică, fizică și tehnologii informaționale
Moodle	– Modular Object-Oriented Dynamic Learning Environment
OO	– orientat pe obiecte
POO	– Programare Orientată pe Obiecte
SMÎ	– Sistem de Management al Învățării
SPSS	– Statistical Package for the Social Sciences (program statistic pentru științe sociale)
ȘE	– științe ale educației
TI / (IT)	– tehnologii informaționale / information technologies
TIC	– tehnologii informaționale și de comunicație
TOO	– tehnologie orientată pe obiecte
UNESCO	– Organizația Națiunilor Unite pentru Educație, Știință și Cultură
UPSC	– Universitatea Pedagogică de Stat <i>Ion Creangă</i> din Chișinău
UST	– Universitatea de Stat din Tiraspol (cu sediul la Chișinău)

INTRODUCERE

Actualitatea temei. Tehnologiile informaționale și de comunicație (TIC) au devenit, într-un timp scurt, parte din elementele indispensabile ale societății moderne. Țările dezvoltate consideră alfabetizarea TIC componentă esențială a educației, alături de citire, scriere și numărare.

Unul dintre obiectivele prioritare ale UNESCO [172] este promovarea accesului la cele mai bune tehnologii educaționale, astfel încât generația tânără să fie capabilă să se integreze ușor în societatea modernă.

Elementele esențiale ale acestui obiectiv sunt:

- menținerea capacității de consiliere a guvernelor naționale cu privire la utilizarea TIC în educație, ținând cont de specificul local;
- sprijinirea țărilor în dezvoltarea propriilor programe și materiale educaționale;
- stimularea alfabetizării, a formării inițiale și a învățării continue (formale și nonformale) de calitate pentru toți și la toate nivelurile.

Strategia europeană în domeniul educației și formării profesionale *ET 2020* [173] pune accent pe consolidarea sistemelor de formare profesională, pentru ca acestea să devină mai atractive, relevante, orientate spre carieră, inovative, accesibile și flexibile, adaptate necesităților pieței forței de muncă.

După cum se menționează în Strategia de dezvoltare a educației pentru anii 2014–2020 *Educația–2020* [174], în contextul viziunii de integrare europeană a Republicii Moldova, cadrul european al competențelor-cheie trebuie incluse în strategiile și politicile țării. Astfel, Strategia națională de dezvoltare a societății informaționale *Moldova Digitală 2020* [175] și Programul național al securității cibernetice a Republicii Moldova [176] propun implementarea TIC în toate domeniile economice și educaționale din țară.

Una dintre direcțiile prioritare de informatizare a societății este informatizarea educației. Acest proces presupune utilizarea metodelor și mijloacelor TIC atât pentru intensificarea calității și eficacității procesului instructiv-educativ, cât și pentru dezvoltarea personală.

Strategia *ET 2020* are drept obiectiv referitor la e-Learning mobilizarea comunităților educaționale și culturale, precum și a agenților din mediile economic și social, cu scopul de a accelera schimbările din sistemele de formare inițială și formare continuă în contextul unei societăți europene bazate pe cunoaștere. Acest deziderat constituie un pas important către realizarea conceptului de *educație permanentă*, bazat pe tehnologie.

În organizarea și desfășurarea formării inițiale a specialiștilor din toate domeniile, un accent aparte este pus pe *Sistemele de Management al Învățării* (SMÎ). SMÎ sunt medii de învățare tehnologice care înglobează cursuri online și oferă servicii sincrone și asincrone cuprinzătoare, ce susțin învățarea prin colaborare.

În instituțiile de învățământ superior, SMÎ încep să fie adoptate ca sisteme de gestionare a resurselor și a activităților educaționale. Colaborările dintre actorii procesului de instruire pot fi îmbunătățite și consolidate prin integrarea tehnologiilor educaționale de ultimă generație în activitățile de învățare. SMÎ, inițial percepute ca accesorii simple de predare și învățare, devin instrumente vitale în procesul educațional.

Conform lui Arthur Recesso [3], o interfață bazată pe tehnologie folosește software, Internet sau alte tehnologii aferente calculatorului ca mijloc de: a) replicare a strategiilor eficiente de instruire; b) furnizare în mod interactiv a conținutului; c) evaluare a învățării.

Schimbările pedagogice se produc destul de rapid și de aceea sunt necesare SMÎ flexibile, adaptabile ușor la cerințele didactice noi. Astfel, SMÎ moderne oferă instrumente pentru a sprijini învățarea activă, fiind flexibile și adaptabile la evoluția cerințelor pedagogice.

Instrumentele TIC pun la dispoziție sisteme pentru crearea unor medii de învățare prin colaborare și permit circuitul global al cunoștințelor.

Astfel, tehnologiile noi conțin instrumente care pot fi folosite pentru îmbunătățirea procesului de predare-învățare, iar SMÎ permit furnizarea diferitor tipuri de resurse educaționale, precum și interacțiunea în timp real dintre profesor și student și student-student.

Actualitatea temei abordate în teză este determinată de faptul că instituțiile de învățământ își pun problema de modificare a tehnologiilor didactice actuale, în vederea realizării unui proces educativ performant, astfel încât specialistul format să fie *competitiv*. În acest context, SMÎ au revoluționat procesul didactic prin implementarea celor mai eficiente concepte moderne: centrarea educației pe student; învățarea prin colaborare; axarea pe competențe și finalități etc.

Descrierea situației în domeniul de cercetare și identificarea problemelor de cercetare. Dezvoltarea sistemelor e-Learning a schimbat paradigma educațională, deplasând accentul de la profesor spre student, modificând astfel rolul cadrului didactic din furnizor de informație în mentor. Instruirea centrată pe student (ICS) încurajează profesorii să optimizeze actul de predare-învățare-evaluare, distribuind resursele la îndemâna celui instruit și realizând activități prin intermediul tehnologiilor noi.

Există mai multe cercetări privind îmbunătățirea calității cursurilor de informatică cu ajutorul TIC. Astfel, E. Doberkat [55], F. Corbera [56], V.V. Riabov și B.J. Higgs [57], X. Huan [58], M. Palumbo și F. Verga [59], J. Ramos [60], S. Ozkam [61] au dezvoltat diferite *produse-*

program pe baza tehnologiei Orientate pe Obiecte, pentru a sprijini conținutul unui curs de bază pentru calculator: laboratoare virtuale pentru predarea cursurilor de informatică; materiale video în care se demonstau pașii scrierii, compilării și corectării unui program în limbaj C++. Cercetările lor au demonstrat eficiența implementării sistemelor e-Learning în cadrul cursurilor de educație informatică.

Cercetătorii R.L. Martens [41], R. Reiser [42], T.M. Duffy [43], C.A. Wolters [44], Th. Bastiaens și R. Martens [45], J. Herrington și R. Oliver [46] și-au axat cercetarea pe *modele de învățare înglobate în SMÎ*: învățarea independentă, activă, autodirijată, învățarea prin simulare și muncă în echipe virtuale.

E. Islas [50], E. Douglas și Van Der Vyver [51], J.B. Arbaugh [53], R.B. Fich [54], J. Gilbert [55] au analizat *dezvoltarea SMÎ-urilor* din punctul de vedere al tehnologiei bazate pe componente de e-Learning și au investigat abilitățile obținute de studenți în urma participării la cursurile unui SMÎ.

Savanții P.W. Pat și M. Chan [64], F. Martin [63] au conchis că, folosind tehnologiile informaționale și strategia adecvată a schimbului de informație, calitatea și cantitatea cunoștințelor pot crește în mod semnificativ. Din lucrările lor tragem concluzia că *SMÎ au fost utile pentru dezvoltarea abilităților de folosire a calculatorului și pentru creșterea competențelor în domeniul informaticii*.

Xiaoli Huan [58] a atestat *creșterea eficacității la cursurile de informatică online* prin implementarea materialelor didactice video și a instrumentelor de colaborare.

Problema implementării SMÎ în sistemul educațional a fost cercetată și de către savanții ruși B.A. Красильникова [72], Д.С. Кобылкин [73], Ю.А. Гладышева [74], care au menționat că stimularea creativității în pregătirea viitorilor specialiști poate fi realizată prin dezvoltarea de proiecte.

Cercetătorii Н.И. Заводчикова și У.В. Плясунова [75] analizează *particularitățile metodelor de instruire prin folosirea SMÎ Moodle*.

Totodată, Ю.В. Позняк, А.С. Гаркун și А.А. Царева [76] specifică *avantajul utilizării SMÎ Moodle în păstrarea abordării individuale a învățării și posibilitatea organizării activităților de grup*.

Impactul utilizării instrumentelor TIC în predarea cursurilor de informatică a format obiectul de studiu al mai multor cercetători din Republica Moldova: A. Gremalschi [85], L. Chiriac [86], S. Gâncu [83], S. Corlat [81], A. Braicov [84], L. Mihălache [88], A. Globa [89], lucrările cărora au fost reflectate într-un șir de publicații științifico-didactice.

Disciplina *Programarea Orientată pe Obiecte* a fost introdusă în curriculumurile instituțiilor de învățământ superior care pregătesc specialiști în domeniul Informaticii. În același timp, în literatura de specialitate sunt atestate un șir de dezacorduri privind caracterizarea POO. D.J. Armstrong [16] și O. Nierstrasz [17] sugerează că *nu există consens în ceea ce privește conceptele de bază ale POO*. Alte studii au arătat că studenții se confruntă cu diferite dificultăți de învățare a POO. Astfel, A. Fjuk, A. Karahasanovic și J. Kaasboll [94], A. Fleury [95], N. Ragonis și M. Ben-Ari [96], C. Schulte și J. Niere [97], P. Luker [98] menționează *dificultățile întâmpinate la înțelegerea conceptelor orientate pe obiecte și a relației dintre aceste concepte*.

Ca urmare a studierii literaturii de specialitate în domeniul POO, e-Learning și a modalităților de implementare a SMÎ în universitățile de peste hotare și în Republica Moldova, s-a constatat importanța utilizării SMÎ în procesul de studiere a disciplinei *Programarea Orientată pe Obiecte*. În același timp, au fost elucidate un șir de **contradicții**:

- 1) între dezvoltarea fulminantă a tehnologiilor din domeniul e-Learning și implementarea insuficientă a acestora în procesul didactic;
- 2) între posibilitățile oferite de tehnologiile TIC privind procesul de studiu prin implementarea SMÎ și lipsa unor repere conceptuale și praxiologice privind utilizarea acestora;
- 3) între cerințele existente pe piața muncii față de specialiștii în informatică (inclusiv în domeniul *Științe ale Educației*) în raport cu abilitatea de programare și nivelul de pregătire al studenților din universități la disciplina *POO*.

Contradicțiile enumerate ne permit să formulăm următoarea **problemă de cercetare**: fundamentarea teoretico-praxiologică a eficientizării procesului de studiere a *Programării Orientate pe Obiecte*, prin elaborarea unui model pedagogic de predare-învățare-evaluare în baza cursului universitar *Programarea Orientată pe Obiecte*, axat pe utilizarea Sistemelor de Management al Învățării și orientat spre formarea competenței de Programare Orientată pe Obiecte la viitorii specialiști în informatică (inclusiv în domeniul *Științe ale Educației*).

Obiectul cercetării: eficientizarea formării competenței POO prin implementarea SMÎ în procesul de studiu al *Programării Orientate pe Obiecte*.

Scopul cercetării: fundamentarea teoretică și elaborarea unui model pedagogic a procesului de studiu al *Programării Orientate pe Obiecte* prin intermediul SMÎ.

Obiectivele cercetării:

1. Determinarea avantajelor oferite de Sistemele de Management al Învățării și argumentarea necesității implementării lor în procesul didactic universitar.
2. Precizarea Sistemelor de Management al Învățării cu caracteristici optime pentru sistemul educațional universitar.

3. Identificarea limbajului orientat pe obiecte adecvat studierii conceptului *Programării Orientate pe Obiecte*.

4. Elaborarea modelului pedagogic centrat pe utilizarea Sistemelor de Management al Învățării în procesul de studiu al *Programării Orientate pe Obiecte*.

5. Argumentarea științifico-metodologică a eficienței utilizării modelului pedagogic în studierea *Programării Orientate pe Obiecte*.

6. Validarea prin experiment didactic a eficienței modelului pedagogic elaborat.

Metodele de cercetare folosite pentru a atinge obiectivele studiului au fost următoarele:

- *metode teoretice*: cercetarea și documentarea științifică; analiza literaturii de specialitate; comparația; sinteza; generalizarea; sistematizarea; proiectarea, descrierea și modelarea pedagogică;

- *metode experimentale*: experimentul pedagogic; observarea, chestionarea, testarea, analiza și evaluarea;

- *metode de analiză*: prelucrarea statistică a datelor experimentale; analiza cantitativă și calitativă a rezultatelor obținute experimental.

Noutatea științifică a rezultatelor cercetării constă în: a) identificarea instrumentelor SMÎ pentru predarea-învățarea-evaluarea *Informaticii*; b) elaborarea modelului pedagogic de studiu al *Programării Orientate pe Obiecte* prin intermediul SMÎ; c) elaborarea metodologiei de implementare al acestui model; d) proiectarea curriculumului la cursul *Programarea Orientată pe Obiecte*, adaptată la specificul utilizării SMÎ.

Problema științifică soluționată constă în fundamentarea teoretico-praxiologică a eficientizării procesului de studiere a *Programării Orientate pe Obiecte*, prin elaborarea unui model pedagogic de predare-învățare-evaluare în baza cursului universitar *Programarea Orientată pe Obiecte*, axat pe utilizarea Sistemelor de Management al Învățării și orientat spre formarea competenței de Programare Orientată pe Obiecte la viitorii specialiști în informatică (inclusiv în domeniul *Științe ale Educației*), având ca efect optimizarea procesului de formare a competenței *Programarea Orientată pe Obiecte*.

Semnificația teoretică a lucrării constă în cercetarea modalităților de implementare a SMÎ în procesul de predare-învățare-evaluare a *Informaticii* și elaborarea unui model pedagogic de studiu a cursului universitar *Programarea Orientată pe Obiecte*.

Valoarea aplicativă a lucrării. În cadrul Universității de Stat din Tiraspol a fost aprobat și aplicat în procesul didactic modelul pedagogic elaborat pentru predarea-învățarea-evaluarea cursului universitar *Programarea Orientată pe Obiecte* prin implementarea SMÎ. Modelul pedagogic elaborat poate fi utilizat la cursul *Programarea Orientată pe Obiecte*. El poate fi ușor

adaptat în procesul de instruire la orice curs de informatică. De asemenea, au fost elaborate reperele instructiv-metodice la disciplina universitară *POO*.

Principalele rezultate științifice înaintate spre susținere:

1. Au fost determinate caracteristicile optime ale Sistemelor de Management al Învățării pentru sistemul educațional universitar și s-a identificat că sistemul Moodle este un SMÎ cu caracteristici optime.

2. Modelul pedagogic centrat pe utilizarea Sistemelor de Management al Învățării în procesul de studiu al Programării Orientate pe Obiecte.

3. Metodologia de implementare a modelului pedagogic elaborat.

4. Curriculumul la cursul universitar *Programarea Orientată pe Obiecte*, adaptat la modelul pedagogic.

5. Resursele digitale pentru cursul universitar *POO*, plasate pe SMÎ Moodle.

Implementarea rezultatelor științifice a fost realizată în cadrul experimentului pedagogic desfășurat pe eșantioane de control și experimentale, care au cuprins 84 de studenți de la specialitățile *Informatică* (Științe Exacte), *Informatică și Matematică*, *Matematică și Informatică*, *Fizică și Informatică* (Științe ale Educației) în Universitatea de Stat din Tiraspol și Universitatea Pedagogică de Stat *Ion Creangă* din Chișinău.

Aprobarea rezultatelor cercetării a fost realizată în corespundere cu etapele fundamentale ale studiului. Principalele rezultate ale cercetării au fost prezentate, discutate și aprobate la ședințele catedrelor *Informatică și Tehnologii Informaționale* și *Didactica Matematicii, Fizicii și Informaticii* din cadrul Universității de Stat din Tiraspol, precum și în cadrul conferințelor științifice naționale și internaționale:

- Conferința științifico-didactică națională cu participare internațională *Probleme actuale ale didacticii științelor reale*, consacrată aniversării a 80 de ani de la nașterea profesorului universitar Andrei Hariton. Chișinău, 4–6 octombrie, 2013;

- Conferința științifico-didactică națională cu participare internațională *Învățământul superior din Republica Moldova la 85 de ani*. Chișinău, 24–25 septembrie, 2015;

- Conferința științifică internațională *Mathematics & Information Technologies: Research and Education. Dedicated to the 70th anniversary of the Moldova State University*. Chișinău, 23–26 iunie, 2016.

- Conferința științifică internațională *The Fourth Conference of Mathematical Society of the Republic of Moldova. Dedicated to the centenary of Vladimir Andrunachevici*. Chișinău, 28 iunie – 2 iulie, 2017.

- Conferința științifică internațională *The Conference on Applied and Industrial Mathematics*: Iași, România, 15-16 septembrie 2017;
- Conferința științifico-didactică națională cu participare internațională ediția a II-a, *Probleme actuale ale didacticii științelor reale*, consacrată aniversării a 80-a a profesorului universitar Ilie Lupu. Chișinău, 11–12 mai, 2018.

Lucrările științifice și științifico-metodice publicate la tema tezei sunt expuse în 10 publicații: 4 articole științifice în reviste naționale de categoriile B și C; 6 comunicări la conferințe științifice.

Sumarul compartimentelor tezei

În *Introducere* este argumentată actualitatea temei de cercetare, gradul de cercetare a temei, importanța ei. Sunt definite problema de cercetare, scopul și obiectivele cercetării; este descrisă noutatea științifică, importanța teoretică și valoarea aplicativă a lucrării, precum și aprobarea rezultatelor științifice obținute.

Capitolul 1, ***Repere teoretice ale Sistemelor de Management al Învățării în procesul de studiu al Programării Orientate pe Obiecte***, este structurat în trei paragrafe. Acest capitol este dedicat studierii și analizei situației în domeniile e-Learning și POO; este cercetată literatura de specialitate din perspectiva implementării SMÎ în procesul didactic; sunt evidențiate facilitățile aduse de implementarea SMÎ în procesul de studii. Este analizat rolul sistemelor de management al învățării în educație. A fost realizată analiza comparativă a sistemelor de management al învățării, ceea ce a permis stabilirea celor mai bune SMÎ pentru aplicarea în procesul didactic la studierea cursului POO. Cercetarea caracteristicilor de bază ale limbajelor de programare orientate pe obiecte au determinat alegerea limbajului de programare POO în predarea cursului respectiv.

Capitolul 2, ***Bazele metodologice de formare a competenței de Programare Orientată pe Obiecte prin utilizarea Sistemelor de Management al Învățării***, este dedicat elaborării și fundamentării teoretice a modelului pedagogic centrat pe utilizarea SMÎ în procesul didactic la disciplina *Programarea Orientată pe Obiecte*. Au fost identificate componentele modelului și ale nucleului lui, care constă din cinci module integratoare: *Proiectare, Activități, Instrumente, Resurse și Evaluaare*. În același capitol sunt reflectate aspectele metodologice privind utilizarea SMÎ Moodle în procesul de studiere a disciplinei *Programarea Orientată pe Obiecte*.

În capitolul 3, ***Demersuri experimentale în implementarea Sistemelor de Management al Învățării în procesul de studiu al Programării Orientate pe Obiecte***, este descris experimentul de constatare și cel de formare; este efectuată analiza statistică a rezultatelor

experimentului pedagogic prin utilizarea aplicațiilor SPSS și MS Excel. Pentru demonstrarea ipotezelor de cercetare au fost aplicate testul parametric t-Student și testul neparametric (U) Mann–Whitney. Pentru a determina tendința grupurilor cercetate a fost calculat indicatorul d al lui Cohen și coeficientul de corelare Pearson.

La finele fiecărui capitol sunt prezentate concluziile respective.

Cuvinte-cheie: proces didactic, Programare Orientată pe Obiecte (POO), model pedagogic, Sistem de Management al Învățării (SMÎ), tehnologii informaționale, strategii interactive, instruire centrată pe student, competențe POO, finalități de studii.

1. REPERE TEORETICE ALE SISTEMELOR DE MANAGEMENT AL ÎNVĂȚĂRII ÎN PROCESUL DE STUDIU AL PROGRAMĂRII ORIENTATE PE OBIECTE

1.1. Analiza comparativă a sistemelor de management al învățării

Modelul tradițional de educație este organizat astfel încât cadrului didactic îi revine rolul activ în procesul didactic, cel de emițător spre un receptor aproape pasiv, de regulă, în sălile de clasă. Tehnologiile informaționale și de comunicație oferă sisteme pentru crearea de medii colaborative de învățare și permit schimbul global de cunoștințe [1]. În acest context, e-Learningul devine o parte importantă a procesului didactic universitar. Unele universități folosesc e-Learningul pentru a îmbunătăți sistemul de instruire tradițional, pe când altele au creat modele de alternativă, bazate pe învățare virtuală, și folosesc aceste modele ca metode noi de învățare.

Potrivit lui M.J. Rosenberg [2], prima și cea mai importantă caracteristică a e-Learningului este faptul că se realizează într-un mediu de rețea. Acest lucru înseamnă că fiecare calculator este în comunicare constantă cu un server central, iar materialele de e-Learning sunt accesibile prin intermediul unui browser pe un computer personal.

Conform lui Arthur Recesso [3], o interfață bazată pe tehnologie folosește software, Internet sau alte tehnologii aferente calculatorului ca mijloc de replicare a strategiilor eficiente de instruire, pentru furnizarea în mod interactiv a conținutului, pentru facilitarea și evaluarea învățării.

Este important de a aplica tehnologiile existente pentru a asigura o interfață între profesor și cursant, folosind strategii de instruire adecvate și eficiente, pentru a îmbunătăți cunoștințele celor instruiți (figura 1.1).

Așadar, e-Learningul poate fi definit ca "utilizarea noilor tehnologii multimedia și Internet pentru îmbunătățirea calității învățării prin facilitarea accesului la resurse și servicii, precum și colaborarea și schimbul de informație la distanță" [4, p. 2]. Se știe că e-Learningul se bazează pe un sistem de management al învățării, care tot mai frecvent se utilizează în universități. Integrarea sistemelor de management al învățării în educație joacă un rol vital în epoca digitală. Sistemele de Management al Învățării sunt medii de instruire tehnologice, care înglobează cursuri online și oferă servicii sincrone și asincrone ample, ce susțin învățarea colaborativă [5].

În instituțiile de învățământ superior, SMÎ-urile sunt adoptate ca sistem de gestionare a cunoștințelor pentru obținerea și susținerea schimbului de cunoștințe între studenți. Colaborarea

studenților poate fi îmbunătățită și consolidată prin implementarea TIC și strategia adecvată a schimbului de cunoștințe.

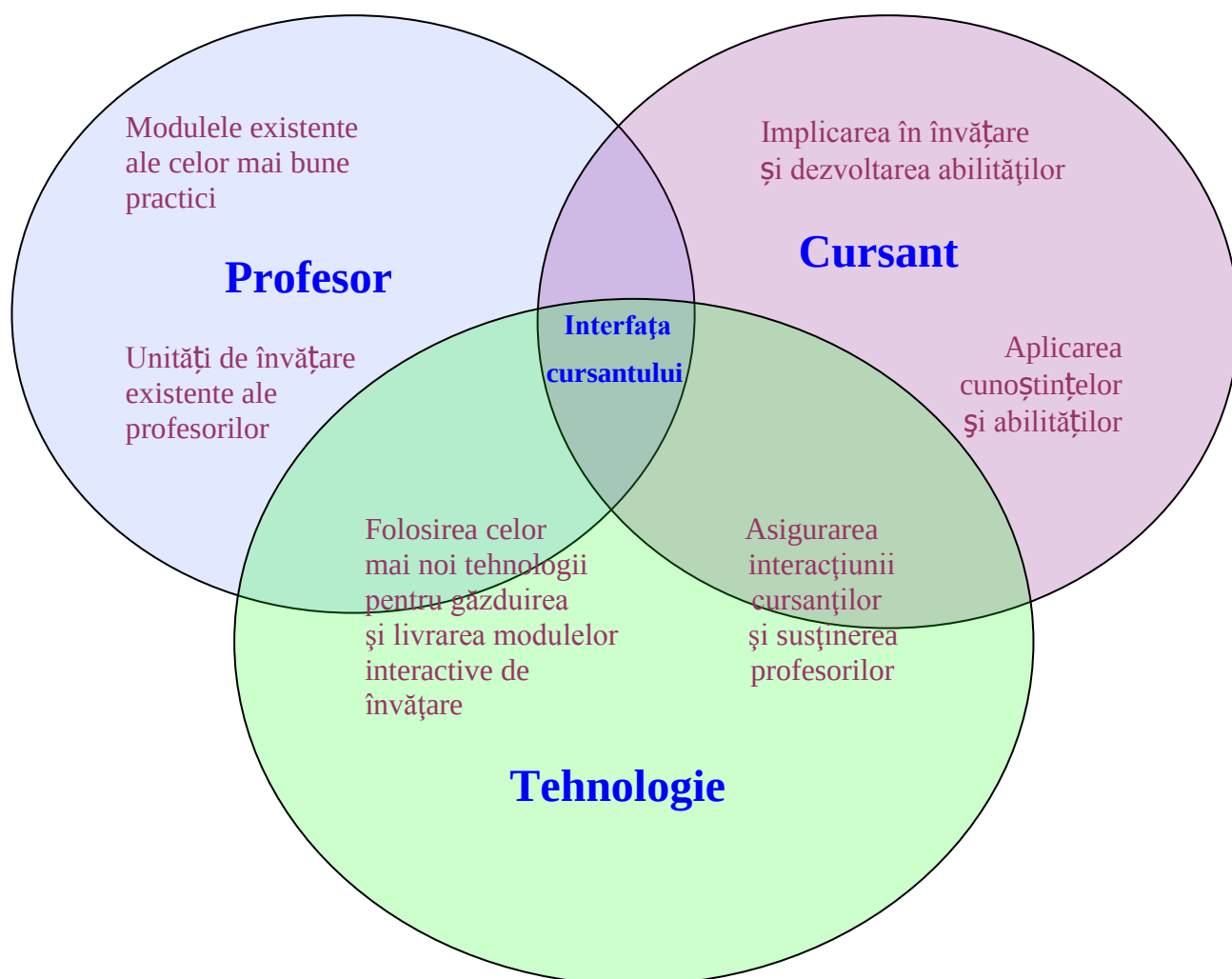


Fig. 1.1. Interfața SMÎ pentru cursant (după Arthur Recesso [3])

Ce este un SMÎ (Learning Management System)

SMÎ este software-ul principal al soluțiilor de e-Learning. SMÎ este „un software care automatizează administrarea evenimentelor de instruire” (figura 1.2) [6].



Fig. 1.2. Funcționalitățile unui SMÎ

Sistemele SMÎ gestionează fluxul activităților de învățare (notificarea utilizatorilor, aprobarea de către manager), furnizează resurse de învățare online, urmăresc activitățile și rezultatele cursantului și prezintă rapoarte de management. Un SMÎ poate include funcții suplimentare, cum ar fi: crearea de conținuturi, instrumentele de colaborare a cursantului (chat, grupuri de discuții etc.) [6].

Sistemele de Management al Învățării joacă un rol central în scenariul de e-Learning bazat pe web. SMÎ-urile conectează conținuturile de învățare și studenții într-o manieră standardizată. Ele gestionează utilizatorii, materialele și evenimentele de învățare; gestionează și administrează progresul instruirii și ține evidența performanței de învățare; gestionează sarcini administrative [3].

Învățământul superior integrează tehnologii educaționale de ultimă generație în activitățile sale, iar SMÎ-urile nu mai sunt simple accesorii de predare și învățare, ci devin instrumente indispensabile în procesul educațional.

SMÎ-ul extinde în format online activitățile din sala de cursuri, interconectând studenții și instructorii lor. Prin intermediul tehnologiilor SMÎ, profesorii pot mări impactul predării prin instrumentele online, iar studenții pot folosi aceste instrumente pentru a spori contactul cu profesorii, colegii și cu informația.

După cum menționează R.K. Ellis [7], un sistem de management al învățării este o aplicație software pentru administrarea, documentarea, urmărirea și anunțarea programelor de instruire, evenimentelor de clasă și online, programelor e-Learning și a conținutului de instruire. SMÎ este

un software proiectat și dezvoltat pentru a urmări și a gestiona instruirea și educația bazată pe calculator. Conform lui Ellis, un SMÎ trebuie să fie responsabil de:

- centralizarea și automatizarea administrării
- utilizarea self-service-ului și service-ului de autoghidare
- asamblarea și livrarea conținutului de învățare într-un mod rapid
- consolidarea acțiunilor de instruire pe o platformă bazată pe web
- asigurarea portabilității și a standardelor de sprijin
- personalizarea conținutului și reutilizarea cunoștințelor.

Într-un SMÎ, interacțiunea are loc prin instrumentele de comunicare sincrone sau asincrone, care favorizează utilizarea strategiilor de stimulare a participării active a studenților la dialog. SMÎ sunt sisteme bazate pe web, care le permit profesorilor și studenților să-și distribuie materiale, să trimită și să returneze sarcini și să comunice online.

În SMÎ, medierea implică atât dobândirea de competențe și abilități de comunicare, cât și crearea posibilităților practice de aplicare a activităților colaborative, în care procesul de învățare se desfășoară într-o manieră participativă. În acest scop, profesorul se bazează pe instrumentele de comunicare, cum ar fi chatul, forumul, blogul, înregistrări video. Un SMÎ trebuie să folosească progrese tehnologice de ultimă oră, din motive de eficiență și pentru asigurarea gradului maxim de interactivitate și comunicare între utilizatori.

Utilizatorii principali ai unui SMÎ sunt: cursanții, instructorii și administratorul.

Cursanții folosesc sistemul de management al învățării pentru a accesa resursele educaționale și consultațiile tutoriale în orice moment și din orice locație. *Instructorii* sunt profesorii și asistenții care utilizează SMÎ pentru a verifica, a ajuta și a evalua cursanții. *Administratorul* menține controlul asupra funcționării corecte a sistemului.

Beneficiile aduse procesului didactic de SMÎ:

• Folosind strategii interactive de învățare, SMÎ-ul crește motivația studenților, promovează învățarea, încurajează interacțiunea, oferă feedback, iar ajutorul poate fi furnizat în timpul procesului de învățare [8].

• Un SMÎ acceptă conținutul de învățare în diferite formate (multimedia, video sau text) [9].

• Un SMÎ oferă posibilitatea de a accesa materialul de curs în orice moment. Conținutul cursului este actualizat și cursanții pot vedea modificările efectuate într-un anumit domeniu. Profesorii pot modifica informațiile cursului în funcție de necesitățile studentului [10].

- Prin intermediul unui SMÎ se pot organiza diferite activități instructiv-educative pentru cursanți.

- Un SMÎ permite reutilizarea activităților de învățare. Prin reutilizarea conținutului pot fi reduse timpul și efortul, precum și costul pentru îmbunătățirea conținutului online [11].

Probleme/riscuri ce apar la utilizarea unui SMÎ:

- Unii profesori au abilități reduse de utilizare a calculatorului și le lipsesc abilitățile de gestionare a informațiilor necesare pentru a folosi cu succes un SMÎ. Personalul didactic, în această situație, trebuie nu numai să învețe cum să activeze în cadrul acestor medii, dar, de asemenea, să dezvolte diferite moduri de utilizare a SMÎ-urilor în predare-învățare-evaluare [10].

- Multe cadre didactice sunt provocate să proiecteze și să organizeze un complex de activități ce nu ar fi potrivite pentru nevoile studentului [10].

- Într-un SMÎ există riscul de a transfera practicile precare de predare.

- Unele cercetări sugerează că predarea online conduce la o creștere a volumului de muncă a cadrelor didactice [12].

Un SMÎ are următoarele **caracteristici**:

- Opțiuni de înregistrare și înscriere pentru profesori și studenți.
- Adăugarea/ștergerea unor cursuri de către persoanele abilitate.
- Determinarea rolului distinct între utilizator și contul de utilizator.
- Stabilirea calendarului de curs.
- Încărcarea și preluarea Sarcinilor și Resurselor.
- Modulul *Forum*.
- Servicii video/audio
- Chatul
- Jurnal online / note

În figura 1.3 este prezentată schematic interacțiunea utilizatorului cu sistemul de management al învățării, cu descrierea diferitor tipuri de utilizatori ai sistemului, precum și a diverselor modalități de interacțiune cu SMÎ.

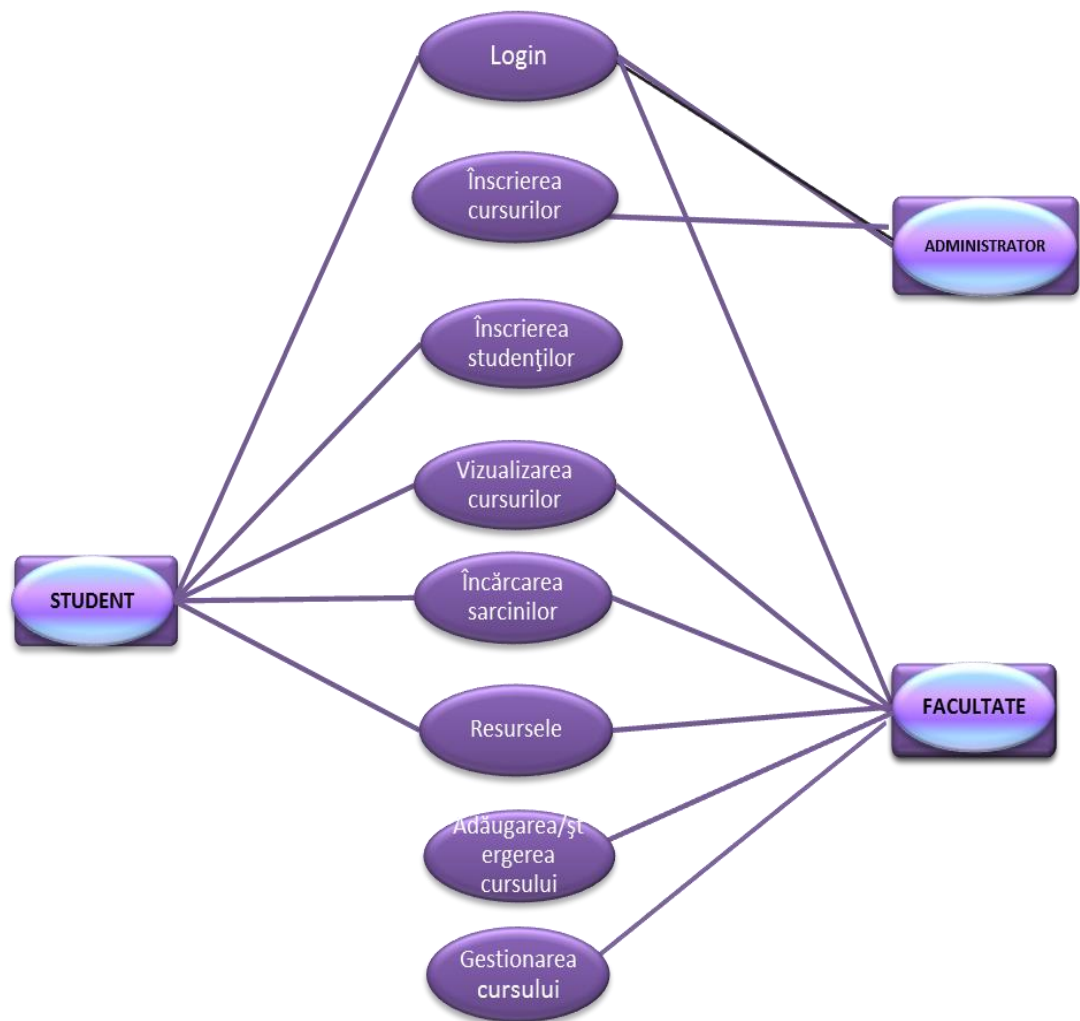


Fig. 1.3. Structura schematică a unui SMÎ [10]

Un SMÎ servește, în primul rând, misiunea academică a instituției. Conținutul disciplinelor de predare și învățare se schimbă mereu, de aceea e nevoie de a evalua în mod regulat SMÎ-ul, pentru a fi siguri că acesta satisface misiunea academică și obiectivele strategice ale instituției de învățământ. Schimbările în procesul didactic se produc destul de rapid și în acest caz e nevoie de o platformă SMÎ flexibilă. Un SMÎ este mai mult decât un curs plasat pe web. Un SMÎ bun trebuie să ofere instrumente pentru a sprijini învățarea activă și trebuie să fie flexibil și adaptabil la evoluția necesităților de instruire.

Pe piață există o varietate mare de SMÎ-uri, care sunt disponibile fie comercial, fie în Open Source. Ambele tipuri de sisteme sunt utilizate pe scară largă în lumea întreagă. Open Source

este codul-sursă al unui software, care este pus la dispoziția publicului pentru extindere și modificare, în funcție de nevoile utilizatorului. După datele din literatură privind evaluarea SMÎ, liderul SMÎ comercial este Blackboard/Angel/WEBCT, iar liderul SMÎ Open Source – Moodle (2002) [13].

În figura 1.4 este reprezentat traficul mediu la nivel mondial de SMÎ Blackboard și Moodle din 2004 până în prezent (grafic din Google Trends). Începând din anul 2007, tendința de utilizare a SMÎ Moodle a urcat peste nivelul de utilizare Blackboard.

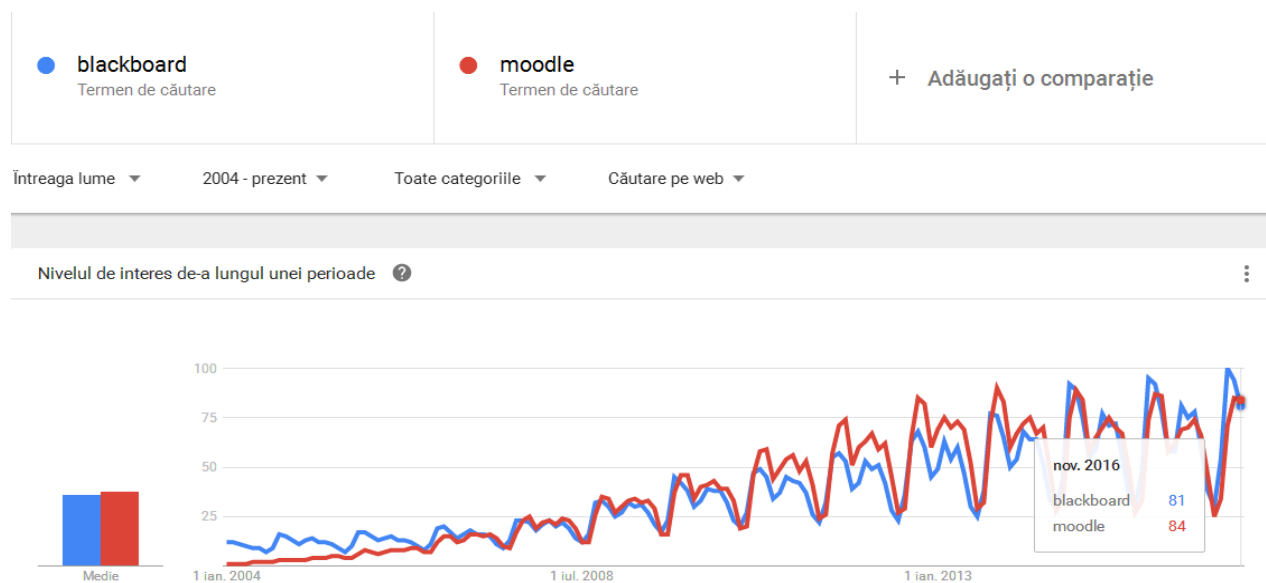


Fig. 1.4. Traficul de utilizare SMÎ (sursa: <https://www.google.com/trends/explore?q=blackboard,moodle>)

Să comparăm cinci dintre cele mai importante SMÎ Open Source [14]. Aceste sisteme au fost alese astfel încât să îndeplinească cel puțin următoarele condiții: sistemul este răspândit pe larg; comunitatea de dezvoltare este foarte activă; sistemul este utilizat de instituții relevante. Acești factori sunt principali pentru adoptarea pe termen lung a unui software Open Source.

Sistemele alese pentru evaluare au fost: Ilias, Moodle, Sakai, ATutor și Claroline. În figura 1.5 este reprezentat traficul mediu la nivel mondial al SMÎ Moodle, Sakai, Ilias, Claroline și ATutor din 2005 până în prezent.

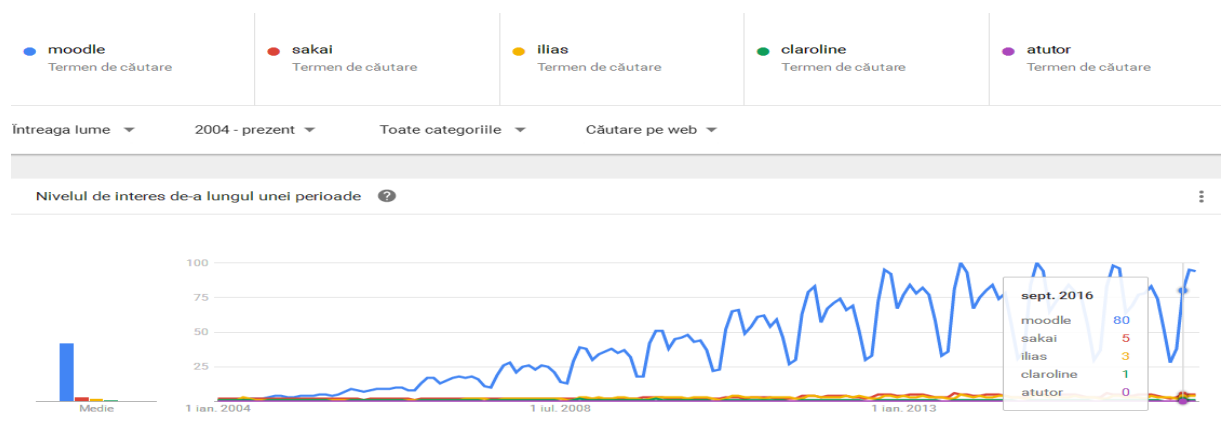


Fig. 1.5. Traficul de utilizare a SMÎ Ilias, Moodle, Sakai, ATutor, Claroline (sursa: <https://www.google.com/trends/explore?q=moodle,sakai,ilias,claroline,atutor>)

Actualmente există mai multe SMÎ Open Source puse la dispoziția publicului, fiecare având puncte forte și puncte slabe. Din acest motiv, pentru a putea face cea mai bună alegere, un potențial utilizator trebuie să fie bine informat. În tabelul 1.1. sunt prezentați cei mai importanți parametri folosiți la compararea SMÎ-urilor (conform cercetătorului Al-Ajlan [15]).

Tabelul 1.1. Parametrii folosiți la compararea SMÎ-urilor (după Al-Ajlan)

INSTRUMENTELE CURSANTULUI	INSTRUMENTELE DE SUPORT	SPECIFICAȚIA TEHNICĂ
1. Instrumente de comunicare • Forumuri de discuții • Schimb de fișiere / e-mail intern • Jurnalul online / note • Chat în timp real • Servicii video / Whiteboard	1. Instrumente de administrare • Autentificarea • Termenul de autorizare • Integrarea înregistrării • Servicii hosting	1. Hardware / Software • Browser la cerere client • Bază de date la cerere • Server software • Server Unix • Server Windows
2. Instrumente de productivitate • Semne de carte • Orientare / ajutor • Căutarea în curs • Calendar / analiză progres • Lucru offline sincronizat	2. Instrumente de remitere a cursului • Curs de management • Instructorul biroului de asistență • Instrumente de clasificare online • Controlul studentului • Testarea automată și notarea	2. Prețuri / licențiere • Profilul companiei • Costuri • Open Source • Dotări opționale • Versiune software
3. Implicarea studentului • Lucru în grup • Autoevaluare • Construirea comunității Student • Portofolii Student	3. Proiectarea curriculumului • Respectarea accesibilității • Template-urile cursului • Managementul curriculumului • Aspect personalizat • Acordul standardelor de instruire • Instrumente de proiectare a instruirii • Partajarea conținutului / re folosire	

Comunicarea este o parte esențială a educației. Un SMÎ tipic trebuie să conțină toate caracteristicile *instrumentelor de comunicare*, pentru a asigura o comunicare ușoară și feedback între instructori și cursanți, precum și între colegi. Numărul de SMÎ-uri crește periodic, iar SMÎ-urile Open Source existente sunt modificate și modernizate, pentru a satisface nevoile tuturor

utilizatorilor. Având în vedere aceste caracteristici, comparația dintre diferite SMÎ se va face analizând *Instrumentele cursantului* și *Instrumentele de comunicare*, acestea fiind cele mai elocvente caracteristici ale unui SMÎ. Importanța *instrumentelor de comunicare* în fiecare SMÎ nu poate fi neglijată. Din acest motiv, compararea caracteristicilor, cum ar fi serviciile video/whiteboard, schimbul de fișiere/e-mail intern, forumul de discuții, jurnalul online și chat pentru SMÎ, sunt concludente pentru o analiză comparativă.

Să analizăm succint serviciile menționate ale fiecărui SMÎ Open Source.

Forumul de discuții

ATutor permite dezvoltarea unei rețele de contacte; înființarea unei rețele de profil, la care se pot alătura diferite grupuri de interese; partajarea imaginilor și discuții.

Claroline are o comunitate de utilizatori și dezvoltatori care se întâlnesc ocazional, pentru a discuta cu privire la modalitățile de a ajuta studenții și profesorii să interacționeze, folosind SMÎ-ul Claroline.

Ilias prezintă forumuri de discuții desfășurate pentru utilizatorii din întreaga lume. Ilias permite schimbul de informații și opinii privind modul de a îmbunătăți instruirea continuă prin utilizarea SMÎ-ului lor .

Sakai prezintă grupuri de discuții de dezvoltare, cunoscut și sub numele de *Sakai-Dev*. Aceste grupuri sunt formate pentru a perfecționa SMÎ-ul Sakai.

Moodle, spre deosebire de celelalte SMÎ-uri, posedă patru tipuri de forumuri de bază:

(1) *O singură intervenție simplă*; (2) *Forum standardizat pentru uz general*; (3) *Fiecare student publică o singură intervenție*; (4) *Forum întrebare și răspuns*. Prin intermediul acestor forumuri, studenții și profesorii fac schimb de idei prin postarea comentariilor și organizarea unor ateliere.

Putem concluziona că cel mai bun modul de *forum* există în SMÎ Moodle.

Schimbul de fișiere / e-mailul intern

În **ATutor** există un mail intern disponibil utilizatorilor pentru a trimite și a primi mesaje private de la alți utilizatori. Mesajele trimise sunt salvate în *sent message* și șterse după un anumit timp. Există totuși o posibilitate ca mesajele să fie exportate și salvate extern.

În **Claroline** nu există informații cu privire la schimbul de fișiere sau e-mailuri interne.

Ilias are un sistem de e-mail intern. Prin logare, mailurile pot fi trimise persoanelor fizice și membrilor grupurilor de învățare.

Sakai conține informații privind schimbul de fișiere și e-mail intern indisponibil.

În **Moodle**, profesorilor li se oferă căi ușoare de a prezenta materiale pentru studenții lor. Fișierele sunt încărcate și accesate prin intermediul Moodle. Tot ceea ce se cere de la cursanți este de a avea dreptul de a deschide software-ul.

Prin urmare, și în ceea ce privește serviciul de *e-mail intern* și *schimbul de fișiere* se poate spune că SMÎ Moodle este cel mai bun.

Jurnalul online / notele

În **ATutor**, pentru instructori sunt prevăzute încărcarea și gestionarea fișierelor conexe ale cursului. Există un manager de fișiere deschis alături de editorul de conținut și editorul de întrebări de testare. Acest lucru permite ca fișierele cursului să fie legate cu ușurință în pagini de conținut sau în itemi de testare.

Claroline cuprinde o caracteristică de jurnal online.

Ilias oferă diverse posibilități de import și de creare a conținutului pentru e-Learning. Se mândrește cu o platformă pentru crearea și publicarea de conținuturi.

În **Sakai** nu există informații privind un jurnal online disponibil pe site-ul Sakai.

Moodle are disponibil un modul de jurnal, oferind o zonă de text în care cursanții pot scrie. Acesta poate fi revăzut și actualizat.

Posibilitatea de a revizui și actualiza informația *jurnalului* plasează SMÎ Moodle asupra celorlalte SMÎ analizate.

Chatul

ATutor conține un chat bazat pe protocoalele XMPP și WAI-ARIA. Acesta constă din mesaje unu la unu și un chat multi-user între membrii cursului.

În **Claroline** nu există informații referitor la un chat în timp real disponibil pe website-ul Claroline.

Ilias oferă un chat-sistem pe baza JAVA, ce trebuie să fie instalat înainte de utilizare.

Sakai oferă instrumente, cum ar fi wiki, chat-uri și bloguri, pentru a efectua dezbateri.

În **Moodle**, chatul în timp real le permite participanților să aibă o discuție sincronă în timp real într-un curs Moodle.

Astfel, încorporarea chatului în SMÎ Moodle reprezintă un avantaj al acestui sistem.

Whiteboard / servicii video

În **ATutor** este utilizată o mesagerie AComm și instrumentul de whiteboard. AComm are caracteristici interesante, cum ar fi funcționalitatea completă a tastaturii, elaborarea desenelor fără utilizarea unui mouse, în cazul în care există un utilizator cu probleme de vedere; posedă o funcție care permite ascultarea unui text

Claroline nu conține nicio caracteristică whiteboard.

Ilias nu are nicio caracteristică whiteboard. Are conferință video și funcția de conferințe video online disponibile, pe care instructorii și cursanții le folosesc pentru a face schimb de idei și informații.

Sakai nu conține whiteboard și nu furnizează informații cu privire la serviciile video.

Moodle are una dintre cele mai bune caracteristici whiteboard. Există tablă Skype și tablă interactivă, disponibilă pentru cursanți și instructori.

Comparația acestor cinci SMÎ Open Source arată că Moodle și ATutor au cele mai bune instrumente de comunicare, cu o interfață ușor de utilizat. Informația de pe paginile web Moodle și ATutor este ușor accesibilă. De asemenea, și Ilias face disponibilă informația pentru clienții potențiali. Claroline și Sakai sunt SMÎ-uri cu pagini web complexe, ceea ce face obținerea informației destul de greoaie.

Astfel, putem concluziona că prin încorporarea instrumentelor colaborative, ce reprezintă un element important în conceptul instruirii centrate pe student, SMÎ Moodle posedă cele mai bune mijloace de comunicare și le pune la dispoziția utilizatorului în cel mai prietenos și confortabil mod.

1.2. Studiu comparativ al caracteristicilor de bază ale limbajelor de programare orientate pe obiecte

Programarea Orientată pe Obiecte(POO) a devenit una dintre cele mai importante paradigme de programare utilizate pe larg și este susținută de mai multe limbaje de programare. În prezent, C++ este unul dintre cele mai disponibile limbaje de programare și majoritatea programatorilor sunt familiarizați cu noțiunea de *programare orientată pe obiecte* prin experiența lor de lucru cu C++. Limbajul respectiv oferă un suport eficient pentru programarea orientată pe obiecte, însă există și alte limbaje care implementează conceptele POO în diferite moduri.

Noțiunea de *limbaje orientate pe obiecte* a apărut odată cu apariția limbajului Simula (creat în perioada 1962–1967), dar și acum se atestă dezacorduri privind caracterizarea programării orientate pe obiecte. În 2006, D.J. Armstrong [16] sugerează că nu există un consens în ceea ce privește conceptele de bază ale programării orientate pe obiecte. Acest lucru face anevoioasă descrierea și definirea limbajelor OO, deoarece nu există un acord asupra unei definiții universale a POO. Alți autori (O. Nierstrasz [17]) consideră că toate limbajele care suportă încapsularea au un anumit grad de orientare pe obiecte. Încapsularea este o caracteristică de bază a programării orientate, dar aceasta nu înseamnă și obiect-orientată – pentru aceasta e necesară prezența moștenirii. „Însă chiar și prezența caracteristicilor de încapsulare și moștenire nu face limbajul de programare complet funcțional în ceea ce privește POO. Principalele beneficii ale POO se manifestă numai atunci când în limbajul de programare este implementat polimorfismul” [18].

Limbajele orientate pe obiecte diferă prin modul de aplicare a caracteristicilor de bază POO. De exemplu, unele limbaje suportă moștenirea multiplă, pe când altele nu o suportă.

Drept urmare, în procesul de formare inițială a specialiștilor IT și a cadrelor didactice în informatică se studiază un șir de limbaje de programare orientate pe obiecte, precum: C++, C#, Java, Object Pascal, PHP, JavaScript etc.

În continuare vom analiza caracteristicile de bază ale POO și modalitatea de implementare a acestora prin limbajele C++, C#, Java, Object Pascal, PHP și JavaScript.

Învățarea unui limbaj de programare presupune examinarea sintaxei lui, dar și a conceptului de implementare a algoritmilor cu acest limbaj [19]. Deoarece conceptul POO se regăsește în fiecare dintre limbajele menționate, în procesul de predare-învățare a acestor concepte este importantă cercetarea următoarelor aspecte:

- a) Stabilirea caracteristicilor POO comune, deci identificarea celor mai importante particularități POO;
- b) Determinarea caracteristicilor POO specifice fiecărui limbaj;
- c) Identificarea celor mai potrivite limbaje de programare pentru studierea conceptului POO.

Acest studiu permite evitarea suprapunerilor de conținuturi curriculare la învățarea limbajelor POO, precum și elaborarea ulterioară a unor strategii de predare a conceptului POO în cadrul formării inițiale a specialiștilor IT și a cadrelor didactice în informatică.

Caracteristicile de bază ale Programării Orientate pe Obiecte

Studiul eficient al limbajelor orientate pe obiecte necesită cunoașterea profundă a caracteristicilor de bază ale POO, abordate în continuare.

Clasele reprezintă tipuri de date abstracte, care înglobează comportamentul și datele asociate unui concept. Comportamentul este descris cu ajutorul metodelor, iar datele – cu ajutorul atributelor. Atributele reprezintă caracteristicile unui obiect, iar metodele sunt acțiunile pe care el le poate face. Obiectele sunt elementele definite printr-o clasă. Fiecare astfel de obiect creat poartă denumirea de *instanță a clasei* (definirea unui obiect poartă numele de *istanțiere*).

O clasă prevede mecanismul de bază prin care atributele și metodele formează un concept comun. Ea oferă o descriere a comportamentului obiectelor instanțiate. Paradigma orientată pe obiecte presupune că metodele dintr-o clasă nu se bazează pe algoritmi comuni. Metodele depind de nivelul de detaliere la care este modelat obiectul. Astfel, o clasă definește o grupare logică de metode și atribute și acționează ca un mijloc prin care sunt atinse abstractizarea și încapsularea datelor.

Abstractizarea. Prin abstractizare se izolează aspectele esențiale de cele neesențiale, în funcție de perspectiva de abordare a problemei de rezolvat. Nivelul de detaliere a acestor aspecte depinde de obiectul ce este abstractizat și de cerințele problemei. În consecință, pentru aceeași problemă pot exista mai multe abstractizări, deci mai multe modele. Prin esență, o abstractizare este o reprezentare incompletă a realității și anume aceasta dă valoare modelului corespunzător.

Încapsularea este principiul conform căruia un obiect trebuie să aibă interfața (metodele publice ale unei clase) complet separată de implementare. Toate datele și codul implementării trebuie să fie complet ascunse în spatele interfeței. După crearea unei interfețe, aplicația poate interacționa cu obiectele. Încapsularea servește la separarea interfeței de implementarea acesteia. Rezultă că un obiect este format din două părți distincte: interfața și, respectiv, implementarea acestei interfețe. Încapsularea este conceptul complementar abstractizării. Abstractizarea este procesul prin care este definită interfața obiectului, pe când încapsularea definește reprezentarea obiectului împreună cu implementarea interfeței.

Încapsularea permite păstrarea unei limite clare între o clasă și lumea exterioară și oferă programatorilor libertatea de a schimba funcționarea internă a unei clase.

Moștenirea. În programarea orientată pe obiecte, conceptul de moștenire este deosebit de important. Moștenirea este mecanismul de bază cu ajutorul căruia pot fi create modelele ierarhice ale claselor. Posibilitatea de a crea o arborescență de clase, de la cea mai generală la cea mai specifică, permite un control mai bun al programelor, permițând, totodată, lizibilitatea programelor de către utilizatorii textelor-sursă sau dezvoltatorii de programe. Moștenirea este o formă de reutilizare a codului, în care noile clase sunt create din clase existente, numite ulterior *clase de bază* sau *superclase*. Crearea claselor noi (*subclaselor* sau *claselor derivate*), pornind de la o clasă de bază, permite moștenirea proprietăților originale ale clasei respective prin absorbirea atributelor și comportamentelor lor, prin înlocuirea unor comportamente și prin adăugarea unor attribute și comportamente noi.

Prin urmare, moștenirea este mecanismul de obținere a unei clase de la o altă clasă, păstrând totodată toate proprietățile și metodele clasei-părinte (de bază) și adăugând, dacă este necesar, unele proprietăți și metode noi. Prin ea se promovează reutilizarea codului și dezvoltarea codului structurat.

Mecanismul moștenirii multiple îi permite unei clase să moștenească trăsături din multiple clase. Dar acesta este un mecanism mai periculos, deoarece proprietățile și metodele unei superclase, prin moștenirea multiplă, pot fi moștenite de către o clasă derivată din ea prin mai multe căi.

Polimorfismul. Într-o ierarhie de clase obținute prin moștenire, o metodă poate avea forme diferite de la un nivel la altul (specifice respectivului nivel de ierarhie) și poate funcționa diferit în diverse obiecte. Polimorfismul este un fenomen în care unul și același cod este realizat diferit, în funcție de clasa sau tipul obiectului utilizat la apelarea acestui cod. Polimorfismul este asigurat de faptul că în clasa derivată se schimbă realizarea metodei din clasa de bază, dar cu păstrarea „signaturii” metodei (același nume și aceiași parametri de intrare/ieșire).

Există mai multe tipuri de polimorfism, care au fost grupate în două categorii mari: 1) polimorfismul ad-hoc și 2) polimorfismul universal [20]. *Polimorfismul ad-hoc* se obține atunci când o funcție lucrează sau pare că lucrează cu mai multe tipuri diferite și se poate comporta în mod independent pentru fiecare tip. *Polimorfismul universal* este atunci când o clasă sau o funcție se comportă la fel pentru o mulțime de tipuri. Principala diferență dintre polimorfismul ad-hoc și cel universal constă în faptul că funcțiile polimorfe ad-hoc execută un cod distinct pentru un mic set de tipuri, potențial independente, pe când funcțiile polimorfe universale execută același cod pentru un număr infinit de tipuri (de exemplu, pentru toate tipurile admisibile) [21].

Polimorfismul ad-hoc, la rândul său, se divizează în două tipuri: supraîncărcarea și conversia. *Supraîncărcarea* utilizează același nume de funcție pentru a se referi la diferite funcții reale, care se disting prin tipul și numărul de argumente. *Conversia* permite valorii unui tip să fie convertită la valoarea altui tip. Conversia este o operație semantică necesară pentru a converti un argument de un tip la tipul așteptat de o funcție. De exemplu, o valoare întreagă poate fi folosită în cazul în care este așteptată o valoare reală, și invers. Conversia poate fi statică, inserând în mod automat conversia tipului necesar între argumente și funcții în timpul compilării, sau conversia tipului necesar poate fi determinată dinamic prin testele runtime (în timpul execuției). Conversia poate reduce dimensiunea programului și poate îmbunătăți lizibilitatea lui, dar poate, de asemenea, cauza erori subtile de sistem, iar uneori chiar periculoase.

Polimorfismul universal conține și el două mari categorii: polimorfismul parametric și polimorfismul de includere, ultimul în limbajele orientate pe obiecte purtând denumirea de „polimorfism de moștenire” [22]. *Polimorfismul parametric* realizează mecanismul prin care se poate defini o metodă cu același nume în aceeași clasă (funcțiile trebuie să difere prin numărul și/sau tipul parametrilor). Selecția funcției se realizează la compilare, ceea ce se numește *legarea timpurie* (early binding). *Polimorfismul de moștenire* este mecanismul prin care o metodă din clasa de bază este redefinită cu aceiași parametri în clasele derivate. Selecția funcției se va realiza la rulare, ceea ce este *legarea întârziată* (late binding, dynamic binding, runtime binding) [23].

Polimorfismul oferă o comoditate semnificativă în programarea orientată pe obiecte prin furnizarea structurilor pentru gestionarea diferitor obiecte. Spre exemplu, metodele ce fac lucruri similare (de exemplu, operația de adunare) pot avea același nume cu diferite „signaturi” (în acest caz vorbim despre *polimorfism de supraîncărcare*); aceeași clasă/metodă (de exemplu, operația de sortare) poate lucra cu mai multe tipuri de obiecte (în acest caz vorbim despre *polimorfism parametric*); o subclasă poate fi substituită de o clasă-părinte (*polimorfism de moștenire*).

Limbaje de programare orientate pe obiecte

Un limbaj de programare poate fi considerat orientat pe obiecte dacă are mecanisme pentru implementarea caracteristicilor POO: abstractizarea, încapsularea, moștenirea, polimorfismul.

O diferență în rândul limbajelor POO îl aduc noțiunile de limbaj *pur* și limbaj *hibrid*. Limbajele POO pure sunt cele care permit doar un singur model de programare – POO. Se pot declara clase și metode, dar nu sunt prezente funcții și proceduri simple, vechi și date globale. Java, C# sunt limbaje POO pure (în sensul definiției de mai sus). C++ și Object Pascal, în schimb, sunt două exemple tipice de limbaje hibride, care le permit programatorilor să folosească limbajul tradițional C și abordări de programare Pascal.

Pentru a studia cu succes un limbaj orientat pe obiecte, este necesar de a reflecta în procesul de predare–învățare care dintre caracteristicile POO menționate se regăsesc în unele dintre cele mai solicitate limbaje de programare.

În continuare vom descrie momentele importante care trebuie să însoțească analiza caracteristicilor POO.

Limbajul C++ este un limbaj de programare orientat pe obiecte utilizat pe larg și oferă o bază bună pentru comparație. C++, fiind derivat din limbajul C, combină avantajele oferite de acesta (eficiența, flexibilitatea și popularitatea) cu avantajele oferite de tehnica POO.

Acest limbaj este unul hibrid, deoarece susține atât programarea procedurală, cât și cea orientată pe obiecte, spre deosebire de limbajul Java, care este unul orientat pe obiecte pur.

Deși adoptă principiile programării orientate pe obiecte, C++ nu impune aplicarea lor strictă. Conceptul fundamental în C++, ca și în alte limbaje POO, este *clasa*. El asigură încapsularea datelor, inițializarea datelor, gestiunea memoriei controlată de utilizator, mecanisme pentru supraîncărcarea operatorilor.

Limbajul C++, spre deosebire de alte limbaje POO, oferă atât moștenirea simplă, cât și moștenirea multiplă. În acest limbaj lipsește mecanismul garbage-collection (mecanismul de colectare a gunoiului – GC – este o formă de management automat al memoriei), dar aici este implementată tratarea excepțiilor. C++ folosește legarea întârziată (late binding), ceea ce

înseamnă că programatorul trebuie să precizeze clasa specifică obiectului, sau cel puțin clasa generală la care poate aparține obiectul. „Pe de o parte, acest lucru face să crească eficiența la lansare și codul să fie de dimensiuni mici, pe de altă parte însă, pierde din puterea de reutilizare a claselor” [24]. C++ poate utiliza pointerii pentru a manipula memoria – o activitate necesară pentru scrierea componentelor sistemului de operare low-level.

Limbaajul Java are o sintaxă similară cu C++, astfel studierea lui devine mult mai ușoară pentru cei care au deprinderi de lucru cu limbaajul C++. Java este un limbaaj orientat pe obiecte pur. Limbaajul dat nu acceptă construcții low-level, precum le avem în C++ (cum ar fi pointerii), ceea ce permite colectorului garbage să realoce obiectele referite. În Java, clasa este structura fundamentală. Toate clasele din Java și C# descind dintr-o clasă de bază, numită *obiect* [25]. Limbaajul Java oferă moștenire, tratarea excepțiilor, modularitate, colector garbage, polimorfism etc. Dar Java susține doar moștenirea simplă, spre deosebire de C++, în schimb oferă interfețe (interfața definește un șir de metode care vor fi implementate de una sau mai multe clase).

Limbaajul C# a fost conceput ca limbaaj de programare orientat pe obiecte pur. Este un limbaaj ce permite programarea structurată, modulară și orientată obiectual, conform perceptelor moderne ale programării. Principiile de bază ale Programării Orientate pe Obiecte– încapsularea, moștenirea, polimorfismul – sunt elemente fundamentale ale programării C#. La baza acestuia stă limbaajul C, dar sunt preluate și principiile de programare din C++. Sintaxa este apropiată și limbaajului Java: ca și acest limbaaj, are colector garbage și se compilează într-un limbaaj intermediar [26]. În C#, conceptele de clasă, moștenire și polimorfism sunt la fel ca și în celelalte limbaje POO.

Limbaajul Object Pascal este un limbaaj de programare derivat din Pascal, ce permite folosirea structurilor din modelul programării orientate pe obiecte: obiecte, moștenire, polimorfism.

Un element al limbajelor POO este modelul obiectual. Unele limbaje tradiționale POO permit crearea de obiecte în stiva, heap sau stocarea statică. În aceste limbaje, o variabilă de tip clasă corespunde unui obiect din memorie. Acesta este modul în care funcționează C++. În ultimul timp se utilizează un alt model, numit *model de referință obiect*. În acest model, fiecare obiect este alocat dinamic în heap, și o variabilă de tip clasă este de fapt o referință sau un pointer la obiectul din memorie. Java, C# și Object Pascal adoptă acest model de referință.

Object Pascal nu are colector garbage. Cu toate acestea, componentele Delphi (Delphi este un limbaaj de programare și un mediu de dezvoltare pentru programe; el este cunoscut ca *Pascal orientat pe obiecte*) susțin ideea unui obiect-proprietar: proprietarul devine responsabil pentru distrugerea tuturor obiectelor pe care le deține. Delphi utilizează, de asemenea, și

mecanismul de numărare a referințelor pentru șiruri de caractere, tablouri dinamice și interfețe, eliberând obiectele din memorie, atunci când nu mai are linkuri [27]. Acest lucru face manipularea cu distrugerea obiectelor destul de simplă. Destructorii din Object Pascal sunt similari cu destructorii din C ++.

Limbajul PHP (*Hypertext Preprocessor*) este un limbaj de scripting, derivat din familia C. Limbajul PHP rulează pe un server web și poate fi integrat în formatul HTML, pentru a da un conținut dinamic paginii web.

Începând cu PHP 5, a fost rescris modelul obiectual, pentru a permite o performanță mai bună a limbajului și introducerea mai multor caracteristici. Limbajul PHP tratează obiectele la fel ca referințele, ceea ce înseamnă că fiecare variabilă conține mai degrabă o referință la obiect, decât o copie a întregului obiect [28].

Limbajul JavaScript este un limbaj de programare orientat pe obiecte independent de sistemul operațional folosit în calculator. Spre deosebire de C# sau C++, JavaScript folosește o altă abordare pentru a crea un limbaj orientat pe obiecte. Acesta este un limbaj bazat pe prototip. Conceptul de *prototip* implică faptul că un comportament poate fi reutilizat prin clonarea obiectelor existente, care servesc drept prototipuri. Fiecare obiect în acest limbaj este descendentul unui prototip, care definește un șir de funcții-membre pe care le poate folosi obiectul [29]. În JavaScript nu există noțiunea de clasă, sunt doar obiecte. Fiecare obiect poate fi apoi utilizat ca un prototip pentru un alt obiect.

Analize comparative

Pentru a efectua o comparare a caracteristicilor POO, propunem o analiză a gradului și a specificului de implementare a acestor caracteristici în limbajele de programare orientate pe obiecte, descrise anterior.

Moștenirea și polimorfismul în limbajele POO. Moștenirea este un mecanism ce realizează relația de ierarhie în modelul de clasă. Relația ierarhică poate fi extinsă la toate clasele din limbajele Java și C#. Aceste limbaje conțin o clasă generică, care este un strămoș pentru toate clasele din limbaj. Prezența unui singur strămoș asigură toate clasele cu anumite funcționalități minime, care sunt moștenite de la acest strămoș. C++ nu are o clasă-strămoș. Avantajul acestei abordări este faptul că pentru funcționarea unei aplicații nu este nevoie de a se lega cu întreaga ierarhie de obiecte. În cazul Java, la lansarea fișierului executabil, pentru orice aplicație trebuie să fie prezente toate clasele din ierarhie.

Accesibilitatea. În limbajele C++ și Java, prin cuvintele-cheie *public*, *private* și *protected*, aplicate la metodele și atributele clasei de bază, se controlează nivelul de vizibilitate. Atât C++, cât și Java definesc trei nivele de acces: *public*, *private* și *protected*. Cu nivelul de acces *public*,

membrii clasei de bază sunt vizibili pentru toate clasele. Caracteristica *private* nu permite vizibilitatea la nici o altă clasă, în afară de clasa în care este declarată. Totuși, în C++ există o excepție: limbajul C++ are, de asemenea, noțiunea de *clasă-prietenă*. Acest limbaj permite claselor-prietene accesul chiar și la metodele și atributele private. Caracteristica *protected* specifică faptul că membrul este vizibil pentru codul din aceeași clasă și clasele descendente.

În limbajul C# sunt definite cinci nivele de acces pentru membrii clasei: *private*, *protected*, *internal*, *protected internal* și *public*. Modificatorul de acces *internal* specifică faptul că membrul este accesibil din interiorul clasei sau într-un bloc funcțional al unei aplicații. Un membru declarat ca *protected internal* va fi accesibil oricărui membru al clasei care îl conține și al claselor derivate, precum și în blocul funcțional.

Atunci când o clasă este moștenită, restricțiile vizibilității sunt moștenite și de subclase. Subclasa poate accesa variabilele protejate, dar nu le poate accesa pe cele private. Subclasa poate reduce vizibilitatea membrilor săi prin schimbarea caracteristicilor *public* cu caracteristici *protected* sau *private*. Caracteristica vizibilității nu poate fi crescută. Membrii ce posedă caracteristica *private* nu sunt vizibili nici chiar în subclasă și uneori ea poate fi cel mai bun mod de extensibilitate a clasei.

Există situații în care unele metode dintr-o superclasă (clasa de bază) e necesar să fie exportate la toți descendenții, dar la fel de esențial este ca aceste metode să nu fie redefinite. Același lucru este valabil și pentru atribute. În limbajele C# și Java există anumite cuvinte-cheie ("sealed", respectiv "final") care indică că aceste metode ori clase nu pot fi redefinite [30, 31]. Limbajul C++ nu are această abilitate.

În Obiect Pascal, pentru moștenire se utilizează o sintaxă specială, adăugând în paranteze numele clasei de bază. Acest limbaj acceptă doar un singur tip de moștenire – *public*. Clasele în Object Pascal sunt derivate dintr-o clasă de bază comună. Accesul la metodele clasei de bază se face prin cuvântul *inherited*. După acest cuvânt se scrie numele metodei din clasa de bază [32].

Object Pascal, la fel ca și limbajele Java și C#, nu susține moștenirea multiplă, în schimb utilizează opțiunea de *interfețe multiple*.

PHP folosește pe larg principiul de moștenire în modelul său obiectual. Acest principiu va influența modul de interacțiune dintre clase și obiecte. De exemplu, atunci când o clasă se extinde, subclasa moștenește toate metodele publice și protejate din clasa de bază. Aceste metode își vor păstra funcționalitatea lor originală, cu excepția cazului în care clasa redefinește aceste metode. Acest lucru este util pentru definirea și abstractizarea funcționalității și permite punerea în aplicare a funcționalităților suplimentare în scopuri similare [28].

În PHP, moștenirea se implementează folosind cuvântul-cheie *extends*. În acest limbaj poate fi pusă în aplicare moștenirea pe mai multe niveluri, adică moștenirea ierarhică, dar nu și cea multiplă.

Limbajele bazate pe prototipuri creează obiecte, fără a recurge la clase. Aceste limbaje sunt bazate pe definirea de prototipuri și manipularea lor. Atunci când este nevoie de un nou obiect, obiectul existent este clonat și apoi modificat. În astfel de limbaje nu există noțiunea de tip, ci doar cea de *similitudine* de obiecte. Când este creat un obiect nou, el poate defini un comportament nou și să păstreze un anumit comportament vechi. JavaScript este un limbaj orientat pe obiecte fără clase (class-free), și în loc de moștenirea clasică folosește moștenirea prototipică, iar obiectele moștenesc de la alte obiecte [29].

Clase abstracte. Există situații când în clasă este dată doar metoda ce trebuie să fie moștenită, fără a fi precizată și implementarea ei. Subclasa, în acest caz, este forțată să ofere implementări pentru toate metodele din superclasă. Acest mecanism poate fi utilizat pentru a se asigura că subclasa corespunde unui anumit design (construcții). Java și C# utilizează cuvântul-cheie *abstract* pentru a indica faptul că metoda sau clasa nu are nicio implementare și, prin urmare, subclasele trebuie să furnizeze definiția. Dar aceste clase abstracte nu pot fi instanțiate. Chiar dacă în clasă este doar o metodă abstractă, clasa nu poate fi instanțiată. Java face acest lucru explicit prin marcarea întregii clase ca fiind abstractă.

În limbajul C++ există posibilitatea de a defini clase abstracte, care sunt destinate creării de noi clase prin derivare, ele neputând fi instanțiate și utilizate ca atare. Ele constituie o bază în cadrul elaborării ierarhiilor de clase, putând fi folosite, spre exemplu, pentru a impune anumite restricții în realizarea claselor derivate. Pentru aceasta, se folosește cuvântul-cheie *virtual*. În vederea construirii unor astfel de clase, s-a introdus și conceptul de *funcție virtuală pură*. O astfel de funcție este declarată în cadrul clasei, dar nu este definită. O clasă care conține o funcție virtuală pură este considerată abstractă. Funcțiile virtuale pure trebuie definite în clasele descendente, altfel și acestea vor fi considerate abstracte.

În limbajul Object Pascal, clasele abstracte sunt clase care posedă metode abstracte sau care moștenesc metode abstracte. Pentru descrierea acestor metode, se folosește cuvântul-cheie *abstract*. Spre deosebire de C++ și Java, în Object Pascal poate fi creat un obiect al unei clase abstracte (deși, în acest caz, compilatorul va da un mesaj de avertizare).

Noțiunile de *clase abstracte* și *metode abstracte* sunt introduse și în PHP 5. În acest limbaj nu este permisă instanțierea unei clase abstracte. Ca și în C++, orice clasă ce conține cel puțin o metodă abstractă trebuie să fie abstractă. Metodele definite ca abstracte declară doar „signatura” metodei, implementarea acesteia se face în clasele descendente (derivate). Când se

moștenește de la o clasă abstractă, toate metodele marcate abstract în declarația clasei de bază trebuie să fie definite în clasele derivate. În plus, aceste metode trebuie să fie definite cu aceeași vizibilitate. De exemplu, în cazul în care metoda abstractă este definită ca fiind *protected*, implementarea ei trebuie definită ca *protected* sau *public*, dar nu ca *private* [28].

Moștenirea multiplă presupune posibilitatea ca o clasă să moștenească din mai multe clase-părinte. Acest mecanism trebuie folosit, însă, cu atenție, pentru că poate cauza o serie de ambiguități. Atunci când o clasă moștenește de la mai multe clase, aceasta nu este o versiune a uneia dintre clasele-părinte. Aceste probleme apar din cauza faptului că moștenirea multiplă definește o moștenire structurată ca un graf orientat aciclic, și nu ca un arbore. Într-un arbore există o cale unică de la orice clasă derivată la orice nod-strămoș. Într-o structură aciclică însă, pot exista mai multe căi între o subclasă și strămoșii săi. Atunci când subclasa se referă la o caracteristică moștenită, căutarea acestei caracteristici creează probleme semnificative.

În moștenirea multiplă, bazată pe structura de graf, apar probleme uzuale, precum conflicte de nume și moștenirea repetată. Conflictele solicită utilizarea diferitor tehnici pentru soluționarea lor. Limbajul C++ folosește pentru rezolvarea unor astfel de conflicte redenumirea metodelor. Problema de moștenire repetată este rezolvată în C++ prin utilizarea moștenirii virtuale. Atunci când o clasă este definită a fi moștenită virtual, C++ se asigură că există doar o singură copie a claselor în această cale. Aceasta rezolvă problemele moștenirii repetate.

Spre deosebire de C++, limbajele C#, Java, Obect Pascal, PHP nu implementează moștenirea multiplă. Ca o alternativă la moștenirea multiplă, limbajele date utilizează opțiunea de *interfețe multiple*. Deoarece o interfață nu specifică niciun fel de implementare, pot fi ”combinate” mai multe interfețe. Combinarea unor interfețe ce conțin o metodă cu același nume este posibilă doar dacă metodele nu au tipuri întoarse diferite și aceeași listă de argumente. Totuși, e bine ca în diferite interfețele ce trebuie combinate să nu existe metode cu același nume, deoarece acest lucru poate duce la confuzii evidente [33].

Polimorfismul. Supraîncărcarea. Limbajele C# și C++ folosesc cuvinte-cheie în scopul implementării caracteristicii polimorfism. Supraîncărcarea operatorilor este criticată deseori, deoarece permite programatorilor să acorde operatorilor o semantică complet diferită în funcție de tipurile operanzilor. Limbajul Java permite doar supraîncărcarea operatorilor aritmetici.

Limbajele Java, C++ și C# permit supraîncărcarea metodelor în mod similar. Atâta timp cât „signaturile” metodelor sunt diferite, compilatorul tratează metodele cu nume supraîncărcate, de parcă ar avea denumiri complet diferite. Mai mult decât atât, în aceste trei limbaje, supraîncărcarea se poate întâmpla atunci când o metodă din clasa de bază este moștenită într-o

subclasă ce are o metodă cu același nume, dar cu semnătură diferită. În acest caz, compilatoarele folosesc din nou legarea timpurie, pentru a face diferență între metodele supraîncărcate [34].

Prin intermediul Delphi și Free Pascal, în Object Pascal au fost introduse o serie de caracteristici noi. Printre acestea se numără și supraîncărcarea [27].

Interpretarea termenului de *supraîncărcare* în PHP este diferită de interpretarea în celelalte limbaje orientate pe obiecte. Supraîncărcarea tradițională oferă posibilitatea de a avea mai multe metode cu același nume, dar diferite cantități și tipuri de argumente. Supraîncărcarea în PHP oferă mijloace dinamice de "a crea" proprietăți și metode. Metodele de supraîncărcare sunt invocate atunci când interacționează cu proprietăți sau metode care nu au fost declarate, sau care nu sunt vizibile în domeniul de aplicare curent [35]. În PHP, toate metodele de supraîncărcare trebuie să fie definite ca publice.

Conversia în diferite limbaje este implementată diferit. C++ permite conversia implicită între tipurile sale, dar și pentru tipurile definite de utilizator, prin intermediul constructorilor de conversie și al operatorilor de conversie definiți de utilizator. Totodată, Java a lărgit conversiile implicite între tipurile native, dar alte conversii necesită o sintaxă explicită. Limbajul PHP folosește filtrele de conversie.

Polimorfismul de moștenire. Polimorfismul runtime (în timpul executării), sau redefinirea metodei, mai este numit și *legare întârziată* (late binding) sau *polimorfism dinamic*.

Posibilitatea de a se referi la obiectul comun din diferite clase dintr-o ierarhie și de a apela metoda clasei respective este foarte utilă, în cazul în care mai multe clase din această ierarhie redefinesc o metodă. Pentru aceasta, compilatorul trebuie să sprijine legarea întârziată, ceea ce înseamnă că în timpul executării se determină tipul real al obiectului și metoda ce trebuie apelată.

În C++, legarea întârziată este accesibilă doar pentru metodele virtuale. Metodele declarate în clasa de bază ca virtuale (*virtual*) susțin această caracteristică (dar numai dacă descrierea metodelor este identică). Metodele obișnuite, nonvirtuale, nu permit legarea întârziată, la fel ca în Object Pascal.

În limbajul Object Pascal, legarea întârziată este introdusă cu ajutorul cuvintelor-cheie *virtual* și *dynamic*. În clasele derivate, metodele redefinite trebuie marcate cu cuvântul-cheie *override*.

În limbajul Java, toate metodele folosesc legarea târzie în mod obligatoriu, cu excepția cazului în care este marcat ca *final* în mod clar.

În limbajul C#, polimorfismul de runtime se implementează prin redefinirea metodei din clasa de bază în clasa derivată. Acest lucru poate fi realizat prin utilizarea principiului de

moștenire și a cuvintelor-cheie *virtual* și *override*. Dacă în clasa de bază declarăm metode cu cuvântul-cheie *virtual*, atunci în clasa derivată aceste metode vor fi redefinite folosind cuvântul-cheie *override*.

În limbajul PHP, începând cu versiunea 5.3, a fost pusă în aplicare o caracteristică numită *late static binding* (*legare statică întârziată*), care poate fi folosită pentru a face referire la apelarea claselor în contextul moștenirii statice.

Din punctul de vedere al caracteristicilor de bază ale tehnologiei orientate pe obiecte, aceste limbaje includ în structura lor un șir de principii (tabelul 1.2).

Tabelul 1.2. Caracteristicile de bază ale limbajelor POO

Caracteristici de bază	Limbajul C++	Limbajul C#	Limbajul Java	Limbajul Obj. Pascal	Limbajul PHP	Limbajul JavaScript
Abstractizare	Suportă	Suportă	Suportă	Suportă	Suportă	Nu suportă (de prototip)
Încapsulare	Suportă	Suportă	Suportă	Suportă doar Public	Suportă	Suportă Public, Private
Moștenire simplă	Suportă	Suportă	Suportă	Suportă	Suportă	Nu suportă (de prototip)
Moștenire multiplă	Suportă	Nu suportă	Nu suportă	Nu suportă	Nu suportă	Nu suportă
Polimorfism	Suportă (ambele)	Suportă (întârziată)	Suportă (întârziată)	Suportă (întârziată)	Suportă (întârziată)	Nu suportă De prototip

Analizând aceste limbaje din perspectiva POO, am putea concluziona:

- Caracteristicile de bază ale POO sunt încapsularea, abstractizarea, moștenirea și polimorfismul. Ele vor constitui reperele pentru elaborarea curriculumului disciplinei *Programarea Orientată pe Obiecte*.

- Se cunoaște că fiecare limbaj a fost conceput pentru a soluționa probleme dintr-un anumit domeniu. Considerăm că, din punct de vedere metodologic, cele mai potrivite limbaje pentru studierea POO sunt C++ și C#.

- Există diferențe între modalitățile de implementare a caracteristicilor de bază ale POO pentru diferite limbaje de programare. Astfel, dacă în limbajele C# și Java există anumite cuvinte-cheie, care arată că unele metode dintr-o superclasă nu pot fi redefinite, limbajul C++ nu are această abilitate. Limbajele C#, Java, Object Pascal și PHP, spre deosebire de C++, nu implementează moștenirea multiplă. Spre deosebire de C++ și Java, în Object Pascal poate fi

creat un obiect al unei clase abstracte. Limbajul JavaScript are propriul concept obiectual și se deosebește radical de celelalte limbaje examinate.

- Fiecare dintre limbajele de programare orientată pe obiecte are domenii prioritare de aplicare, de aceea studierea caracteristicilor POO specifice este importantă pentru alegerea celui mai adecvat limbaj, în funcție de domeniul problemei care urmează a fi soluționată. Acest aspect este important în procesul de predare-învățare a POO.

- În proiectarea curriculară a disciplinei *Programarea Orientată pe Obiecte* este necesar de evidențiat caracteristicile de bază și analiza comparativă a limbajelor de programare orientate pe obiecte.

1.3. Tendințe de aplicare a SMÎ în predarea-învățarea-evaluarea informaticii

În ultimii ani, rolul TIC în educație a crescut și continuă să crească semnificativ. Un accent aparte în organizarea și desfășurarea formării inițiale a specialiștilor din toate domeniile este pus pe sistemul de management al învățării. SMÎ sunt medii de învățare care oferă o multitudine de instrumente sincrone și asincrone, ce susțin învățarea autodirijată, învățarea prin colaborare și cooperare [36].

Din perspectiva modelului tradițional de educație, cadrul didactic transmite cunoștințele în sălile de clasă. Noile tehnologii informaționale însă pun la dispoziție sisteme pentru crearea unor medii de învățare prin colaborare și permit circuitul de cunoștințe la nivel global [37]. În instituțiile de învățământ superior, SMÎ-urile sunt adoptate ca sisteme de gestionare a informațiilor, în scopul facilitării schimbului de cunoștințe, încurajând și consolidând, prin implementarea TIC și a unor strategii didactice adecvate, colaborarea dintre studenți.

Tehnologiile informaționale și de comunicație îmbunătățesc procesul de predare-învățare atât cantitativ, cât și calitativ. F. Concannon, A. Flynn și M. Campbell afirmă că „e-Learningul, bazat pe un raționament pedagogic solid, asigură feedback, interacțiune, acces la materialele de curs și este privit ca un beneficiu și un mijloc de îmbunătățire a calității predării” [38]. Avantajele aplicării TIC au fost recunoscute în științele geografice, inginerie, biologie, educație și psihologie. Prin urmare, TIC pot eficientiza învățământul universitar, concluzionează F. Pedro [39].

În prezent se atestă o creștere a cererilor pentru *învățământul la distanță*. În plus, dezvoltarea sistemelor e-Learning a schimbat paradigma educațională, deplasând accentul de pe profesor pe student (*Student Centred Learning*), modificând, astfel, rolul cadrului didactic din furnizor de informație în mentor [40]. Profesorii optimizează actul de predare-învățare, distribuind resursele la îndemână și realizând activități prin intermediul tehnologiilor noi. După

cum menționează R.L. Martens [41], e-Learningul înglobează modele de *învățare independentă, activă, autodirijată*, de *învățare prin simulare* și *muncă în echipe virtuale*. Cele mai multe dintre ele se bazează pe constructivism; grație acestora, după cum consideră R. Reiser [42], studenții devin responsabili pentru dirijarea propriului proces de învățare. Cei care desfășoară activități de autoreglare sunt motivați, independenți și metacognitiv activi (T.M. Duffy și colab. [43]; C.A. Wolters [44]; Th. Bastiaens și R. Martens [45], J. Herrington și R. Oliver [46]). Aceleași idei le exprimă și profesorul V. Cabac: „Stabilirea echilibrului între autoritatea profesorului și gradul de autonomie al studenților are o consecință importantă – asumarea de către student a responsabilității pentru rezultatele învățării, trecerea lui pe poziția de subiect al propriei învățări în drumul spre o maturitate intelectuală” [47, p. 15].

Un instrument potrivit pentru implementarea învățământului la distanță și a conceptelor amintite mai sus (*învățarea autodirijată, activă, independentă* ș.a.) sunt SMÎ (sau sistemele de e-Learning). Acestea reprezintă o parte componentă a procesului didactic în majoritatea universităților, inclusiv în cele din Republica Moldova. Inițial, SMÎ-urile au fost proiectate în scopul sprijinirii procesului de predare. Mai târziu, accentul s-a mutat pe învățare. Un SMÎ propune instrumente în scopuri atât pedagogice, cât și administrative, iar utilizarea lor în învățământul superior oferă flexibilitate și un acces mai mare la educație, modalități mai eficiente de predare și de îmbunătățire a procesului didactic [48].

SMÎ cuprind o multitudine de instrumente, începând de la controlul învățării și statisticile educaționale până la software flexibile în distribuirea cursurilor prin rețea și colaborare online. Ele pun la dispoziția fiecărui utilizator un mediu pentru interacțiune și mijloace pentru intercomunicare (profesor – cursant, cursant – cursant), oferind acces la informație oricând și oriunde. Aceste caracteristici accentuează utilitatea SMÎ, inclusiv pentru colectarea informațiilor despre actul de predare-învățare, în vederea monitorizării și asigurării unui proces instructiv-adaptiv.

SMÎ constituie o modalitate de sporire a eficienței predării. Aceste sisteme contribuie la facilitarea procedurii de livrare a cursurilor, la identificarea și folosirea resurselor, la organizarea diverselor activități colaborative (conferințe, chat-uri, forumuri), precum și la gestionarea suportului de învățare pentru studenți [49]. Multidisciplinare prin natura lor, SMÎ constituie un tip de sistem informatic ce combină mai multe entități tehnice și sociale. Prin urmare, succesul unui sistem poate fi considerat ca fiind un concept de ”probleme sociale” și ”probleme tehnice” și depinde de numeroase circumstanțe. Cercetători din diferite domenii – informatică, sisteme informaționale, psihologie, educație și tehnologie educațională – au încercat să evalueze sistemele respective. Astfel, E. Islas a analizat dezvoltarea SMÎ-urilor din punctul de vedere al

tehnologiei bazate pe componente de e-Learning [50]; E. Douglas și G. Van Der Vyver au studiat eficacitatea materialelor de curs [51]; J.B. Arbaugh și R.B. Fich au insistat pe importanța interacțiunii beneficiarilor unor cursuri online [52], iar cercetătorul J. Gilbert [53] a investigat abilitățile obținute în urma participării la cursurile unui SMÎ.

We Hu et al. [54] au arătat că mediul e-Learning este diferit de educația clasică. Analizând nu numai procesul de predare-învățare, ci și modul de interacțiune între profesori și studenți, de colaborare între studenți, în funcție de caracteristicile curriculumului, cercetătorii au concluzionat că acțiunile în cauză au un rol important în mediul e-Learning, deoarece acesta poate îngloba diferite stiluri de învățare, poate facilita învățarea printr-o varietate de activități și poate dezvolta abilitățile de utilizare a Internetului și a calculatorului.

Există mai multe cercetări privind îmbunătățirea calității cursurilor de informatică cu ajutorul TIC, inclusiv cu cel al SMÎ.

Această idee a fost prezentată în premieră în 2005 de către E.-E. Doberkat și colegii săi [55], care au lansat conceptul *e-Learning Software Engineering*, menit să susțină procesul educativ în domeniul ingineriei sistemelor soft.

F. Corbera et al. [56] au descris, în anul 2008, un nou produs-program – *Moodle CTPractical*, dezvoltat întru a sprijini conținutul unui curs de bază pentru calculator. Acest modul presupune o gamă vastă de activități educaționale, inclusiv de evaluare în cursurile pentru calculator.

În anul 2010, savanții V.V. Reabov și B.J. Higgs [57] au proiectat un laborator virtual pentru predarea cursurilor de informatică. Laboratorul a fost conceput pentru a ajuta formabilii să exploreze actualele metode de analiză și proiectare a sistemelor de calcul; să studieze astfel de compartimente ca: limbajele de programare, rețelele, simularea și modelarea, procesarea imaginilor, multimedia, bazele de date.

Mai târziu, în anul 2011, X. Huan și colegii săi au propus cursuri online de compilare a programelor (de exemplu, în limbajul C++) [58]. A fost elaborat un video în care se demonstau pașii scrierii, compilării și corectării unui program în limbaj C++, fiind trecute în revistă formatele folosite în mod obișnuit pentru a livra conținutul cursurilor: Microsoft PowerPoint (.ppt), Microsoft Word (.docx) și fișierele Adobe (.pdf).

În anul 2013, M. Palumbo și F. Verga [59] au integrat Moodle cu pachetul Office, oferind un sistem cu o funcționalitate flexibilă – *OpenOffice.org*. Ei au folosit Java și AJAX cu XML pentru a crea un applet Java, care a fost încorporat în pagina web a noului modul Moodle. Prin utilizarea acestor instrumente, studenții pot salva informația în unul sau în mai multe fișiere temporare de pe serverul Moodle.

Tot în anul 2013, J. Ramos și colegii săi [60] au propus utilizarea arhitecturii de evaluare Matlab, pentru extinderea funcționalității Moodle.

Cercetătorii S. Ozkam et al. [61] au prezentat un model complex de evaluare a SMÎ, care demonstrează eficiența implementării sistemelor e-Learning în cadrul cursurilor de educație informatică. Acest studiu a contribuit semnificativ la dezvoltarea domeniului, ghidând teoreticienii și practicienii în înțelegerea mai bună a felului în care poate fi sporită satisfacția cursanților, precum și a modalităților de optimizare a utilizării SMÎ.

A. Tripathi și M. Abhinav [62] au studiat utilizarea SMÎ în cadrul cursului *Securitate Informațională*. Rezultatele au arătat că majoritatea participanților au fost mulțumiți de mediul de învățare e-Learning. Lucrarea a adus în discuție folosirea celor mai recente tehnologii de informație și s-a concentrat pe interacțiunea dintre instructor și cursant prin încorporarea modulelor de chat.

Studiind utilizarea SMÎ în predarea cursului *Tehnologii Informaționale*, F. Martin [63] a urmărit să stabilească măsura în care conținutul și modul de predare au ajutat în procesul de aplicare a abilităților de lucru cu calculatorul. În accepțiunea sa, disciplina *Tehnologii Informaționale* este o parte fundamentală a curriculumului de licență. Studiul a relevat faptul că utilizarea SMÎ a fost utilă atât în predarea cursului, cât și pentru dezvoltarea abilităților de folosire a calculatorului.

Drept concluzie, aplicarea unui SMÎ, concomitent cu predarea aplicațiilor informatice de bază, contribuie la îmbunătățirea atât a competențelor în domeniul tehnologiilor informaționale, cât și a rezultatelor învățării.

Savanții P.W. Pat și M. Chan s-au preocupat de impactul utilizării unui SMÎ la o disciplină informatică. Pentru gestionarea materialelor și a activităților, a fost folosit SMÎ Moodle. Studenților li s-au oferit note de curs și materiale de lectură pe site-ul cursului. Pentru a utiliza funcțiile de atribuire și de test, sarcinile practice au fost puse la dispoziție online. Studenții au folosit forumul pentru a posta și discuta întrebările cu profesorul și colegii în regim online, ceea ce a consolidat învățarea prin colaborare. Evaluarea automatizată asigură consecvență, astfel încât studenții pot obține un feedback prompt. Proiectul de grup permite conlucrarea în echipă. Cercetarea și autoevaluarea antrenează studenții în reflecții asupra propriului proces de învățare. Așadar, folosind tehnologiile informaționale și strategia adecvată a schimbului de informație, calitatea și cantitatea cunoștințelor pot crește în mod semnificativ [64, p. 147].

Pe măsură ce succesul implementării SMÎ-urilor în universități este atestat tot mai mult, crește și numărul cursurilor oferite online. Atunci când se predau cursuri de informatică, apar anumite provocări pentru studenții aflați la distanță și au un program de învățare individual.

După cum menționează Xiaoli Huan et al. [58], predarea cursurilor de informatică implică și un nivel înalt de interactivitate între instructor și cursanți. Gândirea algoritmică este un proces complex și etapizat. A învăța programarea este o sarcină dificilă pentru mulți studenți, chiar și din grupele cu predare tradițională. Un student trebuie să fie capabil să utilizeze eficient calculatorul și softurile necesare, pentru a crea programe, a compila și a găsi soluții. Deoarece instructorul și cursanții sunt separați fizic, se creează o barieră semnificativă în calea formării competențelor de bază în utilizarea unui software nou sau în depanarea unui program. Prin urmare, în scopul de a sprijini și a crește eficacitatea la cursurile de informatică online, este indicat să se recurgă la materiale didactice video și instrumente de colaborare. Astfel, cercetătorii A. Akingbade și colegii săi, încă în 2003, au adus în discuție necesitatea utilizării în cadrul cursurilor de informatică online a softurilor de vizualizare [65].

În 2005, Z. Al-Khanjar și colegii au dezvoltat arhitectura SMÎ pentru extinderea funcționalității e-Learningului în sprijinul cursurilor de informatică online [66]. În anii următori, cercetătorii au propus instrumente software pentru corectarea automată a programelor din limbajele de programare în mediul SMÎ [67, 68]. S. Georgantaki și S. Retalis, prezentând mai multe instrumente software în predarea cursurilor de informatică, au insistat pe dificultățile ce apar dacă acestea nu sunt folosite în sesiunile practice, care sprijină învățarea conceptelor teoretice [69]. De exemplu, conținutul teoretic poate fi gestionat prin intermediul instrumentelor de e-portofoliu, cum ar fi Google Drives, Wiki-spații, bloguri și altele, care ar putea fi integrate în sistemele e-Learning [70]. În 2009, G. Robling et al. au încercat să îmbunătățească instrumentele software de vizualizare a exercițiilor [71].

Savanții ruși B.A. Красильникова [72], Д.С. Кобылкин [73], Ю.А. Гладышева [74] menționează că stimularea creativității în pregătirea viitorilor specialiști poate fi realizată prin dezvoltarea unor proiecte (individuale sau colective), care pot fi implementate ulterior în activitățile lor profesionale.

Н.И. Заводчикова și У.В. Плюсунова analizează particularitățile metodelor de instruire prin folosirea SMÎ Moodle, care îi ajută pe cursanți să își aprofundeze cunoștințele, să își îmbunătățească abilitățile câpătate și să își formeze abilități de autocontrol și planificare a timpului personal, precum și să înțeleagă importanța tehnologiilor informaționale în educație [75].

Ю.В. Позняк, А.С. Гаркун și А.А. Царева [76] susțin că avantajul utilizării SMÎ Moodle în educație constă în păstrarea abordării individuale, în posibilitatea de a organiza activități în grup.

Impactul utilizării instrumentelor TIC în predarea cursurilor de informatică constituie obiectul de studiu al mai multor cercetări din Republica Moldova, cea mai mare parte axându-se pe folosirea portofoliilor electronice și evaluarea studenților [77–80] prin intermediul diferitor softuri educaționale, mai puțin pe SMÎ. Astfel în [77] se menționează faptul că „implementarea modelului de integrare pentru crearea sistemului de situații didactice în cadrul mediilor digitale de creare a portofoliilor electronice duce la creșterea calității și eficacității în procesul de predare-învățare-evaluare, contribuind la formarea competențelor profesionale ale studenților, la creșterea performanțelor și la obținerea rezultatelor de succes” [77, p. 105].

În [78] este menționat că deprinderile de lucru independent au o importanță deosebită la studierea disciplinelor informatice a căror caracteristică definitorie este utilizarea calculatorului, iar interacțiunea cu calculatorul solicită și dezvoltă gândirea logico-algoritmică și sistemico-combinatorie.

După N. Deinego, „evaluarea resurselor necesare pentru demonstrarea competenței poate fi realizată prin administrarea testelor computerizate, iar evaluarea competenței are loc prin efectuarea proiectelor, la realizarea cărora studentul va articula și va mobiliza resursele necesare pentru soluționarea unei situații concrete” [80, p. 144]; eficiența portofoliului de predare este determinată, în cea mai mare măsură, de organizarea comunicării între tutore și cursanți (grupuri de comunicare, e-mail, chat).

Caracterizând tipurile de portofolii, S. Corlat [81] menționează că SMÎ încorporează instrumentele necesare pentru crearea metaportofoliului: forum, mesagerie internă pentru comunicare online; instrumente administrative de creare a structurilor de grup și de colectare a feedbackului; organizarea structurii arborescente în baza blogurilor și a profilurilor personale; controlul activității la curs, atrăgând atenția că „instrumentele TIC pot îmbunătăți calitatea învățării colaborative prin utilizarea aplicațiilor de tip web 2.0 – rețele sociale și specializate, care combină în sine posibilitățile de comunicare prin e-mail, chat, grupuri de discuții, bloguri etc.” [82, p. 60].

S. Gâncu a examinat dezvoltarea competenței de programare orientată pe obiecte din perspectiva instruirii în medii digitale, concluzionând că noile tehnologii impun noi metodologii de instruire. Într-un context informațional, condiționat de tehnologii din ce în ce mai performante, instruirea în învățământul universitar nu mai poate rămâne izolată de comunitățile virtuale. Concomitent, autorul subliniază că, din punct de vedere educațional, un e-portofoliu „poate fi privit ca o înregistrare a activității de învățare, care poate furniza dovezi asupra progreselor făcute”, precum și ca „un mijloc de gestiune ce privilegiază principiul potrivit căruia studentul este în centrul învățării, permițându-i profesorului să urmărească progresul acestuia în

dezvoltarea competențelor de programare orientată pe obiecte, modul de gândire, dar și autoevaluarea critică a propriei învățări” [83, p. 165, 166].

A. Braicov examinează câteva produse-soft ce reprezintă instrumente de elaborare a unor exerciții și probleme interactive, integrabile în SMÎ Moodle [84].

Implementarea TIC în procesul didactic a constituit obiectul de studiu și al cercetătorilor A. Gremalschi [85], L. Chiriac [86], V. Cabac [87], L. Mihălache [88], A. Globa [89].

Învățarea în SMÎ este foarte individualizată, profesorul având posibilitatea de a lucra în mod direct cu fiecare student, utilizând chaturile, ceea ce nu poate fi realizat întotdeauna în sala de studiu. Cu toate acestea, o mare atenție se acordă lucrului colectiv, cu ajutorul forumurilor și al altor componente de comunicare ale Moodle. Prin urmare, una din noile tendințe constă anume în a utiliza instrumente web online pentru predare, adaptare și integrare în SMÎ. Printre disciplinele în cadrul cărora au fost aplicate cu succes SMÎ enumerăm: ingineria sistemelor e-Learning, dezvoltarea bazelor de date, limbaje de programare, rețelele, procesarea imaginilor, analiza și proiectarea sistemelor, securitate informațională, tehnologii informaționale.

În pofida succeselor obținute, unii cercetători evocă mai multe probleme ce apar la predarea cursurilor de informatică online. Astfel, R. Fojtík și H. Habiballa se referă la: necesitatea creării unor materiale de calitate pentru cursurile online în SMÎ; necesitatea unei conexiuni de mare viteză la Internet; necesitatea elaborării unor suporturi didactice originale pentru predarea teoriei limbajelor formale și automate, a logicii etc. [90].

Cercetând cu atenție literatura de specialitate, am putea să deducem următoarele:

- Analiza literaturii de specialitate în domeniul utilizării SMÎ în procesul educațional la informatică a evidențiat o serie de probleme legate de: îmbunătățirea SMÎ-urilor cu module software noi; optimizarea cursurilor de informatică; utilizarea portofoliilor electronice; folosirea evaluărilor automatizate; interacțiunea dintre instructor și cursant; formarea deprinderilor de lucru independent; formarea competențelor profesionale ale studenților; dezvoltarea competenței de programare orientată pe obiecte.

- S-a constatat că folosirea SMÎ în procesul educațional la informatică aduce următoarele beneficii: perfecționarea abilităților de lucru individual și în grup; aprofundarea cunoștințelor în domeniul informaticii; dobândirea deprinderii de autocontrol și de planificare a timpului personal; creșterea calității și cantității cunoștințelor; îmbunătățirea competențelor în domeniul tehnologiilor informaționale.

- Nu au fost examinate aspecte legate de: impactul SMÎ asupra procesului de predare-învățare-evaluare a programării orientate pe obiecte; metodele de utilizare a instrumentelor SMÎ în predarea programării orientate pe obiecte, în funcție de necesitățile, interesele și capacitățile

studentilor; procesul de formare a competențelor (și abilităților) de programare orientată pe obiecte prin prisma SMÎ.

- Rămân la ordinea zilei următoarele probleme de cercetare:
 - Pentru ce discipline este mai indicată educația prin intermediul SMÎ?
 - Din punctul de vedere al formei de instruire, pentru ce categorie de studenți este mai potrivită educația la distanță?
 - Cum ar trebui abordat învățământul la distanță: ca o modalitate de alternativă sau complementară de a face educație?
- În urma investigațiilor la care ne-am referit mai sus, putem formula următoarele ipoteze:
- ✓ Grație utilizării mijloacelor puse la dispoziție de TIC, în general, și de SMÎ, în special, sporește eficiența procesului de învățare.
 - ✓ Utilizarea SMÎ contribuie la optimizarea procesului de formare a competențelor de programare în cadrul studierii specialităților TIC.

1.4. Concluzii la capitolul 1

Tehnologiile noi oferă instrumente care pot fi folosite pentru îmbunătățirea procesului de predare-învățare, iar SMÎ permit furnizarea diferitor tipuri de materiale, precum și interacțiunea în timp real. Astfel, tehnologiile informaționale poziționează procesul didactic într-o lumină nouă și schimbă perspectiva asupra întregului proces instructiv-educativ. Implementarea SMÎ în procesul de instruire, datorită funcțiilor puternice SMÎ, permite planificarea mult mai eficientă a cursurilor, centrarea resurselor de învățământ pe student și pe nivelul de cunoștințe al acestuia, accesul în timp real la informație de oriunde și oricând, precum și conectarea cu o comunitate educațională globală.

Analiza cercetărilor la nivel internațional în domeniul predării-învățării-evaluării tehnologiei Programării Orientate pe Obiecte demonstrează importanța acestora în pregătirea tinerelor generații de informaticieni profesioniști.

Complexitatea paradigmei Programării Orientate pe Obiecte, care stă la baza majorității sistemelor de operare și mediilor de dezvoltare a produselor-program, dificultățile ce apar în procesul de predare a POO și formarea competențelor de POO necesită abordări educaționale și metode de predare-învățare-evaluare noi, iar pe alocuri radicale. Astfel, este important de a identifica strategiile, mijloacele și metodele de instruire eficiente în domeniul POO.

Având în vedere cele menționate, considerăm actuală următoarea **problemă de cercetare** care constă în fundamentarea teoretico-praxiologică a eficientizării procesului de studiere a

Programării Orientate pe Obiecte, prin elaborarea unui model pedagogic de predare-învățare-evaluare în baza cursului universitar *Programarea Orientată pe Obiecte*, axat pe utilizarea Sistemelor de Management al Învățării și orientat spre formarea competenței de Programare Orientată pe Obiecte la viitorii specialiști în informatică (inclusiv în domeniul *Științe ale Educației*).

Pentru rezolvarea problemei de cercetare:

- a) au fost determinate avantajele oferite de Sistemele de Management al Învățării și argumentarea necesității implementării lor în procesul didactic universitar. Cele mai importante dintre ele se referă la: promovarea învățării active, a interacțiunii, a gândirii critice; creșterea motivației studenților; oferirea instantă a feedback-ului; furnizarea conținuturilor de învățare în diferite formate (multimedia, video, text); extinderea timpului și a locației de acces; individualizarea conținuturilor și traseelor de învățare; reutilizarea activităților de învățare; diversificarea metodelor și tehnologiilor de predare-învățare.
- b) s-a precizat Sistemul de Management al Învățării cu caracteristici optime pentru sistemul educațional universitar. Astfel, s-a stabilit că Sistemul de Management al Învățării (open source) cu caracteristici optime pentru sistemul educațional universitar este SMÎ Moodle.
- c) a fost identificat limbajul orientat pe obiecte adecvat studierii conceptului de Programare Orientată pe Obiecte. Astfel, în urma analizei comparative a limbajelor orientate pe obiecte, s-a constatat că limbajul C++ este cel mai potrivit pentru studierea conceptului de Programare Orientată pe Obiecte.

1. Pentru rezolvarea completă a problemei de cercetare este necesar de a realiza următoarele obiective:

- a) De elaborat un model pedagogic centrat pe utilizarea Sistemelor de Management al Învățării în procesul de studiu al Programării Orientate pe Obiecte.
- b) De argumentat științifico-metodologic eficiența utilizării modelului pedagogic în studierea Programării Orientate pe Obiecte.
- c) De elaborat curriculumul la cursul universitar Programarea Orientată pe Obiecte, adaptat la modelul pedagogic creat.
- d) De validat prin experiment didactic eficiența modelului pedagogic elaborat.

2. BAZELE METODOLOGICE DE FORMARE A COMPETENȚEI DE PROGRAMARE ORIENTATĂ PE OBIECTE PRIN UTILIZAREA SISTEMELOR DE MANAGEMENT AL ÎNVĂȚĂRII

2.1. Elaborarea modelului pedagogic de predare-învățare-evaluare a cursului *Programarea Orientată pe Obiecte prin utilizarea SMÎ*

Premisele elaborării modelului pedagogic

Programarea este domeniul de bază al Informaticii. Din punct de vedere istoric, până în anii '70 ai secolului al XX-lea a dominat programarea *imperativă* (altfel zis, *procedurală*). Aceasta era bazată pe ideile lui Von Newmann și starea programului varia în funcție de timp. Abstractizarea algoritmului presupunea alegerea celor mai conforme proceduri. Acest concept de programare a dus la crearea bibliotecilor mari de proceduri. Exemple de limbaje de programare procedurală sunt Fortran, Cobol, Algol.

Programarea *modulară* (sau *structurată*) presupune punerea în valoare a modalității de organizare a datelor. Astfel, au apărut modulele (unitățile de program) în care puteau fi stocate nu doar subprograme (funcții, proceduri), dar și structuri de date. Exemple de limbaje de programare modulară sunt Pascal, C.

Programarea *funcțională* (anii 1970–1990) este declarativă, bazată pe matematică, în special pe teoria funcțiilor. Exemple de limbaje de programare funcțională sunt Lisp, Haskell, ML.

Programarea *logică*, de asemenea, este declarativă, bazată pe demonstrații automate (inteligență artificială). Exemple de limbaje de programare logică ar fi Prolog, Lisp.

Programarea Orientată pe Obiecte (POO) a apărut la începutul anilor '90 și este considerată revoluționară, deoarece este bazată pe teoria conceptelor și pe interacțiunea umană cu mediul. Acest concept de programare constituie o paradigmă, deoarece a fost acceptat pe larg, oferind comunității programatorilor, pe o perioadă îndelungată, o bază pentru dezvoltarea și a altor concepte de programare adaptate anumitor tipuri de produse-soft (de exemplu, programarea orientată pe aplicații).

Paradigma Programării Orientate pe Obiecte a fost implementată în majoritatea limbajelor de programare de nivel înalt: C++, C#, Object Pascal, Perl, Java, Python, PHP etc. Eficiența POO a fost demonstrată inițial îndeosebi prin programele C++, aici fiind utilizată în toată detalierea ei.

Tabelul 2.1. Evoluția conceptelor de programare

Perioada	1960–1970	1970–1990			Din 1990
Conceptul de programare	Imperativă / Procedurală	Modulară / Structurată	Funcțională	Logică	Orientată pe obiecte

Cursul *Programarea Orientată pe Obiecte* este unul fundamental, fiind parte a curriculumului de formare profesională inițială a specialiștilor în domeniul Informaticii.

Mai multe studii [91–95] au arătat că studenții se confruntă cu diferite dificultăți de învățare a POO. Aceste dificultăți se referă la:

- ✓ înțelegerea conceptelor orientate pe obiecte (obiect, clasă, proprietate, metodă, moștenire, polimorfism etc.) și a relației dintre aceste concepte;
- ✓ percepția principiilor orientate pe obiect (OO), cum ar fi încapsularea și reutilizarea, [92] și a aspectelor dinamice ale programelor OO [93];
- ✓ scrierea algoritmilor organizați în cheia POO etc. [94].

Studenții care au fost deja învățați cu paradigma procedurală de programare se confruntă cu probleme suplimentare de învățare, tipice ”schimbării paradigmei”, adică parvenite din cauza trecerii la o paradigmă nouă – cea a POO [95].

Luând în considerare aceste dificultăți, sunt căutate variate modalități eficiente de predare a cursului POO [96]. Chiar și în etapa programării procedurale existau recomandări de utilizare a instrumentelor educaționale menite să ajute studenții să depășească impedimentele și să realizeze cât mai bine obiectivele de învățare [97, 98].

Bineînțeles, în etapa programării obiectuale, ținând cont de dificultățile menționate, aplicarea instrumentelor și tehnologiilor noi este recomandată îndeosebi. Se caută metode și modele de instruire noi, care ar duce la facilitarea învățării și aprofundarea cunoștințelor de bază POO [99]. Astfel, tehnologiile noi oferă instrumente ce pot fi folosite pentru îmbunătățirea procesului de predare-învățare-evaluare, iar tehnologiile SMÎ permit furnizarea diferitor tipuri de materiale, precum și interacțiunea în timp real a studentului cu profesorul și colegii.

Conținutul unui curs online poate fi reprezentat în diverse moduri: text, grafică, audio, animație, video, dar această varietate de la sine nu facilitează învățarea. Proiectarea unui curs trebuie să implice mai mulți factori. Astfel, cercetătorul J. Bennedsen [100] evidențiază șase parametri care influențează rezultatul învățării:

1. *Precondițiile (prerecuzitele) de învățare* – cunoștințele, experiențele, abilitățile pe care le posedă deja studenții atunci când vin la prima lecție a cursului.

2. *Condiții externe* – condițiile ce limitează sau fac posibilă învățarea, cum ar fi echipamente, artefacte, timp, loc, resurse ale profesorilor, resursele de învățare etc.

3. *Obiectivele activității de învățare* – afirmații care descriu ce trebuie să învețe studenții în cadrul cursului din punctul de vedere al cunoștințelor, abilităților și competențelor.

4. *Conținutul* – colecția de resurse informaționale oferite studenților pentru parcurgerea cursului.

5. *Procesul de învățare* – procesul de organizare a conținuturilor, a scenariilor academice, dar și de modificare a subiectelor de studiu (implică reflecții asupra modalității de îmbunătățire a acestui proces).

6. *Evaluarea* – examinarea sistemică a valorii/caracteristicilor învățării (în funcție de procesul de predare, de obiectivele cursului și de cunoștințele studenților).

Activitățile de învățare din cadrul SMÎ includ tutoriale online, exerciții, chestionare, forumuri, chaturi etc. De fapt, un SMÎ execută legătura dintre instructori și cursanți, iar beneficiile cognitive ale utilizării acestor sisteme se exprimă prin faptul că ele le permit studenților și instructorilor să se întâlnească în sălile virtuale de clasă. Mai mult decât atât, SMÎ le permit profesorilor să observe cu mare atenție abilitățile de învățare și ratele de succes ale studenților. La rândul lor, studenții beneficiază de oportunitățile SMÎ prin:

- studierea cursurilor în propriile locuri de studiu;
- întâlnirea instructorilor și colegilor în sălile virtuale de clasă (deci fără restricții de locație și timp);
- pregătirea pentru examene prin rezolvarea testelor de autocontrol (ori de câte ori au necesitate).

▪ Un avantaj important oferit de educația online este *învățarea prin colaborare*. Asigurarea unei metodologii diversificate, fundamentate pe îmbinarea activităților de învățare și a lucrului independent cu activitățile de colaborare și de învățare în grup este un deziderat primordial al educației progresiviste. După cum menționează cercetătorul român I. Cerghit, “învățarea în grup exersează capacitatea de decizie și de inițiativă, dă o notă mai personală muncii, dar și o complementaritate mai mare aptitudinilor și talentelor, ceea ce asigură o participare mai vie, mai activă, susținută de foarte multe elemente de emulație, de stimulare reciprocă, de cooperare fructuoasă” [101, p. 54]. Hubscher-Younger și Narayanan [102] menționează că, din punct de vedere metodologic, utilizarea *grupului* în învățarea colaborativă posedă un șir de avantaje:

- ✓ Apar idei mai multe și mai bune.
- ✓ Instruiții învață unul de la altul.

- ✓ Încurajează contribuția personală.
- ✓ Se valorifică informațiile colegilor în construirea cunoașterii proprii.
- ✓ Stimulează efortul și productivitatea individului.
- ✓ Se dezvoltă relații interpersonale mai bune.

▪ Alt avantaj al instruirii online este faptul că în calculator (prin logarea fiecărui utilizator) se păstrează o evidență detaliată a interacțiunii studenților cu materialele de referință pe care le utilizează.

În consecință, creșterea recentă a comunicării prin mesaje instantanee în rândul studenților a deschis posibilitatea dezvoltării unor sisteme de învățare în grup de la egal la egal, care ar putea fi bine recepționate de actuala generație de studenți.

Metodele de învățare folosite la predarea cursurilor pentru studenții din domeniul informaticii au anumite caracteristici. Instrumentele de comunicare bazate pe Internet, cum ar fi chatul, forumul, e-mailul, sunt necesare, dar nu sunt suficiente pentru a interacționa cu profesorul sau colegii. În cazul predării informaticii, instrumentele de colaborare trebuie să fie compatibile cu logica și metodele de predare a disciplinelor informatice. Un instrument de colaborare compatibil trebuie să aibă un mediu grafic prietenos, prin care atât profesorul, cât și studenții să poată interacționa simultan cu instrumentele colaborative de învățare.

Necesitatea implementării instrumentelor TIC pentru îmbunătățirea procesului didactic, a metodelor interactive, aplicate la predarea cursului *POO*, și necesitatea îmbinării activităților de învățare cu activitățile de colaborare ne-a condus la ideea elaborării unui model pedagogic care ar reflecta aceste concepte.

Modelarea pedagogică în contextul e-Learningului

În pedagogie, modelul este un instrument important de cercetare științifică. Modalitatea de cunoaștere a realității prin reprezentarea fenomenului studiat cu ajutorul unui sistem construit artificial definește noțiunea de *model*.

Cercetătorii I. Lupu, V. Cabac și S. Gîncu menționează că modelarea pedagogică implică creativitate și se realizează după schema din figura 2.1 [103].

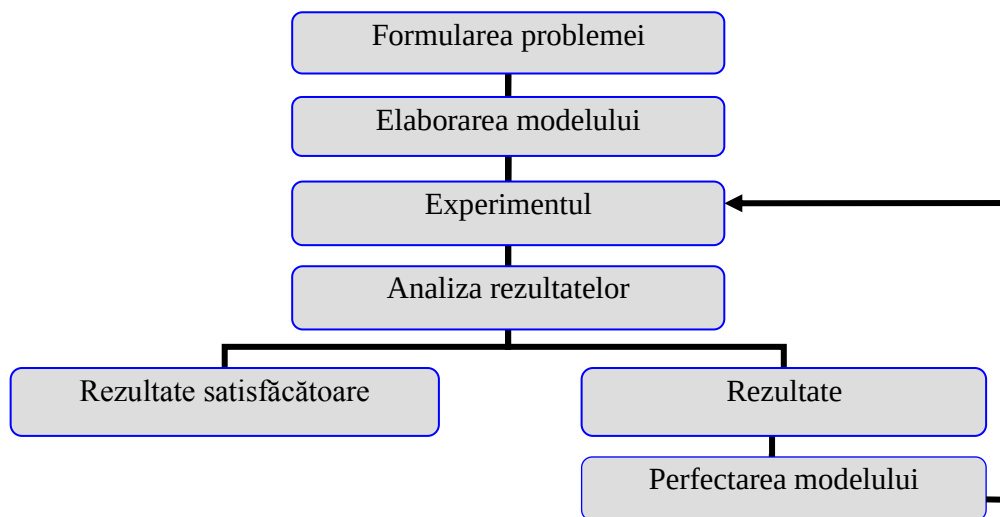


Fig. 2.1. Etapele modelării pedagogice

Cercetătorul C. Bîrzea definește modelul drept ”o reprezentare analogică a unor structuri prin altele, reduse sau schematizate”, această construcție figurativă facilitând cunoașterea și acțiunea [104, p. 107].

Savantul А.Н. Дахин consideră că „*modelarea pedagogică* reprezintă o metodă de cercetare orientată într-o anumită direcție, ce reflectă caracteristicile fenomenelor cercetate ca formă și conținut” [105, p. 17].

Modelul este un sistem care reproduce, la un alt nivel, structura unui alt sistem pe care vrem să-l cunoaștem. Prin intermediul modelului este redat sistemul original, obiect, fenomen sau proces real. Caracteristica modelului este posibilitatea de a stabili relații analogice cu obiectul sau cu procesul pe care-l modelează. Astfel, modelul didactic îndeplinește o funcție demonstrativă, înglobând informații care urmează să fie redescoperite. Această funcție se manifestă pe două planuri: unul în care modelul constituie punct de plecare sau izvor de informații, ce urmează să fie descoperite sau observate, iar celălalt în care modelul confirmă cunoștințele transmise pe alte căi.

Se pot distinge următoarele tipuri de modele didactice, după specificul lor [106, p. 17]:

- *Modele didactice obiectuale.* În aceste modele sunt încadrate toate acele modele ce reproduc în micro, sub formă materială, diferite obiecte, fenomene și procese din realitate. Originalul este reprodus cu fidelitate, atât ca formă externă, cât și ca structură internă.

- *Modele didactice figurative.* Acestea sunt modele ce reproduc obiectul, fenomenul sau procesul original prin intermediul imaginii, arătând relațiile funcționale dintre obiectele și fenomenele realității, concretizând o idee, teorie, lege, teoremă, un principiu etc.

- *Modele didactice simbolice.* Aceste modele reproduc originalul cu ajutorul semnelor convenționale: reprezentare grafică simplificată sau tridimensională a unei idei, a unui proces sau sistem.

Indiferent de natura lor, modelele didactice sunt considerate a fi analoage realității, și nu copii ale ei, reproducând structura obiectelor, fenomenelor și proceselor.

R. Iucu [107, p. 53] prezintă următoarele caracteristici ale modelelor didactice:

- ✓ modelele didactice sunt un produs al activității științifice sau, mai precis, un rezultat al reflecției teoretice și al sintezelor practice;

- ✓ modelele didactice aspiră la reprezentarea unei realități didactice complexe sau a unui aspect concret al acesteia;

- ✓ modelul didactic poate fi asociat cu ideea de simulare, în condițiile în care reprezentarea realității prin el este una aproximativă;

- ✓ modelele sunt temporale și temporare, beneficiind de o doză de prevedere, derivată din aspectul finalist, care le conferă un accent proiectiv și orientativ;

- ✓ modelele sunt adaptabile, putând suferi revizuiți și ajustări după analiza realității didactice pe care o exprimă;

- ✓ modelele sunt ipotetice, ceea ce înseamnă că trebuie supuse unui proces de verificare, trăsătură ce le conferă dinamism;

- ✓ modelele sunt optimizabile, putând evolua în funcție de investiția de tip analitico-investigativ;

- ✓ modelele au rol de orientare și de organizare a activităților didactice;

- ✓ modelele didactice sunt evaluabile, informațiile inițiale putând fi confirmate sau infirmate, total sau parțial, în urma aplicării lor.

Modelele pedagogice reflectă, de obicei, o abordare pedagogică sau teoriile de învățare. În condițiile actuale, când e-Learningul ia o amploare tot mai mare, au fost realizate o serie de revizuiți ale teoriilor de învățare și ale relevanței lor pentru e-Learning [108].

În continuare vom analiza noțiunea de model pedagogic din perspectiva importanței pentru e-Learning.

Cercetătorii T. Mayes și S. Freitas [109] au grupat teoriile de învățare în trei categorii:

- *asociativă* (învățarea ca activitate prin sarcini structurate);

- *cognitivă* (învățare prin înțelegere);
- *situatională* (învățarea ca practică socială).

Perspectiva asociativă se concentrează asupra modificării comportamentului prin intermediul perechilor stimulator-răspuns, învățare prin încercare și erori, învățare prin asociere și întărirea rezultatelor învățării și a rezultatelor observabile, dând naștere unor teorii behavioriste. Modelul instructiv bazat pe teoria învățării a lui Gagné, proiectat pentru formarea cunoștințelor și abilităților printr-un șir de pași, este cea mai recentă abordare teoretică, ce face parte din perspectiva asociativă.

D.M. Merrill a revizuit teoriile și modelele de design instructiv, propunând un șir de principii pentru instruirea prescriptivă [110]:

- *Principiul demonstrației* – învățarea este promovată atunci când studenții examinează o demonstrație.
- *Principiul aplicării* – învățarea este promovată atunci când studenții aplică noile cunoștințe.
- *Principiul bazat pe sarcini* – învățarea este promovată atunci când studenții se angajează într-o strategie de instruire centrată pe sarcini.
- *Principiul activării* – învățarea este promovată atunci când studenții activează cunoștințele sau experiențele anterioare relevante.
- *Principiul integrării* – învățarea este promovată atunci când studenții integrează noile cunoștințe în viața de zi cu zi.

Analizând aceste cinci principii ale modelului lui D. Merrill, am putea susține că cele mai eficiente medii de învățare sunt cele bazate pe probleme, în care studenții sunt implicați în patru etape distincte (figura 2.2):

- activarea cunoștințelor anterioare;
- demonstrarea competențelor;
- aplicarea competențelor;
- integrarea activităților în lumea reală.

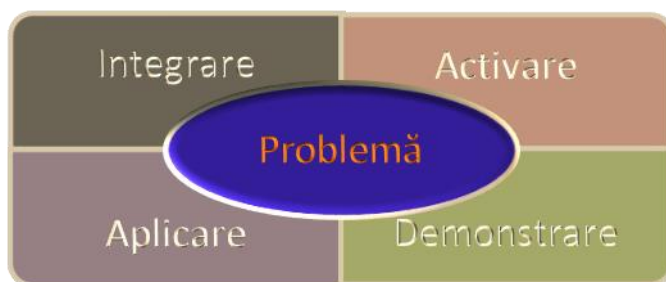


Fig. 2.2. Etape în proiectarea instruirii eficiente (după D. Merrill [110])

Perspectiva cognitivă privește învățarea ca transformare în structurile cognitive interne. Din punct de vedere pedagogic, se caracterizează prin procesarea și transmiterea informațiilor prin comunicare, explicare, recombinație, contrast, inferență și rezolvarea problemelor. Ea dă naștere unor atitudini constructiviste și empiric-reflexive.

Un mecanism de promovare a unui mediu constructiv, care a fost adoptat pe scară largă în crearea mediilor de învățare electronice, este *schema cognitivă* (reprezentării mentale abstracte ce rezumă și organizează în mod structurat evenimente, obiecte, situații sau experiențe asemănătoare [111]), în care activitățile în care se implică cursantul sunt susținute de o serie de principii, pentru a-l sprijini și a-l ajuta să reflecteze asupra acțiunilor sale.

Multe medii electronice de instruire oferă diferite forme de suport cognitiv, care ghidează acțiunile cursanților și promovează reflecția.

Un exemplu care promovează constructivismul este modelul dezvoltat de D. Jonassen și colegii săi [112]. Acest model poate fi utilizat ca îndrumar pentru dezvoltarea mediilor de învățare constructiviste. Argumentul-cheie este că învățarea apare atunci când cursanții se angajează activ în crearea conceptului. Modelul cuprinde cinci părți:

- *Activă și manipulativă*: învățarea are loc atunci când cursanții dezvoltă cunoștințe și abilități ca răspuns la antrenament, manipulând obiecte și învățând din rezultate.
- *Constructivă și reflexivă*: învățarea apare pe măsură ce cursanții reflectă asupra activității și observațiilor și exprimă ceea ce au învățat.
- *Intențională*: învățarea apare atunci când cursanții sunt motivați să atingă un scop cognitiv.
- *Autentică* (complexă și contextualizată): învățarea este situată într-un context semnificativ, în loc să fie simplificată și prezentată în mod izolat.
- *Cooperativă* (colaborativă/conversațională): învățarea se bazează pe înțelegeri care îi ajută pe cursanți să învețe din propria lor cunoaștere și din cunoștințele celorlalți, pentru a construi noi cunoștințe.

Perspectiva situațională privește învățarea ca participare socială și subliniază relațiile interpersonale, care implică imitarea, modelarea și construirea în comun a cunoștințelor. Din punctul de vedere al perspectivei situaționale, obiectivul final al învățării este de a trata lumea ca fiind semnificativă.

Un model specific de e-Learning, care descrie etapele de sporire a competenței din punctul de vedere al perspectivei situaționale, a fost propus de către Salmon pentru sprijinirea

eficientă a e-moderării în forumurile de discuții [113]. Acest model este alcătuit din următoarele cinci etape:

- Acces și motivație.
- Socializare online.
- Schimb de informații.
- Construire de cunoștințe.
- Dezvoltare.

Pe lângă modelele ce corespund uneia dintre cele trei perspective teoretice, există modele care oferă o imagine generală. Un astfel de model este și Suportul Open Learning (SOL) pentru Open University.

Stilul de predare Open University mai este denumit și ”învățare open learning”, cunoscut și sub denumirea de ”învățământ la distanță” [114]. Tipul de învățare open learning presupune că studenții învață în timpul lor liber, citind materiale de curs, lucrând la activități de curs, scriind sarcini și, eventual, lucrând cu alți cursanți. Suport înseamnă asistență din partea unui tutore. Modelul SOL este descris ca fiind bazat pe trei factori:

- *Învățare la distanță* sau ”open learning” (învățarea individuală: citire, activități și sarcini, lucrul cu alți cursanți).
- *Resurse* (texte de curs, DVD-uri, experimente acasă, materiale interactive, materiale web).
- *Asistență sistematică* (prin intermediul unui tutore, suport IT, plus tutoriale și suport online).

În anii '50 ai secolului XX apare conceptul de *design instrucțional* (DI). Funcția DI constă în elaborarea, dezvoltarea și punerea în practică a celor mai eficiente metode și strategii de învățare pentru atingerea obiectivelor propuse [115]. În baza conceptului DI, în 1975, Centrul de Tehnologie Educațională de la Universitatea de Stat din Florida elaborează pentru armata SUA modelul ADDIE [116]. Mai târziu, acest model este implementat pe larg în diverse ramuri.

Modelul ADDIE (figura 2.3) constă din următoarele cinci etape:

1. Etapa *analiză* (*Analysis*), în cadrul căreia se stabilesc obiectivele cursului, destinatarii și competențele inițiale.
2. Etapa *proiectare* (*Design*) – este destinată precizării structurii și interfeței materialului, precum și mijloacelor utilizate pentru elaborarea lor.
3. Etapa *dezvoltare* (*Develop*) – fixează forma de legătură și dependență între materialul cursului și componentele lui: text, grafică, animație.

4. Etapa *implementare* (*Implement*) – este rezervată pentru plasarea materialului online, derularea cursului și asistența tehnică.

5. Etapa *evaluare* (*Evaluate*) – determină eficiența cursului, gradul de implicare a cursanților, cercetează feedbackul, adaptând și îmbunătățind materialul.

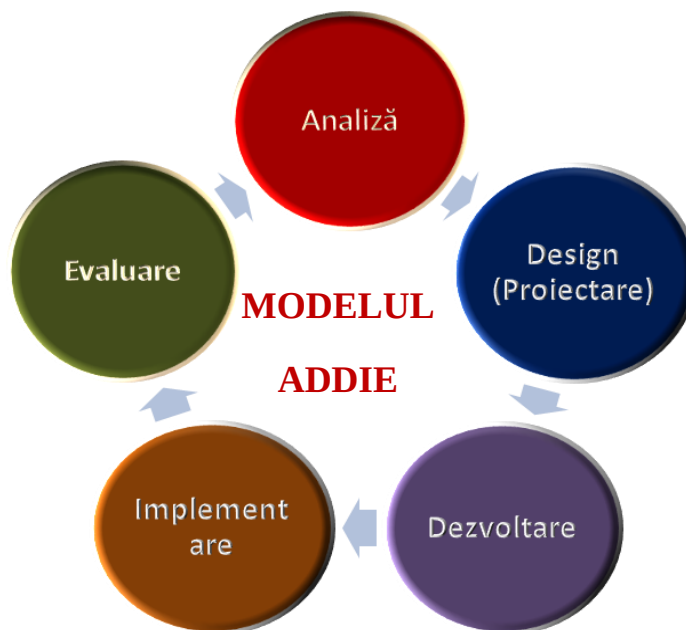


Fig. 2.3. Schema modelului ADDIE

Modelul ADDIE stă la baza proiectării și dezvoltării programelor educaționale și de formare online și este un proces care necesită o evaluare continuă și feedback. Această secvență nu impune o progresie liniară strictă. O astfel de abordare este considerată foarte utilă, deoarece definirea clară a etapelor facilitează implementarea unor instrumente de instruire eficiente.

Modelul ADDIE este criticat că ar fi prea liniar, inflexibil, necesitând un timp mare de implementare și având un feedback întârziat.

Ca alternative ale modelului ADDIE sunt propuse o serie de alte modele, printre care și Backward Design. Modelul *Backward Design* sau *planificarea Backward* este un proces în care instructorii proiectează experiențele de învățare și tehnicile de instruire pentru a atinge obiectivele propuse.

Modelul Backward [117] cere de la instructor să se concentreze inițial asupra obiectivelor finale și finalităților de curs. Mai apoi, profesorul planifică conținutul cursului și activitățile de învățare în funcție de ce trebuie “să știe” sau ce trebuie “să facă” instruiții, conform așteptărilor educaționale ale cursului. Spre deosebire de ADDIE, designul Backward pornește de la scopul unui curs, ce constă în însușirea rapidă a abilităților de către cei care învață (figura 2.4).

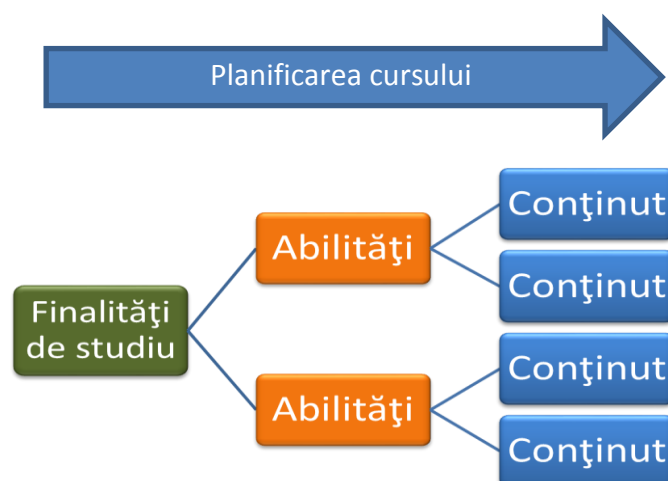


Fig. 2.4. Modelul Backward de planificare a unui curs

Pentru prima dată, modelul Backward a fost adus în discuție în 1949 de către Ralph Tyler. Ulterior, în anii 1998–1999, Jay McTighe și Grant Wiggins au introdus ideea acestui model în proiectarea curriculumului.

Logica proiectării Backward sugerează o planificare secvențială a curriculumului. Modelul Backward al cursului este împărțit în trei etape (figura 2.5):

1. Stabilirea obiectivelor de învățare ale cursului.
2. Identificarea abilităților care vor fi obținute prin realizarea obiectivelor de învățare.
3. Determinarea conținutului necesar de elaborat pentru a susține aceste abilități.



Fig. 2.5. Etapele modelului Backward

În etapa „Identificarea rezultatelor dorite” trebuie de ținut cont de așteptările rezultatului de învățare pentru curs. În această etapă e necesar de a pune următoarele întrebări:

- Ce ar trebui să știe și să poată face studenții până la sfârșitul cursului?
- Ce trebuie să învețe studenții pe parcursul cursului, pentru a atinge obiectivele finale ale acestuia?
- Ce priceperi durabile ar trebui să obțină studenții prin cursul respectiv?

- Care sunt ideile principale, conceptele și întrebările esențiale din cadrul cursului, pe care studenții le vor revizui, probabil, pe parcursul studiilor și chiar pe parcursul întregii vieți?

În etapa „Stabilirea dovezilor realizabile” trebuie definite dovezile necesare pentru a evalua cunoștințele, abilitățile și atitudinile dorite. În Backward design este necesar de planificat și de creat toate evaluările, cum ar fi scrierea de proiecte, chestionare și teste, înainte de a trece la planificarea și crearea activităților de învățare ale cursului. Evaluările proiectate cu grijă vor furniza dovezi că studenții au obținut rezultatele de învățare dorite și vor ajuta la planificarea conținutului lecțiilor și al activităților.

Selectarea strategiilor de predare, în cadrul modelului Backward, este efectuată în etapa „Planificarea procesului de învățare și instruire”. Cursul proiectat trebuie să fie pe cât de eficient, pe atât de captivant. Un design atrăgător stimulează participarea activă a studenților, în timp ce un design eficient include dovezi de obținere a rezultatelor dorite. În planificarea lecțiilor este necesar de ținut cont de activitățile ce vor fi realizate de studenți. Strategiile și activitățile de instruire trebuie alese astfel încât studenții să poată atinge obiectivele cursului.

În contextul e-Learningului, modelul Backward se extinde ca în figura 2.6.

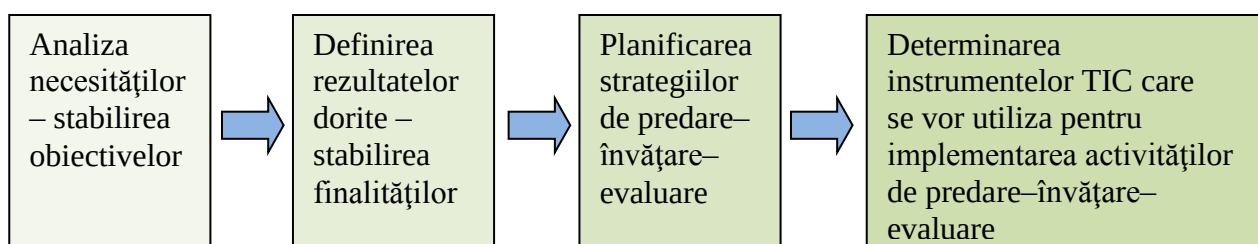


Fig. 2.6. Etapele modelului Backward în contextul e-Learningului

Caracteristicile modelului pedagogic al procesului de studiu al Programării Orientate pe Obiecte prin utilizarea SMÎ (modelul POO SMÎ)

După o analiză profundă a cerințelor impuse de societatea modernă și a practicilor de aplicare a TIC în scopul eficientizării și ajustării procesului de învățământ la imperativele timpului, a problemelor ce apar în procesul de studiere a principiilor POO, care stau la baza formării inițiale a specialiștilor în informatică (inclusiv în domeniul *Științe ale Educației*), a fost elaborat **modelul pedagogic de predare-învățare-evaluare al cursului universitar Programarea Orientată pe Obiecte prin utilizarea SMÎ (modelul POO SMÎ)**, (figura 2.7).

La elaborarea modelului POO SMÎ s-a ținut cont de principiile modelelor pedagogice care reflectă teoriile de învățare prin prisma e-Learning-ului și, în mod special, de principiile modelului de proiectare Backward.

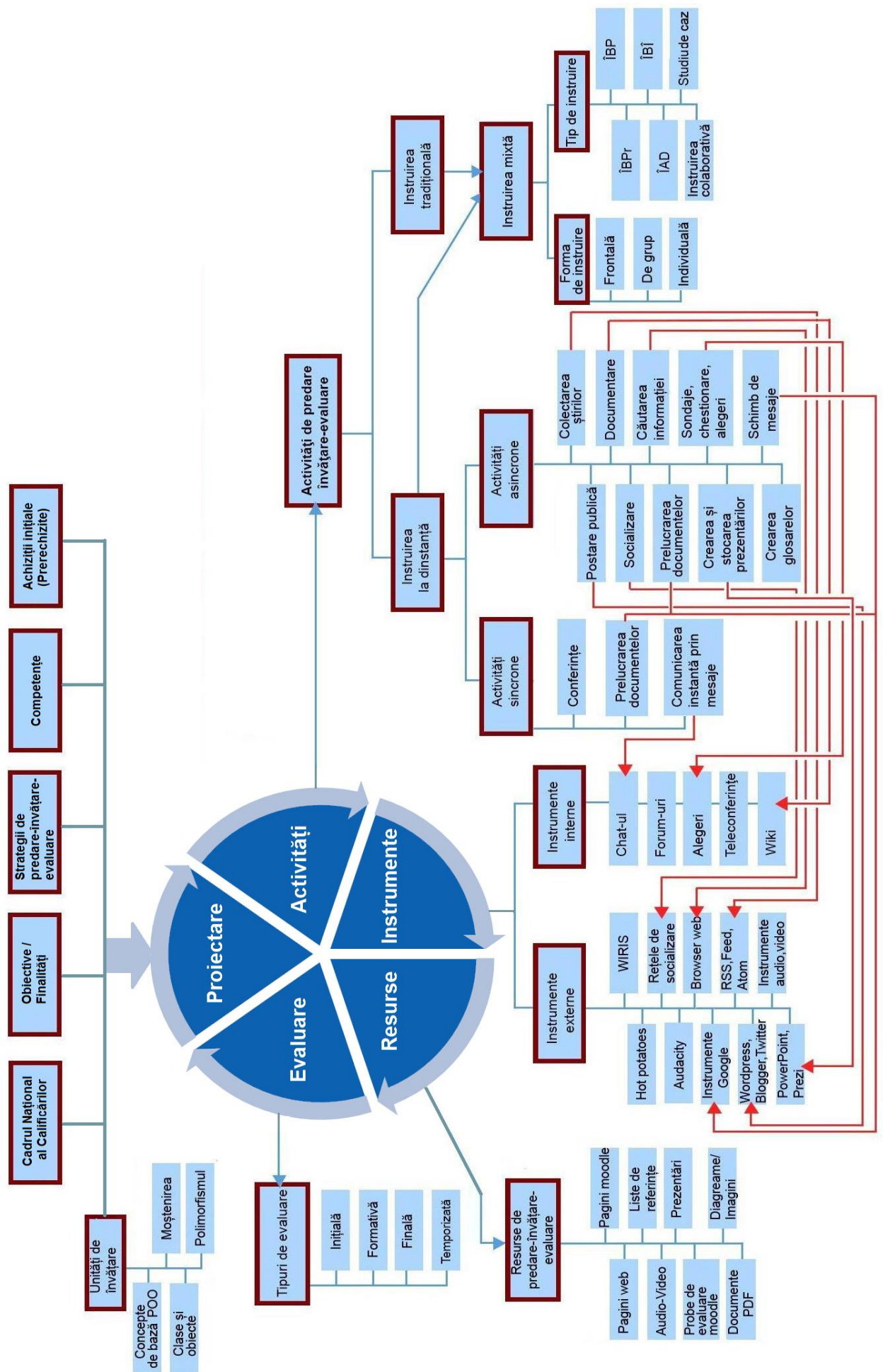


Fig. 2.7. Modelul POO SMÎ

Raportat la disciplina *Programarea Orientată pe Obiecte*, modelul elaborat posedă următoarele proprietăți:

- *Plurivalență*: pentru dezvoltarea competențelor specifice disciplinei *Programarea Orientată pe Obiecte*, cum ar fi, de exemplu, analiza unei probleme din perspectiva orientată pe obiecte, inițial este necesar de a dezvolta competența de descriere a principiilor POO, iar acest lucru este obținut, în modelul prezentat, prin abordarea complexă a modulelor didactice care sunt centrate pe nevoile și posibilitățile individuale ale studenților, adaptat intereselor cognitive, ritmului personal de lucru și stilului de învățare.

- *Continuitate*: valorificarea mecanismului de funcționare al modelului pedagogic propus asigură legătura dintre competențele inițiale, caracteristice cursului universitar *Programarea Orientată pe Obiecte*, ce se formează începând de la prerechizite, și competențele formate la finele studiilor. Competențele dobândite în cadrul acestui curs servesc drept prerechizite pentru alte cursuri, de exemplu, pentru *Programare Java*, *Limbaje și Sisteme Contemporane de Programare* (ciclul II, masterat) etc.

- *Colaborativitate*: muncind în comun la diferite module ale cursului *POO*, studenții își formează și își modelează cunoștințele, asistând și atestând dezvoltarea abilităților fiecărui membru și apreciind contribuția fiecăruia la munca în comun.

- *Contextualitate*: este asigurată de confruntarea cu sarcini din viața reală, care sunt folosite pe deplin în partea practică a cursului *Programarea Orientată pe Obiecte*, îmbogățind cunoștințele și abilitățile. Multitudinea variată de sarcini condiționează realizarea transferului cunoștințelor învățate.

- *Originalitate*: caracterul inedit al modelului pedagogic elaborat este asigurat de obiectivele propuse, de implementarea SMÎ în procesul de predare-învățare-evaluare a cursului universitar *Programarea Orientată pe Obiecte*, care dirijează studenții spre performanțe academice sporite.

Considerăm că modelul propus posedă un șir de avantaje, dintre care cele mai importante ar fi:

- *economicitatea* – poate fi transmis un volum mare de cunoștințe la un număr foarte mare de studenți într-un timp redus;

- *creativitatea* – dezvoltarea gândirii creative și a inteligenței studenților prin participare activă la crearea activităților de curs;

- *tehnologizarea* – sunt folosite o multitudine de procedee și de tehnici de lucru didactice.

2.2. Metodologia utilizării modelului elaborat

Obiective, strategii de predare-învățare-evaluare, competențe

Cerințele modernizării și perfecționării metodologiei didactice conduc la sporirea metodelor de învățământ cu caracter activ, la aplicarea unor metode cu caracter formativ, la utilizarea problematizării în toate tehnicile și metodele de predare-învățare, la valorificarea noilor tehnologii instrucționale, contribuindu-se, astfel, la dezvoltarea întregului potențial al instruitului. Învățământul modern începe să se axeze pe o metodologie care promovează metodele interactive, ce solicită mecanismele gândirii, inteligenței, imaginației și creativității.

Totodată, cercetătorii V. Cabac și N. Deinego [118] scot în evidență un șir de particularități legate de formarea în domeniul programării:

- necesitatea de aliniere continuă la schimbările extrem de rapide, care au loc în domeniul informaticii și tehnologiilor informaționale;
- elaborarea produselor-program moderne, care presupune lucrul în echipă;
- în condițiile organizării învățământului universitar pe cicluri, formarea specialistului presupune achiziționarea unui fundament solid de cunoștințe, care asigură dezvoltarea profesională și face posibilă extinderea educației pe tot parcursul vieții.

În același timp, sunt documentate o serie de dificultăți întâlnite în predarea și învățarea programării orientate pe obiecte. Unele dintre aceste dificultăți țin de înțelegerea și utilizarea celor mai esențiale caracteristici POO: moștenirea, abstractizarea, clasele, polimorfismul etc.

Așadar, ***obiectivele educaționale*** urmărite la aplicarea modelului POO SMÎ, racordate la necesitățile imperative ale timpului, pot fi formulate astfel:

- A valorifica la cel mai înalt nivel posibilitățile oferite de tehnologiile informaționale.
- A îmbunătăți calitatea educației în domeniul programării obiectuale.
- A dezvolta metodele și contextele de predare și învățare ale cursului universitar *POO*, necesare pentru asigurarea procesului de instruire.
- A asigura accesul continuu la resursele de învățare, pentru a forma studenților competențele necesare în domeniul POO.
- A oferi oportunități de învățare permanentă cât mai aproape posibil de utilizatori.
- A evalua impactul utilizării modelului elaborat asupra performanței studenților la disciplina *Programarea Orientată pe Obiecte*.

Sistemul de învățământ, influențat de societatea informațională bazată pe cunoaștere, încearcă introducerea mijloacelor tehnologice în procesul educațional prin modernizarea

materialelor didactice și a strategiilor folosite, precum și a felului în care se realizează procesele de predare-învățare-evaluare.

Conceptul de *strategie didactică* a fost definit de mai mulți cercetători. Astfel, savantul român, profesorul I. Cerghit, menționează că strategia didactică reprezintă „un ansamblu de acțiuni și operații de predare-învățare în mod deliberat, structurate sau programate, orientate în direcția atingerii, în condiții de maximă eficacitate, a obiectivelor prestabilite” [119].

După S. Cristea [120], strategia didactică este „un grup de două sau mai multe metode și procedee integrate într-o structură operațională, angajată la nivelul activității de predare-învățare-evaluare, pentru realizarea obiectivelor pedagogice generale, specifice și concrete ale acesteia, la parametri de calitate superioară”.

I. Nicola [121] definește strategia didactică ca fiind „un ansamblu de procedee prin care se realizează conlucrarea dintre profesor și studenți, în vederea predării și învățării unui volum de informații, a formării unor priceperi și deprinderi, a dezvoltării personalității umane”.

O generalizare a definițiilor precedente poate fi considerată definiția strategiei dată de Gaston Mialaret: strategia didactică este „știința sau arta de a combina și a coordona acțiunile în vederea atingerii unui scop. Ea corespunde unei planificări pentru a ajunge la un rezultat, propunând obiective de atins și mijloace vizate pentru a le atinge” [122].

Strategiile didactice pot fi clasificate după mai multe criterii. În figura 2.8 este prezentată una dintre aceste modalități.

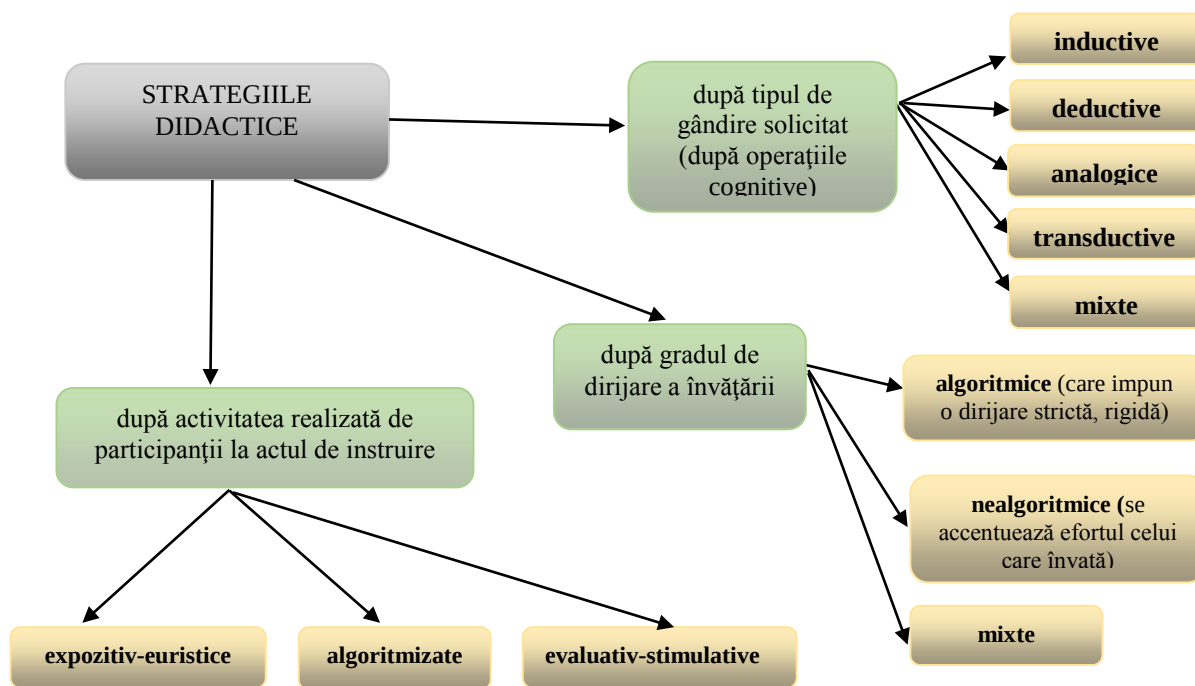


Fig. 2.8. Clasificarea strategiilor didactice ([123])

În contextul optimizării instruirii și implementării tehnologiilor informaționale se folosesc strategiile didactice interactive, care sunt valorificate pentru asigurarea eficienței procesului de învățământ. Strategiile didactice interactive susțin învățarea activă, unde cel instruit acționează asupra informației și o transformă în una nouă, proprie. Astfel, instruitul devine răspunzător și participant la procesul educativ prin explorări, cercetări, rezolvări de probleme, aplicând în diferite contexte ceea ce a dobândit în procesul de învățare.

Cercetătoarea C.-L. Oprea definește elementele de determinare a opțiunii pentru o anumită strategie didactică în felul următor (figura 2.9) [124]:

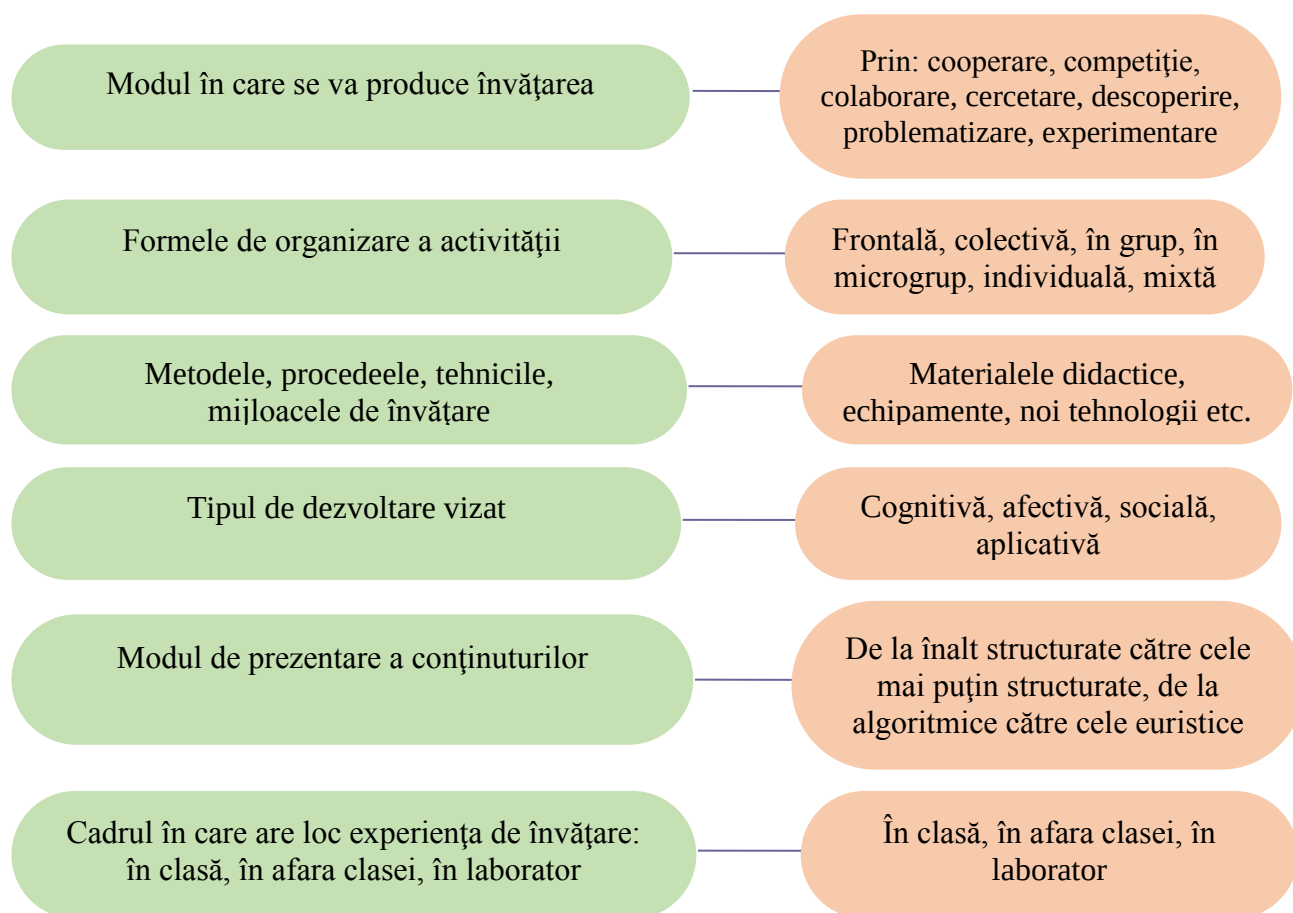


Fig. 2.9. Elementele pentru determinarea unei strategii

Strategiile didactice interactive dețin următoarele caracteristici [124, p. 26]:

- „sunt strategii de grup, în care studenții sunt organizați în grupuri mici de lucru și obiectivele preconizate sunt obținute grație muncii în colaborare;
- se prevede crearea de programe care să corespundă nevoii de interrelaționare și de răspuns diferențiat la reacțiile studenților;
- susțin învățarea activă, în cadrul căreia cel instruit acționează asupra informației, transformând-o în una nouă, personală;

- sunt dezvoltate procesele cognitive complexe prin participarea subiecților la acțiune, sunt dezvoltate capacitățile de înțelegere și de (auto)evaluare prin folosirea metodelor active”.

Strategiile didactice interactive promovează o învățare activă și presupun o colaborare între subiecții organizați în microgrupuri, care lucrează împreună pentru atingerea obiectivelor prestabilite. Rolul profesorului rămâne în continuare unul important, dar se renunță la practicile educaționale vechi, rigide și uniforme. Profesorul devine organizatorul unui mediu de învățare adaptat necesităților și specificului studenților, facilitând procesul învățării și formării competențelor. Studenții au ocazia de a se implica activ în procesul propriei formări, de a-și dezvolta competențele metacognitive, își pot exprima în mod liber ideile, opiniile.

Utilizarea strategiilor didactice de tip interactiv-participativ și metacognitive îl plasează pe student în centrul procesului instructiv-educativ, stimulându-l să se implice efectiv în activitățile teoretice și aplicative, punând accent pe spiritul de inițiativă, independență și potențialul creator. Strategiile ce conduc la competențe metacognitive și la o motivație intrinsecă creează premisele unei învățări conștiente, autonome, sistematice.

Strategiile metacognitive se referă atât la planificarea învățării, cât și la conștientizarea propriilor limite, prin evaluarea corectă a rezultatelor în raport cu obiectivele.

În procesul predării cursului *Programarea Orientată pe Obiecte*, strategiile didactice aplicate de către autor au fost gândite astfel încât să asigure formarea competențelor specifice POO prin alegerea diferitor combinații de metode, tehnici și principii de instruire, mijloace de învățare și forme de organizare a studenților. Aceste strategii sunt redată în figura 2.10.

Modelul didactic elaborat de autor se bazează pe principiile didactice care asigură orientarea generală a procesului de învățământ, oferă un sens funcțional activităților didactice și asigură condițiile de îndeplinire a obiectivelor scontate.

Principiul reprezintă elementul de continuitate prezent în diferite situații de predare-învățare, expunând modul de aplicare a legilor.

Funcțiile principiilor didactice în procesul de învățământ sunt [125]:

- orientarea traseului educativ înspre obiective;
- normarea practicii, precizând obligațiile ce trebuie respectate, criteriile, condițiile;
- prescrierea modurilor de relaționare, de acțiune în raport cu situația de învățare;
- coordonarea activității conform rezultatelor obținute.

Autorul propune aplicarea principiilor didactice în modelul elaborat, pentru asigurarea condițiilor de atingere a obiectivelor propuse, din perspectiva disciplinei *Programarea Orientată pe Obiecte* și a SMÎ Moodle.

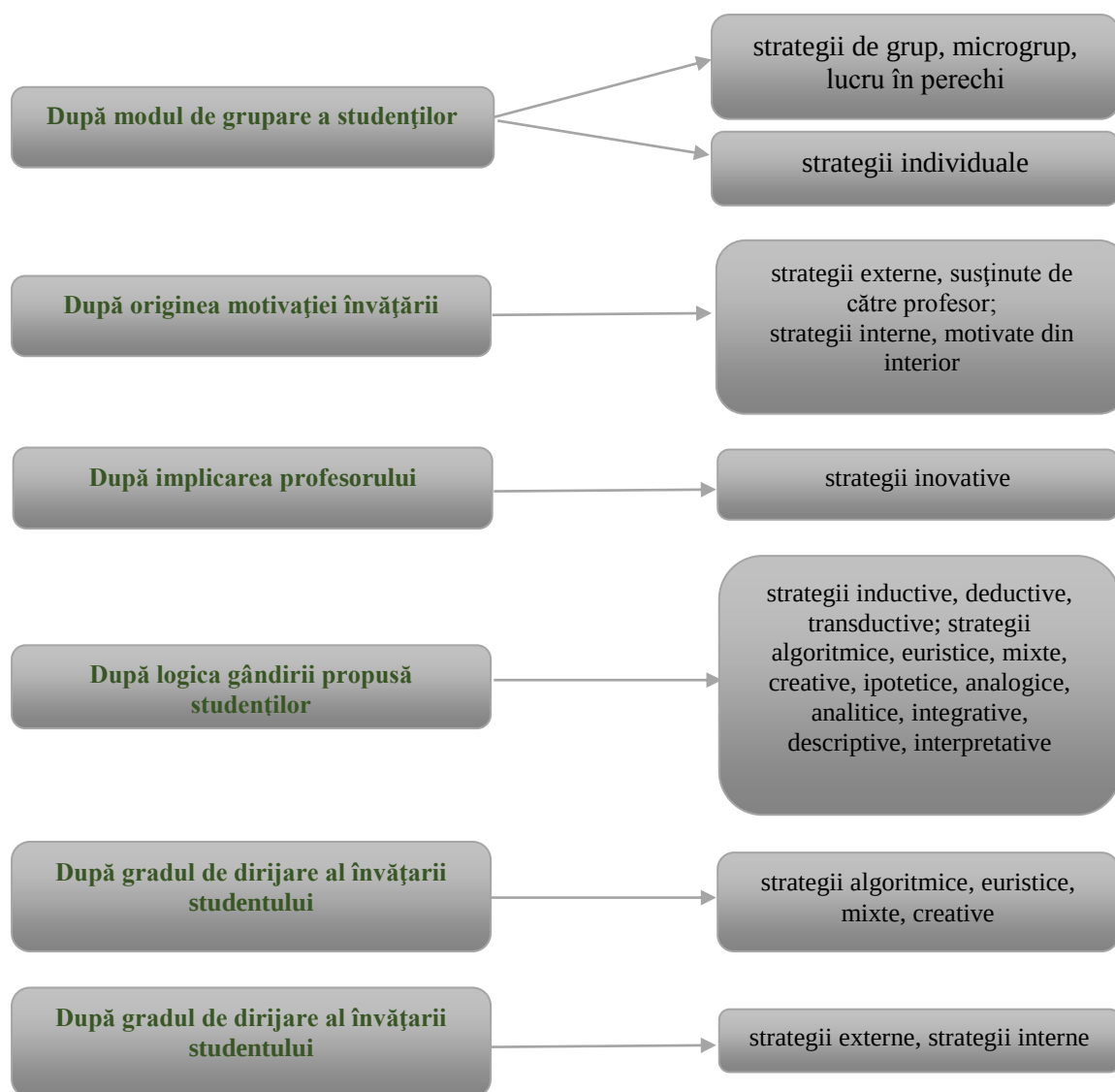


Fig. 2.10. Strategii didactice utilizate la predarea cursului POO

• *Principiul accesibilității* stabilește concordanța dintre sarcinile de învățare și posibilitățile reale ale studentului. În general, tehnologia POO este folosită la soluționarea problemelor cu un grad sporit de complexitate. Prin urmare, e necesar de elaborat sarcini accesibile pentru student, ținând cont de gradul de interes din perspectiva programării orientate pe obiecte. Prin prisma SMÎ Moodle, implementarea principiului accesibilității este realizat, în funcție de stilul de învățare și inteligența studentului, prin parcurgerea conținuturilor modulelor cursului *Programarea Orientată pe Obiecte* plasat pe Moodle, în ritmul propriu. Unele module ale cursului au fost create având în vedere activități în grup sau individuale, cum ar fi: alegerea unei teme de referat, alegerea unei probleme pentru soluționare, participarea la dezvoltarea modulului wiki.

- *Principiul corelației dintre concret și abstract* este aplicat pentru a ajuta studenții să treacă de la gândirea concretă la formarea gândirii abstracte. Abstractizarea, fiind caracteristica de bază a POO, este importantă în rezolvarea problemelor. Abstractizarea înseamnă concentrarea pe detaliile esențiale în rezolvarea problemelor, în timp ce detaliile neesențiale sunt ignorate, aceasta având ca efect simplificarea problemei și concentrarea atenției asupra aspectelor de soluționare a problemei. Pe sistemul Moodle, acest principiu se aplică prin utilizarea demonstrației inductive sau deductive, prin intermediul unor modele de simulare a situațiilor reale.

- *Principiul sistematizării și continuității.* Conținuturile cursului *Programarea Orientată pe Obiecte* sunt organizate și proiectate astfel încât să asigure o parcurgere logică, formând competențe OO, care, la rândul lor, asigură formarea de noi competențe. Astfel, însușind modalitatea de creare a claselor în C++, principiul moștenirii în POO, reprezentat prin derivarea claselor, va forma capacitatea de operare cu conceptul de moștenire și, în continuare, de polimorfism. Din perspectiva SMÎ Moodle, principiul sistematizării și continuității este realizat în POO prin proiectarea conținutului pe module cu o continuitate logică, coerentă, care permite o parcurgere în ritm propriu și asigură o învățare progresivă.

- *Principiul însușirii profunde a cunoștințelor* constă în aprofundarea cunoștințelor, în folosirea eficientă a capacităților și a deprinderilor dobândite anterior în activitățile de învățare. Cursul *Programarea Orientată pe Obiecte*, plasat pe sistemul Moodle, are o bază intuitivă cu aplicații practice; este un curs sistematizat și permite însușirea conștientă prin efort propriu; sunt acceptate aprofundări și completări continue. Se poate reveni la modulele cursului, până când vor fi atinse toate obiectivele.

- *Principiul participării active și conștiente la procesul didactic* este exprimat prin gradul de implicare a studenților în propriul proces de formare a competențelor obiect-orientate, bazat pe participarea lor conștientă și activă în procesul de studiere a principiilor de bază ale POO, care determină eficiența procesului educațional. În SMÎ Moodle, acest principiu este respectat prin implicarea în multiplele activități oferite de sistem, desfășurate sincron sau asincron, cu îndeplinirea mai multor sarcini (forum, chat, wiki).

Eficiența învățării în cadrul procesului instructiv poate fi crescută printr-un management superior al activității de învățare. Managementul învățării la început se realizează de către profesor și instruit, trecând treptat în atribuțiile celui instruit. Dacă până în prezent managementul învățării era axat doar pe aspectele cognitive ale învățării, cercetările

psihopedagogice demonstrează, din ce în ce mai mult, necesitatea unui management complex: al atitudinilor, al motivației, al timpului, al tuturor forțelor psihoeducaționale ale studentului.

Cererile de perfecționare și modernizare a metodologiei didactice conduc la dezvoltarea caracterului activ al metodelor de instruire, la aplicarea unor metode cu un caracter formativ pronunțat, întru valorificarea tehnologiilor informaționale.

Ioan Neacșu afirmă că „educatorii sunt solicitați astăzi, în mod continuu, să promoveze învățarea eficientă. Și nu orice învățare eficientă, ci una participativă, activă și creativă” [126, p. 12]. Conceptul *metodă de învățământ* provine din cuvântul grec „methodos”, ce are semnificația de “drum spre”, “cale de urmat” – în scopul atingerii unor obiective determinate în prealabil. Astăzi însă, metodele de învățământ reprezintă modalități de acțiune, instrumente prin intermediul cărora instruiții își însușesc cunoștințe, își formează și își dezvoltă competențe și deprinderi intelectuale și practice în mod independent sau sub tutela profesorului.

Dezvoltarea calităților intelectuale și morale, dobândirea cunoștințelor și formarea abilităților sunt influențate de metodele de instruire și educație. Prin urmare, metodele îndeplinesc funcțiile de instruire și cunoaștere (însușirea unor cunoștințe, priceperi, deprinderi) și formative (formarea și desăvârșirea calităților personale).

Astfel, din punct de vedere metodologic, procesul de învățământ este orientat spre obiectivele de cunoaștere și acțiune, adică spre crearea unor noi structuri cognitive ale instruitului.

Cercetătorul I. Cerghit afirmă că ”metoda este știință, pentru că cere un efort de elaborare științifică; este tehnică, pentru că reprezintă un mod concret de acțiune; este artă, pentru că are o mare putere de adaptabilitate la situația în care este implicată” [127, p. 9].

De fapt, metodele de instruire sunt modalități practice de utilizare a mijloacelor didactice și a unor tehnici specifice învățământului, modalități de organizare și desfășurare a activității instructiv-educative.

După M. Ionescu [128], principalele funcții ale metodelor de învățământ sunt:

- *funcția operațională / instrumentală* – asigură atingerea obiectivelor educaționale;
- *funcția cognitivă* – de dirijare a cercetării, cunoașterii, învățării, în scopul însușirii unor cunoștințe;
- *funcția formativ-educativă* – de formare și dezvoltare a unor capacități, deprinderi, atitudini, comportamente;
- *funcția normativă* – aspecte metodologice care oferă sugestii adresate profesorilor despre cum să perdea, iar studenților – despre cum să învețe;

- *funcția motivațională* – de stimulare a interesului cognitiv, de susținere a procesului educațional.

Metodele de instruire constituie elementul de bază al strategiilor didactice. De aceea, opțiunea pentru o anumită strategie didactică implică utilizarea unor metode de învățare specifice.

Totodată, metodele didactice determină eficiența învățării. De aceea, alegerea metodelor corespunzătoare fiecărei activități didactice are un impact important asupra procesului educațional.

Unele dintre metodele utilizate de autor în predarea cursului *Programarea Orientată pe Obiecte* cu ajutorul sistemului Moodle sunt:

- ***Învățarea bazată pe proiect*** (ÎBPr), care este un model de instruire centrat pe student. Metoda proiectului include diferite strategii de instruire, destinate implicării studenților în procesul didactic, indiferent de stilul lor de învățare [129]. Aceasta este una dintre principalele metode pentru atingerea obiectivelor educaționale principale.

”Project method” a fost inițiată de J. Dewey și se bazează pe principiul învățării prin acțiunea practică, cu finalitate reală – “learning by doing” sau “învățare prin practică”. Proiectul este înțeles ca o temă de cercetare care va fi realizată de către studenți prin îmbinarea cunoștințelor cu activitatea practică într-un termen determinat. Învățarea bazată pe proiect presupune colectarea informațiilor, prelucrarea și sintetizarea lor, interpretarea și reflecția personală, cooperarea în realizarea sarcinilor.

În procesul de învățare bazată pe proiecte se dezvoltă următoarele abilități: de a lucra în echipă, de a rezolva probleme, de comunicare scrisă și verbală, de inițiativă. Acestea sunt principalele abilități pe care angajatorii le apreciază cel mai mult la potențialii angajați.

Includerea activității de proiect în studierea programării este o încercare de a crea situații similare celor din lumea reală. Motivul pentru care s-a dorit aplicarea metodei proiectului la acest curs a fost susținută de ideea că implicarea unui număr mare de studenți permite rezolvarea unor sarcini mai largi și mult mai complexe din domeniul POO. Activitățile au fost realizate în câteva etape.

1. *Examinarea subiectului.* Împreună cu studenții a fost examinat un anumit subiect, s-a ales o problemă importantă, apoi s-a dezvoltat un plan de acțiune.

2. *Elaborarea planului de acțiune.* După ce au fost adunate informațiile despre subiect, studenții au fost capabili să aleagă o sarcină precisă în legătură cu subiectul. Aceasta a fost urmată de repartizarea sarcinilor care trebuiau îndeplinite, calendarul și definirea evenimentelor-cheie. În această fază pot fi dezvoltate idei individuale, pentru că gândurile originale și soluțiile

unice sunt binevenite. Fiecare pas trebuie realizat într-un anumit termen și să aibă un responsabil.

3. *Realizarea planului de acțiune.* În această etapă, studenții ar trebui să experimenteze succesul deja în primii pași, astfel încât ei să nu-și piardă entuziasmul. Pe măsură ce procesul avansa, studenților li s-a transferat tot mai multă responsabilitate. Anumiți studenți au fost desemnați lideri de grup, alții au fost responsabili pentru echipament, iar alții au prezentat colegilor următoarele sarcini. Este necesar de menținut entuziasmul studenților pe întreaga durată a proiectului.

Avantajele metodei proiectului:

- ajută la dezvoltarea normelor și valorilor sociale în rândul studenților;
- oferă oportunități pentru corelarea diferitelor elemente ale subiectului;
- ajută la creșterea cunoștințelor studenților ca rezultat al cooperării strânse între ei.

Dezavantajele metodei proiectului:

- proiectul nu poate fi planificat pentru toate subiectele;
- nu este economic din punct de vedere al timpului și al costurilor;
- planificarea și supravegherea proiectelor este foarte dificilă pentru un profesor [129];

Metoda proiectului implică gândirea critică, rezolvarea de probleme, diverse forme de colaborare. Dar cel mai important moment acestei metode este etapa de prezentare publică a rezultatelor. În această fază, cunoștințele sunt demonstrate prin punerea lor practică, iar rezultatele sunt supuse analizei și criticii externe.

În [82] sunt prezentate opt caracteristici ale metodei proiectului:

- ✓ Studenții se angajează în rezolvarea unor probleme din lumea reală.
- ✓ Studenții sunt implicați în activități de cercetare, formându-și abilități de planificare și de soluționare a problemelor.
- ✓ Își creează aptitudini specifice în contextul activităților din proiect.
- ✓ Metoda facilitează învățarea și aplicarea abilităților de comunicare.
- ✓ Permite utilizarea practică a competențelor necesare pentru dezvoltarea profesională.
- ✓ Include perspective de atingere a finalităților de studiu.
- ✓ Metoda dezvoltă analiza critică.
- ✓ Se finalizează cu o prezentare ce demonstrează realizarea sarcinilor.

De regulă, finalitatea procesului de predare-învățare la disciplinele informatice constă în elaborarea unor proiecte. Prin urmare, această metodă este una dintre metodele indicate în

procesul de predare-învățare-evaluare a disciplinelor informatice, care are ca efect dezvoltarea capacităților cognitive de nivel superior ale studenților.

În cadrul experimentului pedagogic desfășurat pentru cursul *Programarea Orientată pe Obiecte*, această metodă a fost folosită la elaborarea unor resurse digitale (prezentări) pentru disciplina POO.

În procesul de determinare a etapelor proiectului au fost stabilite:

- etapele fiecărei activități (studierea bibliografiei, elaborarea prezentării în format PowerPoint) și sarcinile de lucru;
- durata fiecărei etape;
- etapa de finalizare a sarcinii de lucru.

Profesorul a monitorizat activitatea fiecărui student, iar evaluarea a fost efectuată după următoarele criterii:

- fiecare student prezintă bibliografia studiată și subiectul elaborat în format Power Point;
- proiectul este finalizat prin prezentarea subiectului elaborat în fața colegilor;
- produsul rezultat în cadrul proiectului este încărcat (upload) în sistemul Moodle;
- profesorul, împreună cu studenții, analizează și apreciază activitatea fiecărui participant la proiect.

În urma aplicării metodei proiectelor, studenții au dezvoltat capacități cognitive, abilități de comunicare, de rezolvare a problemelor, dezvoltând următoarele competențe:

- colectarea informațiilor la tematica propusă;
- prelucrarea și sintetizarea informațiilor;
- interpretarea și reflecția personală;
- cooperarea în realizarea sarcinilor.

Concluzionând asupra experienței de utilizare a acestei metode, menționăm faptul că studenții și-au dezvoltat capacitățile de investigare și de sistematizare a informațiilor; au identificat și au valorificat diverse surse de informare. Un alt rezultat benefic al utilizării acestei metode a fost sporirea motivației pentru învățare, precum și creșterea autonomiei și creativității în învățare.

• **Învățarea bazată pe probleme** (ÎBP) este o altă metodă practică de autor în experimentul pedagogic, care constă în punerea în fața studentului a unor dificultăți create în mod deliberat, în depășirea cărora, prin efort propriu, acesta învață ceva nou [130].

Esența acestei metode constă în crearea, pe parcursul învățării, a unor „situații-problemă” și în rezolvarea acestora de către studenți, pornind de la cunoștințele însușite anterior. Noile cunoștințe nu mai sunt „predate” gata elaborate, ci sunt obținute prin efort propriu. O situație-problemă desemnează o situație contradictorie, conflictuală, ce rezultă din trăirea simultană a două realități: experiența anterioară (cognitiv-emoțională) și elementul de noutate și de surpriză, necunoscutul cu care se confruntă subiectul [130].

După cum menționează cercetătoarea A. Globa, învățarea bazată pe probleme „constituie un model de implicare activă și interactivă a studenților în procesul didactic concentrat pe proiectarea activităților de lucru independent (individual sau de grup) în căutarea soluțiilor unei probleme dificile.”[89, p. 104]

Învățarea bazată pe probleme este o abordare ce provoacă studenții să învețe prin implicarea într-o problemă reală. Este un format care dezvoltă simultan atât strategii de rezolvare a problemelor, cât și baze de cunoștințe și abilități disciplinare, plasând studenții în rolul activ al rezolvatorilor de probleme.

Învățarea pe bază de probleme este centrată pe student. ÎBP face o schimbare fundamentală: accentul este pus nu atât pe predare, cât pe învățare. Procesul vizează utilizarea puterii de rezolvare a problemelor pentru a-și îmbunătăți instruirea și motivația.

Există câteva aspecte unice care definesc abordarea ÎBP:

- Învățarea are loc în contextul unor probleme autentice din lumea reală.
- Într-un curs ÎBP, studenții și instructorul devin colegi și coevaluatori, pe măsură ce proiectează, implementează și perfecționează continuu planurile de învățământ.
- ÎBP stimulează studenții să își asume responsabilitatea pentru propria învățare.
- ÎBP este unică prin faptul că favorizează colaborarea dintre studenți, accentuează dezvoltarea abilităților de soluționare a problemelor în contextul practicii profesionale, promovează raționamentul eficient și învățarea autodirijată, vizează creșterea motivației pentru instruirea continuă de-a lungul vieții.

Situațiile-problemă pot lua naștere dacă:

- ✓ se produce o tensiune între achizițiile anterioare și noua situație;
- ✓ există dorința de a cunoaște sau a explica noua situație;
- ✓ există condiții cognitive/emotiv-motivaționale de rezolvare;
- ✓ există climat favorabil creativității și rezolvării de probleme.

Pașii învățării prin problematizare sunt [128]:

- Formularea problemei – confruntarea cu problema – perceperea și conștientizarea problemei – primii indici orientativi pentru rezolvarea problemei.
- Studierea aprofundată – înțelegerea problemei – restructurarea datelor problemei.
- Căutarea soluțiilor posibile la problema pusă – analiza condițiilor sarcinii problematice – selectarea și actualizarea unor achiziții.
- Formularea ipotezelor de soluționare a sarcinii problematice.
- Verificarea ipotezelor emise.
- Obținerea rezultatului final.
- Validarea soluției.

Prin natura lor, disciplinele informatice invocă rezolvarea de probleme, iar disciplina *Programarea Orientată pe Obiecte* în mod special operează cu probleme din lumea reală. Astfel, pe parcursul experimentului pedagogic, studenților le-au fost propuse probleme formulate în limbajul obișnuit, care necesitau o analiză și o reformulare în noțiuni ale POO. Problemele erau analizate și discutate în cadrul grupului, apoi soluțiile propuse erau ordonate după importanța lor. În acest timp, grupurile erau monitorizate de către profesor, acesta dirijând procesul de învățare. În continuare, soluțiile erau analizate de întregul grup, pentru a elucida soluția optimă. Instrumentele Moodle folosite în abordarea metodei ÎBP au fost chatul, forumul, e-mailul.

Unul dintre cele mai mari avantaje ale ÎBP este că studenții se bucură cu adevărat de procesul de învățare. ÎBP este o metodă provocatoare, care face procesul didactic interesant pentru studenți, deoarece aceștea sunt motivați să învețe prin nevoia de a înțelege și a rezolva probleme reale.

- **Învățarea autodirijată (ÎAD).** Conceptul de învățare autodirijată a fost discutat pentru prima dată, în literatura de specialitate, încă în 1926. Învățarea autodirijată este un proces prin care indivizii își identifică nevoile de învățare, formulând obiectivele de instruire; identifică resursele de învățare, aleg și implementează strategiile de instruire și finalitățile de studii. Educația pe tot parcursul vieții a fost identificată ca o abilitate importantă pentru lumea actuală.

Caracteristicile principale ale învățării autodirijate sunt [82]:

- responsabilitate înaltă pentru deciziile legate de procesul de învățare;
- studenții implicați în învățarea autodirijată dobândesc încredere și abilități pentru instruirea pe tot parcursul vieții;
- învățarea autodirijată implică diferite activități și resurse, cum ar fi participarea în diverse grupuri de studii, discuții online, activități reflective (bloguri);

- profesorii au un rol important în a ajuta studenții să dobândească abilitățile de învățare autodirijată, în dezvoltarea gândirii critice, în evaluarea finalităților de studii.

Conceptul de învățare autodirijată presupune o învățare autoresponsabilă și un control asupra propriului comportament. Scopul educației moderne este de a transforma omul care este educat de alții în omul care se educă pe sine însuși.

Un beneficiu important al abordării autodirecționate este că se poate aborda una dintre cele mai persistente probleme în educație: creșterea exponențială a cunoștințelor.

• **Învățarea bazată pe întrebări** (ÎBÎ) este o abordare a predării și a învățării care pune întrebările, ideile și observațiile studenților în centrul experienței de instruire. La baza acestei abordări se află ideea că atât instructorii, cât și studenții împărtășesc responsabilitatea pentru procesul didactic. Pentru studenți, procesul implică investigații deschise într-o întrebare sau o problemă, cerându-le să se angajeze în raționamente bazate pe dovezi și în rezolvarea problemelor creative. Pentru instructori este important să fie receptivi la nevoile de învățare ale studenților [131].

Modelul de învățare bazat pe întrebări a apărut în anii '60 ai secolului XX, în timpul mișcării "descoperirea învățării", și se bazează pe ideea că indivizii pot învăța prin investigarea diferitor probleme și prin experiențe sociale. În loc să fie nevoiți să memoreze informații din materiale tipărite, instructorii au încurajat studenții să efectueze investigații și să-și satisfacă curiozitatea, să-și extindă baza de cunoștințe și să-și dezvolte abilitățile mintale.

Învățarea bazată pe întrebări are capacitatea de a spori potențialul intelectual al studenților, îndemnându-i să:

- dezvolte abilități de interogare, cercetare și comunicare;
- colaboreze în afara sălii de clasă;
- rezolve probleme din viața reală;
- participe la crearea ideilor și cunoștințelor.

Există anumite principii ce guvernează învățarea bazată pe întrebări:

1. Studenții se află în centrul întregului proces, în timp ce instructorii, resursele și tehnologia sunt organizate în mod adecvat, pentru a-i sprijini.
2. Toate activitățile de învățare se bazează pe abilitățile de procesare a informațiilor.
3. Instructorii facilitează procesul de învățare.
4. Accentul se pune pe dezvoltarea abilităților de procesare a informațiilor și a înțelegerii conceptuale.

Există patru forme de întrebări care sunt utilizate în acest model de învățare :

- *Întrebare de confirmare.* Studenții primesc o întrebare și metoda de obținere a rezultatului, iar rezultatul final fiind deja cunoscut. Scopul este de a confirma rezultatele. Acest lucru le permite studenților să consolideze ideile deja existente și să-și perfecționeze abilitățile de investigare.

- *Întrebare structurată.* Studenții primesc întrebarea și metoda de realizare a rezultatului. Scopul este de a oferi o explicație care este deja susținută de dovezile adunate în timpul procesului de investigație.

- *Întrebare ghidată.* Studenților le este adresată o întrebare, iar scopul principal este de a proiecta metoda de investigare și apoi de a testa întrebarea însăși.

- *Întrebare deschisă.* Studenții formulează propriile întrebări, elaborează metode de investigare și apoi efectuează investigarea. Rezultatele sunt prezentate la sfârșitul procesului.

Învățarea pe bază de întrebări le oferă studenților posibilitatea de a explora pe deplin problemele și scenariile, astfel încât să învețe nu numai rezultatele, ci și procesul în sine. Ei sunt încurajați să pună întrebări, să exploreze informații, să obțină dovezi, să conceapă argumente convingătoare cu privire la modul în care au ajuns la rezultatul final.

Metoda este strâns legată de instrumentele soft și comunicaționale. În calitate de resurse de învățare apar sistemele de asistență locale sau web.

Această metodă a fost implementată la disciplina *Programarea Orientată pe Obiecte* în cadrul experimentului pedagogic la tema “Constructori și destructori”. A fost definită întrebarea la care trebuiau să răspundă studenții, a fost inițiată o discuție pe acest subiect, iar studenții au fost rugați să noteze cinci proprietăți ale constructorilor.

Lucrând în grupuri mici, studenții au împărtășit informațiile pe care le-au colectat, utilizând forumul și chatul. În timpul discuției, ei au avut oportunitatea de a căuta informație referitor la subiect și de a face conexiuni la un video sau prezentări legate de constructori, oferind informații explicate despre subiectul pus în discuție, demonstrând abilități de gândire critică.

Faptul că studenții își notează cele cinci idei despre ceea ce înseamnă un constructor oferă o oportunitate pentru o evaluare de bază a înțelegerii. Mai târziu, pe măsură ce studenții se dezvoltă și își exprimă în continuare înțelegerea asupra constructorilor, exercițiul inițial este repetat, iar studenții pot compara rezultatul obținut cu munca lor anterioară, atestând o creștere a înțelegerii lor.

Astfel, în urma aplicării metodei de învățare pe bază de întrebări, studenții au dezvoltat abilități de interogare, cercetare, comunicare și colaborare.

- Metoda **studiului de caz** este un stil de predare care implică învățarea bazată pe probleme și promovează dezvoltarea abilităților analitice. Studiul de caz facilitează dezvoltarea învățării cognitive și învățarea interdisciplinară și poate fi folosit pentru a evidenția conexiunile dintre subiectele academice specifice și problemele și aplicațiile din lumea reală. Astfel, crește motivația studenților de a participa la activități de clasă, ceea ce promovează învățarea și crește performanța evaluărilor.

Această metodă a fost inițiată în Franța, în 1926, de către Andre Siegfried. Cercetătorul R. Yin definește studiul de caz ca o cercetare empirică ce analizează un fenomen contemporan în contextul vieții reale, o analiză în care sunt utilizate surse multiple de date [132].

Scopul acestei metode, valoroase din punct de vedere euristic, constă în: realizarea contactului participanților cu realitățile complexe, dezvoltarea capacităților decizionale și abilităților de soluționare a problemelor, sistematizarea și consolidarea cunoștințelor, educarea personalității și a atitudinii față de ceilalți participanți, precum și în exersarea capacității de evaluare și luare a deciziei în condiții similare celor reale.

Elementele principale ale unui studiu de caz sunt:

- 1) studenții, care sunt angajați în procesul de învățare printr-o problemă ce trebuie rezolvată;
- 2) descrierea contextului problemei;
- 3) datele, care pot fi prezentate începând cu tabele și terminând cu linkuri către adrese URL, documente, imagini, video sau audio;
- 4) sarcinile cazului, care pot fi individuale sau de grup, astfel încât studenții să se poată gândi la soluții și să-și împărtășească ideile.

Procesul de rezolvare a unei probleme din lumea reală îi ajută pe studenți să asimileze cunoștințe noi. Un avantaj major al metodei studiului de caz constă în faptul că studenții se angajează activ în găsirea principiilor prin abstractizarea exemplurilor, dezvoltând abilități în:

- Rezolvarea de probleme.
- Utilizarea instrumentelor analitice.
- Luarea deciziilor în situații complexe.
- Confruntarea cu ambiguitățile.

- Metoda **cubului** este o metodă de învățare prin cooperare, ce presupune explorarea unui subiect din mai multe perspective, permițând abordarea complexă și integratoare a unei teme [129].

Această metodă a fost aplicată în experimentul pedagogic condus de autor, la studierea noțiunilor de *clasă*, *obiect*, *instanță*, *date membru* și *metode membru*, *metode de acces*. Studenții

au fost repartizați în grupuri mici, fiecare grup examinând tema din perspectiva cerinței de pe una dintre fețele cubului. Astfel, studenții au descris noțiunile de: *clasă, obiect, instanță, date și metode membru, metode de acces*; au comparat obiectul de tip clasă cu instanța clasei; au analizat datele membru și metodele membru; au asociat noțiunea de obiecte ale clasei cu lumea reală; au aplicat metodele de acces pentru încapsularea informației; au argumentat eficiența utilizării claselor.

În timpul aplicării metodei cubului, studenții au elaborat materialul comun, utilizând GoogleDocs. Documentele realizate au fost uploadate pe platforma Moodle, iar fiecare student și-a expus punctul de vedere în legătură cu tema propusă în cadrul forumului dedicat acestei teme.

Pe Wiki a fost lansată tema “Controlul accesului la datele și funcțiile membre ale claselor”, unde echipele au plasat informația acumulată pe parcursul lecției.

Această metodă stimulează creativitatea studenților, dezvoltă capacitățile lor decizionale și colaborative, stimulează gândirea critică.

- Metoda **brainstorming** (sau *asaltul de idei*) reprezintă o metodă intuitivă de stimulare a creativității. Ca metodă de discuție, brainstormingul a fost sistematizat în 1948 de către profesorul Alexander Osborn. Scopul metodei asaltului de idei este emiterea unui număr cât mai mare de idei privind modalitatea de rezolvare a unei probleme, în speranța găsirii soluției optime prin combinarea lor. Ideile sunt obținute prin stimularea creativității în cadrul grupului, într-o atmosferă dezinhibată, lipsită de critică, obținută prin amânarea momentului evaluării. Un principiu al brainstormingului este: „cantitatea generează calitatea” [129].

Deoarece experimentul pedagogic, desfășurat la cursul *Programarea Orientată pe Obiecte*, implică folosirea intensă a instrumentelor colaborative Moodle, această metodă a fost folosită în varianta *brainstorming electronic* la tema “Moștenirea datelor și a funcțiilor membre din clasele de baza”. Prin intermediul forumului de discuții, studenții au emis o mulțime de idei în cadrul tematicii propuse. Participanții grupurilor electronice nu sunt „față în față” (sunt în contact doar în rețeaua electronică) și astfel dispar barierele de statut și sustragerea de la efort, fiind eliminate toate inconvenientele. Varianta electronică a brainstormingului este mai relaxantă relațional, iar ideile noi de gândire asupra unor probleme apar mai bine într-un brainstorming individual.

Tehnologia didactică reliefează strategiile educaționale care contribuie la o bună asimilare a cunoștințelor, la formarea priceperilor, deprinderilor, aptitudinilor și astfel formează competențe.

Conform Curriculumului național, competența este definită ca „un ansamblu/sistem integrat de cunoștințe, capacități, deprinderi și atitudini dobândite de studenți prin învățare și mobilizate în contexte specifice de realizare, adaptate vârstei studentului și nivelului cognitiv al acestuia, în vederea rezolvării unor probleme cu care acesta se poate confrunta în viața reală”.

Conceptul de competență, în pedagogie, a fost introdus de J. Piaget și L. D'Hainaut. Acest termen are numeroase interpretări. Astfel, pedagogii N. Silistraru și S. Golubițchi [133, p. 71-87] menționează că „a fi competent înseamnă a avea capacitatea de a rezolva o problemă, a adopta o decizie, a desfășura cu eficacitate o anumită activitate”.

În opinia lui C. Cucos, „competențele sunt ansambluri structurate de cunoștințe și deprinderi dobândite prin învățare, ce apar ca structuri operante cu ajutorul cărora se pot identifica și rezolva, în contexte diverse, probleme caracteristice unui anumit domeniu” [134, p. 50].

Metodele de învățământ, sub influența tehnologiilor informaționale, sunt într-o evoluție continuă și se orientează spre eficientizarea procesului de formare din perspectiva educației permanente. Didactica centrată pe competențe denotă noile dimensiuni pe care le pot lua metodele în contextul orientării spre formarea competențelor. Iar mijloacele de învățământ, relativ noi, de prezentare a informațiilor în format electronic au dinamizat posibilitățile de organizare a instruirii și permit utilizarea unui volum considerabil de cunoștințe, structurate în forme accesibile [135,136].

Deoarece majoritatea mediilor de dezvoltare a produselor-program sunt bazate pe tehnologia programării orientate pe obiecte, se poate spune că deținerea competențelor de POO este o caracteristică absolut necesară informaticianului profesionist modern.

B.F. Skinner susține că eficiența învățării este condiționată de organizarea condițiilor de învățare a instruiților, mai exact spus, de organizarea „condițiilor de întărire” a învățării [137]. Prin urmare, un rol important în formarea competențelor POO îl au resursele de învățare, care în cazul nostru au fost plasate pe SMÎ *Moodle*.

După cum menționează N. Deinego, dobândirea competențelor depinde de doi factori importanți: conținuturile și situațiile în care se produce învățarea [118]. Acest lucru presupune restructurarea și eșalonarea unităților de conținut. Ca principiu de structurare a conținutului disciplinei a fost ales *principiul modular*. Ideea de modularitate în proiectarea curriculară a apărut din necesitatea de flexibilizare a organizării conținuturilor, indusă de educația permanentă [138].

Învățământul modular este caracterizat prin structurarea conținuturilor în module didactice, care includ volume de cunoștințe, situații didactice, activități și mijloace de instruire

delimitate, menite să satisfacă cerințele și posibilitățile unor grupuri de studenți (figura 2.11). Resursele de învățare, corespunzătoare materialul de curs, sunt împărțite în mai multe module, conform conținutului curriculumului. Modalitatea prietenoasă de abordare oferă o atractivitate deosebită și ajută studentul să treacă de barierele unui conținut anost și neprietenos. Accentul în cursurile elaborate în Moodle nu este pus pe furnizarea de informații, ci pe activitățile care presupun schimb de idei și achiziție de cunoștințe noi bazate pe cunoștințe anterioare.

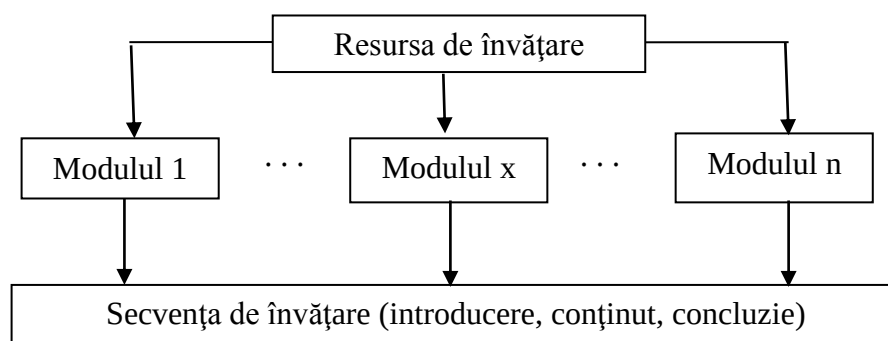


Fig. 2.11. Principiul modular al resursei de învățare

Astfel, luând ca bază principiul modular, structurarea conținuturilor la disciplina *Programarea Orientate pe Obiecte* a fost efectuată în felul următor:

Tabelul 2.1. Structurarea conținuturilor cursului universitar *POO*

Modulul	Unități de învățare
Modulul 1	Concepte de bază privind programarea orientată pe obiecte
Modulul 2	Clase și obiecte.
Modulul 3	Moștenire
Modulul 4	Polimorfism

În procesul elaborării resurselor de învățare, care contribuie la formarea competențelor specifice POO, autorul a respectat următoarele principii:

1. Organizarea subiectelor principale ale POO în module. Acest lucru permite o asimilare mai bună a conținutului prin vizualizarea secvențială a materialului online.
2. Utilizarea sistemului Moodle pentru revizuirea cursului ori de câte ori este necesar, folosind concomitent și comunicarea cu grupul de colaborare.
3. Efectuarea testării inițiale, care stabilește gradul de înțelegere a subiectului studiat și identifică problemele ce urmează a fi soluționate.
4. Propunerea exercițiilor practice axate pe teorie, necesare pentru stabilirea relațiilor între realitate și soluție.

5. Împărțirea procesului de învățare în câteva etape, fiecare etapă având un obiect de învățare, activități și teste pentru evaluarea progresului de învățare.

În consecință, ținând cont de aspectul didactic și metodologic al cursului *Programarea Orientată pe Obiecte*, de modalitatea de restructurare a unităților de conținut, au fost identificate următoarele competențe POO:

CPOO_1. Elucidarea noțiunilor și conceptelor de bază ale tehnologiei orientate pe obiecte.

CPOO_2. Implementarea ierarhizării în baza relației de moștenire.

CPOO_3. Modificarea stării și comportamentului obiectului în funcție de specificul problemei.

CPOO_4. Implementarea conceptelor POO în soluționarea problemelor, folosind un limbaj de programare orientat pe obiecte (limbajul C++)

Finalitățile de studiu sunt rezultatele învățării de care dau dovadă studenții la sfârșitul procesului de învățare. Finalitățile de studiu sunt ceea ce „pot face” studenții, ceea ce „știu” și „au învățat”. Această abordare este în concordanță directă cu modelul centrat pe student. Astfel, rezultă că finalitățile sunt acțiuni și performanțe care întruchipează și reflectă competența cursantului în utilizarea cu succes a conținutului, informațiilor, ideilor și instrumentelor. Deoarece finalitățile apar la finele unui proces de învățare, putem spune că ele reprezintă rezultatul final care este căutat de învățare.

Cunoștințele și abilitățile curriculare specifice sunt dezvoltate în baza finalităților de studiu și ajută în mod direct studenții la dezvoltarea abilităților de performanță. Conceptul contemporan al învățării ca activitate autodirijată susține că studenții poartă o mare responsabilitate pentru învățarea lor, astfel încât responsabilitatea supremă este văzută ca fiind împărțită între instruit–școală–profesor–instruit [139].

Sistemele bazate pe finalități de studiu construiesc totul pe un cadru clar definit de rezultatele finale. Curriculumul, strategiile de instruire, evaluările și standardele de performanță sunt dezvoltate și implementate pentru a facilita obținerea rezultatelor finale. În învățământul bazat pe finalități, curriculumul, instruirea și evaluarea sunt privite ca mijloace flexibile, pentru realizarea unor scopuri de învățare clar definite [139].

Utilizarea finalităților de studiu și competențelor este necesară pentru a face programele de studiu și unitățile de curs sau modulele orientate către student. Această abordare impune ca abilitățile și cunoștințele-cheie de care trebuie să dea dovadă studentul, în timpul procesului de învățare, să determine conținutul programului de studii.

Finalitățile de studiu și competențele se concentrează atât asupra cerințelor disciplinei, cât și asupra necesităților societății, din perspectiva încadrării într-o societate modernă și a cerințelor pieței de muncă.

Diferențele dintre competențe și finalități de studiu sunt formulate și ilustrate în figura 2.12.

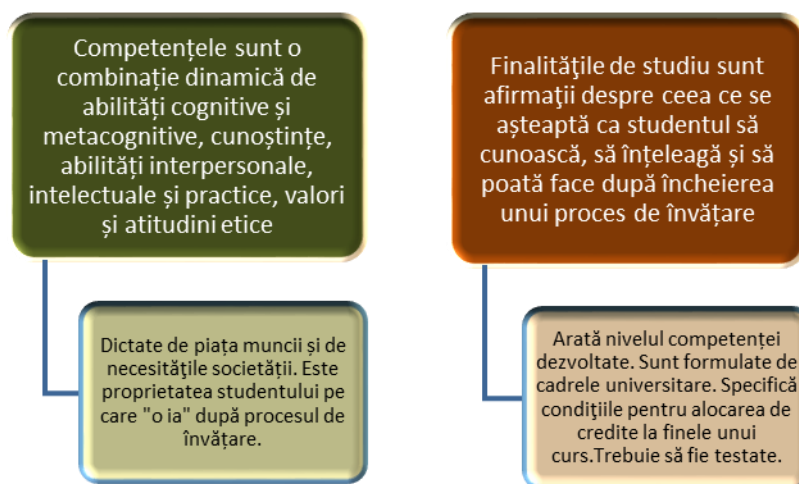


Fig. 2.12. Competențe și finalități de studiu

Curriculumul centrat pe competențe asigură o eficiență mai mare proceselor de predare-învățare-evaluare, competența devenind o coeziune a lor. Avantajele curriculumului centrat pe competențe sunt:

- Răspunde exigențelor pieței muncii.
- Instruirea este orientată către finalitățile de studiu.
- Se lucrează în echipă, pentru actualizarea ofertei educaționale.
- Studenții sunt implicați în activitatea de învățare continuă și de autoevaluare a competențelor.
- Asigură flexibilitatea parcursului educațional prin organizarea sistemului modular.
- Poate încorpora, în orice moment al procesului educativ, noi mijloace sau resurse didactice.
- La modulele parcurse pot fi adăugate altele noi, pentru formarea continuă.

Tehnologiile informaționale moderne dau o nouă interpretare educației, care pune în evidență cunoștințele și competențele studenților din perspectiva achiziției noțiunilor și a aptitudinilor cu caracter general, ce corespund cerințelor actuale ale vieții profesionale și sociale, ale pieței muncii.

Astfel, luând în considerație conceptele analizate în acest capitol, la disciplina *Programarea Orientată pe Obiecte* a fost propus un curriculum nou, ce reflectă pe deplin principiul modelului centrat pe student din perspectiva utilizării SMÎ.

Activități de predare-învățare-evaluare

Învățarea tradițională sau face to face (față în față) este o metodă de predare încă destul de populară, deoarece majoritatea oamenilor s-au dezvoltat într-un mediu tradițional de instruire. Învățarea față în față are câteva avantaje, cum ar fi:

- învățarea într-un mediu social de interacțiune;
- facilitarea schimbului de idei;
- reducerea posibilităților neînțelegeri.

Cu toate acestea, învățarea face to face limitează posibilitățile de personalizare a conținutului cursului, învățarea autodirijată și învățarea centrată pe student.

Evoluția tehnologică a produs modificări în toate domeniile vieții și, ca urmare a acestui efect, au avut loc schimbări și în procesul didactic. Instruiții s-au pomenit în fața necesității de a învăța să gestioneze un număr impresionant de informații, să le analizeze, să ia decizii să-și dezvolte cunoștințele, pentru a face față provocărilor tehnologice actuale.

Modernizarea procesului educațional în contextul unei societăți informaționale este imposibilă fără utilizarea instrumentelor digitale moderne.

Utilizarea tehnologiilor informaționale și de comunicare întru susținerea procesului de predare-învățare, în sălile de clasă și la disciplinele curriculare, poate fi menținută prin abordarea echilibrată a integrării tehnologiei, pedagogiei și conținutului, în scopul de a asigura un mediu propice de învățare.

Astfel, actul învățării nu mai este considerat a fi doar efectul demersurilor și muncii profesorului, ci rodul interacțiunii studenților cu calculatorul și al colaborării cu profesorul.

Analizând modalitatea de realizare a învățării în învățământul tradițional și modalitatea de realizare a instruirii în învățământul modern, am putea determina un șir de distincții (tabelul 2.2).

Învățământul la distanță (ÎD) este o formă de învățământ dinamică și foarte flexibilă, accesibilă unei categorii largi de populație. ÎD se bazează pe motivația personală a cursantului. Cursurile propuse pe platformele ÎD acoperă o paletă largă de domenii – de la computere și robotică, fizică sau matematică, până la științele sociale, economie sau drept.

Tabelul 2.2. Distincțiile dintre învățarea în învățământul tradițional și în cel modern [140]

Realizarea învățării în învățământul tradițional		Realizarea învățării în învățământul modern	
- memorarea și reproducerea cunoștințelor transmise de cadrul didactic - competiția cu scop de ierarhizare		- apel la experiența proprie - promovează învățarea prin colaborare - dezvoltă gândirea critică în confruntări	
Avantaje:	Limite:	Avantaje:	
- Stimulează productivitatea, promovează aspirații mai înalte - pregătește studenții pentru viața competitivă	- Generează conflicte, agresivitate - lipsa comunicării și a încrederii - amplifică teama de eșec - egoismul	Stimulează: - argumentarea unei opinii - cooperarea în rezolvarea sarcinilor și problemelor de lucru (de învățare) - inițiativa, spiritul întreprinzător - cutezanța, asumarea riscurilor - implicarea personală, gândirea liberă, creativă, critică	
Evaluarea în învățământul tradițional		Evaluarea în învățământul modern	
- măsurarea și aprecierea cunoștințelor (ce știe studentul) - accent pe aspectul cantitativ (cât de multă informație deține studentul)		- măsurarea și aprecierea capacităților (ce știe și ce poate să facă studentul) - accent pe elementele de ordin calitativ (sentimente, atitudini, relații etc.)	

Învățământul la distanță are următoarele avantaje față de abordarea clasică, ”tradițională” a instruirii:

- Oferă posibilitatea învățării continue (*Lifelong learning*) și dezvoltarea profesională.
- Instruiții învață în mod independent, în ritmul lor, în locul și timpul pe care ei îl aleg.
- Ritmul propriu: instruiții citesc materialele de studiu în pasul pe care ei îl aleg și ori de câte ori doresc.
- Locul de învățare este liber ales – depinde de mijlocul utilizat pentru distribuirea materialelor didactic.
- Oferta unor teme pe care nu le oferă cursurile / programele din regiunea respectivă.
- Posibilitatea de a participa la programe de cele mai înalte calitate și prestigiu.
- Alegerea modalității de învățare: învățarea activă sau pasivă, nivele diferite de interacțiune “clasice” – materiale scrise pe lângă notițele proprii, simulări interactive, discuții cu alți cursanți (e-mail, teleconferințe), mai multe componente multimedia (grafică, animații, sunet).

Blended Learning (BL) sau *învățământul mixt* este un concept de învățare modern, foarte flexibil, dezvoltat în scopul de a oferi fiecărui instruit un nivel avansat de cunoștințe.

Termenul de „blended learning” începe să fie folosit la sfârșitul secolului XX. În anul 2006 apare primul manual de învățământ mixt, scris de C.J. Bonk și C.R. Graham [141]. Autorii definesc sistemele de blended learning ca fiind acele sisteme care „combină interacțiunea directă dintre profesori și studenți cu instruirea asistată de calculator”. Ideea centrală a acestui tip de sistem o reprezintă „alegerea” oferită utilizatorilor, favorizând selectarea de către profesori a celor mai bune metode de predare, care să reflecte stilul de învățare al studenților, oferind, astfel, posibilitatea realizării unui învățământ centrat pe student [142].

Blended learning îmbină învățarea online cu învățarea clasică. Practic, un program de BL reunește avantajele celor două tipuri de instruire: calitate și flexibilitate, plus memorabilitate. În mediul unui program BL, cursanții au acces nelimitat la resursele de învățare, pot învăța în ritmul propriu, iar pentru o asimilare mai bună a conceptelor învățate și pentru aplicarea lor în activitatea curentă sunt organizate întâlnirile de grup, unde sunt discutate conceptele noi.

Tehnologiile învățământului mixt au proprietatea de a:

- extinde spațiile și oportunitățile disponibile pentru învățare;
- susține activitățile de gestionare a cursurilor (de exemplu, comunicarea, evaluarea, prezentarea, feedbackul);
- furniza studenților informații și resurse;
- angaja și motiva studenții prin interacțiune și colaborare.

Învățământul mixt reprezintă descoperirea unor modalități mai eficiente în sprijinirea studenților pentru atingerea obiectivelor de instruire și oferirea celor mai bune experiențe de învățare și predare. Integrarea învățării mixte în cursuri variază în funcție de: disciplină, anul de studiu, caracteristicile și necesitățile studenților, obiectivele cursurilor sau programelor, maniera de predare, încrederea și experiența în utilizarea tehnologiei.

Prin implementarea tehnologiei BL la disciplina *Programarea Orientată pe Obiecte* s-a urmărit îmbogățirea calității experienței de învățare a studenților nu numai prin interacțiunea față în față în clasă, ci și prin activități interactive de învățare prin utilizarea SMÎ Moodle ca, de exemplu, comunicarea, colaborarea, evaluarea.

Activitățile de predare și învățare în BL trebuie să fie semnificative și relevante, iar experiențele de învățare ar trebui să fie nu numai interactive, dar și participative, astfel încât procesele de cunoaștere și de colaborare să fie intensificate.

Proiectarea pentru învățarea mixtă necesită o abordare sistematică și conține următoarele etape (figura 2.13):

1. Planificarea integrării învățării mixte.
2. Proiectarea și dezvoltarea elementelor de învățare mixtă.

3. Implementarea designului învățării mixte.
4. Revizuirea (evaluarea) eficacității designului învățării mixte.
5. Planificarea pentru următoarea livrare a cursului, care implică îmbunătățirea experienței de învățare mixtă.

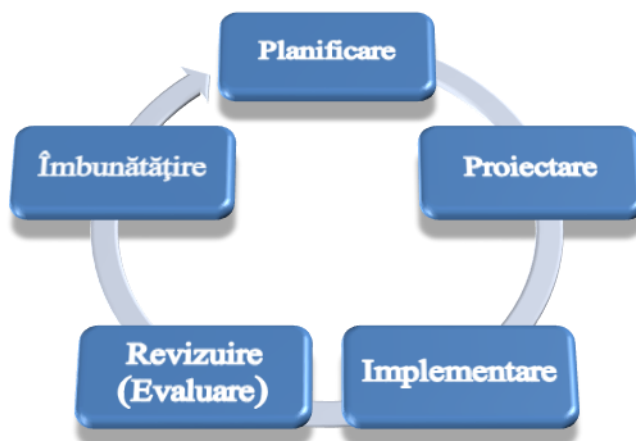


Fig. 2.13. Procesul de design al învățării mixte (după J.B. Biggs [143])

La planificarea cursului *Programarea Orientată pe Obiecte* s-a ținut cont de:

- Scopul și obiectivele de învățare, competențele și finalitățile de studiu pentru disciplina POO;
- Experiența studenților în utilizarea tehnologiilor informaționale;
- Accesul studenților la tehnologiile moderne;
- Motivația și capacitatea studenților de a utiliza aceste tehnologii.

La proiectarea componentelor învățării mixte pentru cursul *Programarea Orientată pe Obiecte* s-a ținut cont de câteva principii generale de proiectare, și anume:

✓ Obiectivele cursului, activitățile de predare–învățare și sarcinile de evaluare corespund etapei respective. Aceasta înseamnă că: (1) resursele și activitățile de predare–învățare sprijină în mod direct studenții, pentru ca aceștia să atingă obiectivele de învățare; (2) sarcinile de evaluare sunt în concordanță cu activitățile și obiectivele cursului și le permit studenților să demonstreze aceste obiective de învățare.

- ✓ Activitățile sunt consecvente și bine gândite.
- ✓ Activitățile de predare–învățare sunt corelate cu conținutul; de exemplu, activitățile de învățare sunt coordonate cu partea teoretică a cursului.

✓ Volumul cursului în condițiile blended learning nu depășește volumul cursului în modul tradițional.

În tabelul 2.3 sunt prezentate câteva exemple de utilizare a instrumentelor oferite de BL, în cadrul experimentului pedagogic pentru cursul *Programarea Orientată pe Obiecte*, întru susținerea studenților în vederea realizării anumitor obiective de învățare.

Tabelul 2.3. Exemple de utilizare a BL în cadrul cursului *Programarea Orientată pe Obiecte*

Obiectivul de învățare	Abordare BL în sprijinul obiectivului
Elucidarea noțiunilor și conceptelor de bază ale POO	Este oferit accesul la cursul <i>POO</i> plasat pe SMÎ Moodle
Cunoașterea tipurilor și a modalității de utilizare a constructorilor și a destructorilor în rezolvarea problemelor	✓ Este oferit accesul la cursul <i>POO</i> plasat pe SMÎ Moodle. ✓ Este prezentat un videoclip pentru sesiunea de consiliere. ✓ După vizionare, studenții vor rezolva probleme după modelul propus în videoclip. ✓ După aceasta, vor iniția discuții pe forumul de discuții asupra întrebărilor ce apar în timpul rezolvării problemelor.
Reamintirea termenilor specifici POO și a noțiunilor de <i>clasă</i> , <i>obiect</i> , <i>constructor</i> , <i>destructor</i>	La sfârșitul modulului plasat pe SMÎ Moodle, va fi efectuată evaluarea sumativă online. Pentru această evaluare se utilizează întrebări cu răspunsuri multiple, eseu, matching (potrivire) sau răspuns scurt.
Explicarea și interpretarea modalității de accesare a datelor și a metodelor membru în moștenire	Pe SMÎ Moodle se organizează o activitate de lucru în grup, pentru a facilita învățarea. Se configurează un wiki pentru grup, care le permite studenților să lucreze online. Discutând în colaborare și împărtășindu-și observațiile, studenții vor lucra asupra opiniilor proprii referitor la problema propusă spre discuție, pentru prezentarea ulterioară în clasă.

Instruirea la distanță și tehnologiile de e-Learning au generat o multitudine de forme și metodologii de învățare, care se dezvoltă atât independent, cât și prin implicarea instruirii tradiționale. Dezvoltarea resurselor digitale de învățare și trecerea în format digital al resurselor tradiționale duc la integrarea organică a formelor procesului didactic. Convergența progresivă dintre instruirea tradițională și cea online este arătată în figura 2.14.

Învățarea mixtă aduce beneficii indiscutabile:

- ✓ flexibilitatea procesului de predare-învățare;
- ✓ micșorarea costurilor pentru instruire;
- ✓ posibilitatea de a combina procesul de instruire cu cel de activitate profesională;
- ✓ accesul la resurse digitale de calitate, fără restricții de spațiu sau timp.

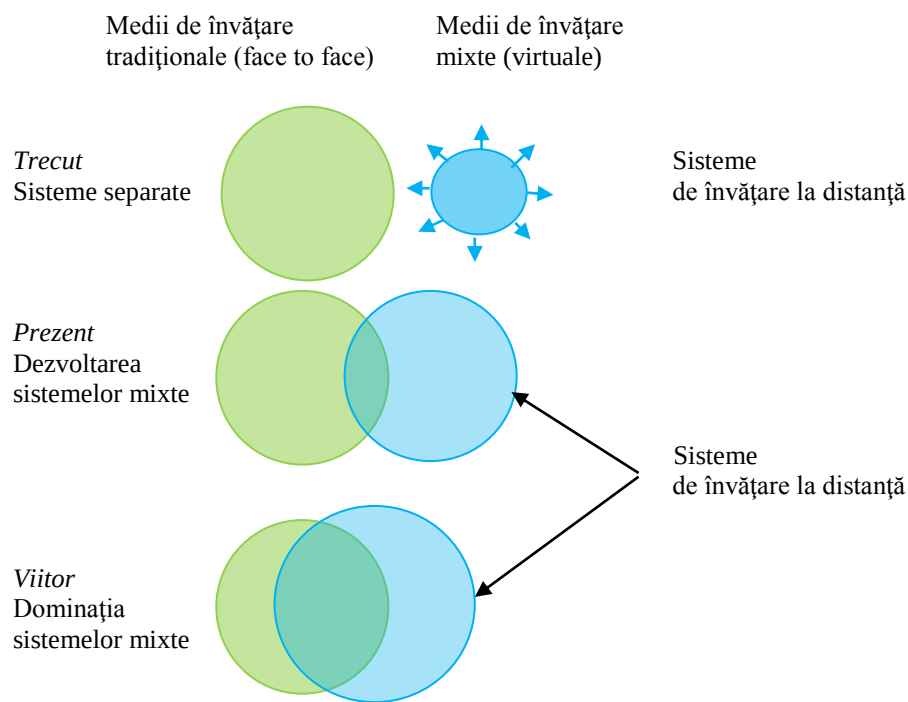


Fig. 2.14. Convergența mediilor de învățare (după Bonk, Graham [141])

După cum menționează S. Corlat [144], factorii ce determină integrarea formelor tradiționale de instruire cu cele la distanță sunt:

- *Factorul tehnologic.* Apariția unor tehnologii de comunicare calitativ noi, unor echipamente hardware educaționale sau a unor instrumente software pentru elaborarea resurselor educaționale are drept consecință accelerarea integrării metodelor tradiționale și a celor de instruire la distanță.
- *Factorul economic.* Implementarea BL este un indicator de progres al instituției de învățământ. Dar progresul necesită o modernizare calitativă a infrastructurii TIC a instituției, ceea ce implică cheltuieli considerabile.
- *Factorul uman.* Transformarea cursurilor pentru BL necesită o modificare cardinală a modului de gândire, atât pentru tutori, cât și pentru cursanți.

Activități de colaborare

Angajarea activă în procesul didactic este vitală pentru învățare. Acest adevăr se bazează pe cercetări care demonstrează că învățarea activă nu este doar mai potrivită, ci este mai îmbogățită (calitativ mai bună) atunci când studenții depășesc starea pasivă de ascultare, citire sau vizionare. Angajarea activă poate fi obținută atât prin activitate individuală, cât și prin colaborare, după cum se arată în figura 2.15.



Fig. 2.15. Învățarea activă versus învățarea pasivă [145]

Învățarea prin colaborare se bazează pe teoria „constructivismului social”. Această teorie a învățării privește modul în care învățarea individuală are loc datorită interacțiunilor din grup. Se susține că discuția dezvoltă capacitatea studenților de a-și testa ideile, de a sintetiza ideile altora și de a construi o înțelegere mai profundă a ceea ce învață. De asemenea, discuția facilitează luarea deciziilor, analiza ideilor, argumentarea și gândirea critică.

Cercetătorul M. Dougiamas [146] demonstrează că SMÎ Moodle se bazează pe teoria „constructivismului social”, deoarece oferă studenților și profesorilor libertatea de a-și construi cunoștințele prin intermediul interacțiunilor reciproce în cadrul forumurilor de discuții, al chaturilor și al altor activități de grup.

Activitatea studenților în afara sălii de clasă trebuie să implice o combinație atât a activităților individuale, cât și a celor colaborative, pentru a-i sprijini în procesul de instruire și în realizarea obiectivelor cursului.

O privire de ansamblu asupra unora dintre cele mai utilizate instrumente în activitățile studențești de colaborare și cele individuale este prezentată în tabelul 2.4.

Tabelul 2.4. Instrumente utilizate în activitățile studentești

<i>Instrumente</i>	<i>Scopul educațional</i>	
	Activitate colaborativă	Activitate individuală
Wiki	potrivit	potrivit
Blog	poate fi utilizat	potrivit
Forumuri	potrivit	
Teleconferințe	potrivit	
Alegeri		potrivit

Motivul de utilizare a noilor tehnologii în învățământul superior a fost formulat de A. Bates [147] în șase raționamente:

1. Oferirea de abilități în domeniul tehnologiei informației.
2. Răspunsul la imperativul tehnologic.
3. Extinderea accesului și sporirea flexibilității.
4. Reducerea costurilor.
5. Îmbunătățirea rentabilității educației.
6. Ameliorarea calității predării.

Noile tehnologii de predare și învățare pot fi clasificate în două mari categorii:

- a) tehnologii utilizate în principal pentru comunicarea între oameni (interacțiunea *om – om*);
- b) tehnologii utilizate în principal de către indivizi în scop propriu (interacțiunea *om – calculator*).

Prima categorie este denumită „tehnologie de comunicare”, iar a doua categorie – „tehnologie a resurselor de învățare”.

Tehnologiile de comunicare se împart în sincrone și asincrone. Tehnologiile *sincrone* sunt cele care cer ca persoanele ce comunică să utilizeze tehnologia simultan. Acestea sunt bazate pe computer, incluzând: chatul online, înregistrări audio, videoconferințe online și transmițeri online. Tehnologiile *asincrone* oferă o perioadă de timp între mesaj și răspuns. Tehnologiile asincrone bazate pe computer includ: e-mail, conferințe.

Instrumente de comunicare asincrone

Instrumentele asincrone au o importanță foarte mare în cadrul SMÎ Moodle, deoarece suportă flexibilitatea învățării.

Anunțul este o notificare de mărime mică cu caracter general, care este pus la dispoziția tuturor utilizatorilor [148]. Anunțurile pot fi filtrate în funcție de contextul utilizatorului și nu sunt adresate unor utilizatori concreți. La predarea cursului *Programarea Orientată pe Obiecte* prin SMÎ Moodle, anunțurile au fost folosite pentru a-i menține pe studenți „informați” asupra temelor de pregătit, asupra deadline-ului de prezentare a sarcinilor propuse și a altor aspecte operaționale ale cursului.

Cele mai bune cursuri online creează comunități de învățare, în care toți studenții au sentimentul că fac parte dintr-un grup prietenos, de susținere. Ei postează cu nerăbdare pe forumuri și își răspund rapid într-un mod pozitiv și productiv. Participanții la forum își împărtășesc gândurile, impresiile, iar învățarea devine distractivă și interesantă.

Forumul este și el un instrument pentru discuții asincrone [148]. Profesorul poate solicita tuturor studenților înscriși la curs să subscrie la un forum. Fiind o comunicare asincronă, studenții au timp să își formuleze mai bine răspunsul și sunt mai puțin presați de timp. Forumul de discuții este un instrument de comunicare utilizat destul de frecvent.

Obiectivul unui forum este promovarea dezbatelor prin mesaje publicate pe aceeași temă. Forumurile de discuție sunt organizate pe teme și subiecte, în care orice utilizator cu acces la forum poate posta un mesaj la un anumit subiect sau poate răspunde unui mesaj deja existent. Mesajele sunt vizibile pentru toți utilizatorii – studenți, profesori și alte categorii de utilizatori încadrați în acest curs sau modul de învățare. În general, mesajele și subiectele din cadrul unui forum sunt ordonate în funcție de dată, dar există și opțiuni pentru ordonarea lor în ordine alfabetică, după utilizator sau după subiect. Pentru fiecare grup de utilizatori pot fi stabilite diferite permisiuni și funcții. Forumurile permit anexarea de imagini, fișiere generice și linkuri.

În cadrul experimentului pedagogic desfășurat la disciplina *Programarea Orientată pe Obiecte*, pentru obținerea unor rezultate performante în utilizarea forumului a fost folosită următoarea strategie:

- Studenții au primit instrucțiuni clare despre cerințele de discuție online, termenul-limită și procedurile de organizare.
- Au fost evaluate calitatea și cantitatea postărilor online ale studenților. Folosirea tematicii le-a permis să-și creeze o imagine clară despre calitatea postărilor lor.
- A fost specificat orarul (limitele de timp) forumului de discuții.
- Profesorul trebuie să fie vizibil în discuție. Studenții sunt mult mai activi atunci când profesorul ia parte la discuție.

- Nu a fost permisă dominația discuției. Dacă studenții dominau discuția, în mod privat au fost rugați să încetinească puțin.

Folosind experiența proprie în utilizarea forumului la disciplina *Programarea Orientată pe Obiecte*, autorul sugerează câteva modalități încercate și testate, pentru a optimiza experiența forumului interactiv.

- ✓ Feedbackul la tema de pe forum trebuie furnizat cât mai rapid, iar comunicarea să fie productivă.

- ✓ Sugestiile trebuie să fie încurajatoare și pozitive.

- ✓ Întrebările postate pe forum necesită să fie legate de obiectivele de învățare.

- ✓ Studenții urmează să fie stimulați, prin intermediul forumului, de a lega materialele cursului cu experiența proprie.

- ✓ Participarea la forumuri trebuie să fie o activitate indispensabilă în clasele studenților.

Forumul poate fi folosit pentru a răspunde la întrebările studenților, dar sunt și alte modalități de a-l utiliza pentru a recunoaște munca și a ajuta studenții să dezvolte atitudinea „Eu pot să fac”. De exemplu, în cadrul cursului *Programarea Orientată pe Obiecte*, la modulul “Moștenirea”, tema “Moștenirea simplă”, pe forum autorul a lansat subiectul: ”Veniți cu exemple din lumea reală, ce demonstrează principiul moștenirii simple”. Deoarece forumul este un instrument asincron, studenții au avut timp să pregătească exemple elocvente la subiectul propus. Apoi postările au fost discutate în clasă, frontal, pentru a fi evaluate, totodată apreciind contribuția fiecărui participant la discuție. Implicându-se în discuție, studenții au dat dovadă de aptitudini de analiză și sinteză. Astfel, a fost confirmat principiul ”Eu pot să fac”, ce stă la baza competențelor OO.

SMÎ Moodle conține mai multe tipuri de forumuri. Aceste tipuri sunt ilustrate în tabelul 2.3.

Deoarece forumul este un instrument de lucru al metodei colaborative, este important de a păstra forumul foarte bine organizat, iar acest lucru va fi posibil doar dacă profesorul va crea subiecte noi. Fiecare subiect postat de profesor este un exemplu de lucru al studentului.

Forumurile sunt una dintre cele mai puternice caracteristici ale SMÎ Moodle. Ele pot fi utilizate ca blocuri de construcție, în care se pot organiza materialele de instruire pentru întregul curs. Prin urmare, este important de a nu se limita la noțiunea tradițională de forum ca un loc de discuții în grup.

Totodată, forumul poate fi folosit și pentru discuții private, “tête-a-tête”, între student și instructor. Astfel, forumul, fiind un instrument colaborativ, îndeplinește concomitent și funcția de instrument al metodelor interactive centrate pe student.

Tabelul 2.3. Tipurile de forum în SMÎ Moodle

Tipul forumului	Descriere
<i>A single simple discussion</i> <i>(O singură intervenție simplă)</i>	Întregul forum apare pe o singură pagină. Prima postare, în partea de sus a paginii, este subiectul forumului. Acest subiect este, de obicei, creat de către profesor. Studenții postează răspunsuri la acest subiect. Un forum cu un singur subiect este cel mai util pentru discuții scurte, extrem de concentrate.
<i>Standard forum for general use</i> <i>(Forum standardizat pentru uz general)</i>	Într-un forum standardizat, oricine poate începe un subiect nou. Profesorii și studenții pot crea subiecte noi și pot răspunde la postările existente.
<i>Each person posts one discussion</i> <i>(Fiecare student publică o singură intervenție)</i>	Fiecare student poate crea un singur subiect nou. Toată lumea poate răspunde la fiecare subiect.
<i>Q and A forum</i> <i>(Forum întrebare și răspuns)</i>	Acesta este un forum unic, în care profesorul creează subiectul forumului. Studenții răspund apoi la acest subiect. Cu toate acestea, un student nu poate vedea răspunsul nimănui, până când nu a trimis un răspuns. Subiectul este, de regulă, o întrebare pusă de către profesor, iar răspunsurile studenților sunt, de obicei, răspunsuri la această întrebare.

Tot în cadrul cursului *Programarea Orientată pe Obiecte*, autorul a folosit mai multe tipuri de forum: *forumul de știri*, *forumul de discuții*, *forumul confidențial*, oglindite în figura 2.16.



Fig. 2.16. Tipuri de forum utilizate la cursul POO

Scopul principal al participării studenților la forum este ca ei să însușească cerințele cursului. De aceea este important să le oferim șansa ca pe forum să practice o abilitate, să colaboreze la un proiect sau să creeze resurse pentru alte persoane.

Filosofia de bază a platformei Moodle este ideea implicării active a studenților în procesul de învățare. Forumurile le permit studenților și instructorilor să interacționeze unul cu altul. Alte activități ale Moodle dezvoltă tipuri de interactivitate, care oferă studenților posibilitatea să opteze în favoarea anumitor itemi, să-și împărtășească opiniile, să răspundă la sondaje. Instructorii pot trezi interesul studenților de a-și măsura propriul progres, de a obține feedback de la studenți, de a facilita comunicarea în cadrul grupului. Una dintre cele mai importante calități ale SMÎ Moodle este că ea cuprinde activități ce îi permit profesorului să se asigure că studenții sunt pe cale să atingă finalitățile de studii ale cursului și că îndeplinesc obiectivele individuale de învățare.

Sondajele, alegerile, chestionarele pot motiva studenții și îi pot face să se simtă membri ai unei comunități de învățare favorizantă și încurajatoare. Implementarea alegerilor și a altor sondaje interactive poate fi unul dintre cele mai populare elemente din curs. Activitatea de alegere, în mod special, poate facilita măsurarea atitudinilor și activităților studenților, încurajând astfel participarea lor la curs.

Alegerea. În Moodle, alegerea este cel mai simplu tip de activitate [149]. În această activitate se creează o întrebare și se specifică o alegere de răspuns. Se poate utiliza o opțiune pentru:

- a face un sondaj rapid;
- a cere studenților să aleagă o temă de cercetare;
- a cere confirmarea studenților cu privire la un acord.

Activitatea de alegere este, de asemenea, bună pentru selectarea ”timpului de întâlnire / programare” între studenți și profesor. Pentru a face acest lucru, este eficient de limitat numărul de răspunsuri și apoi de setat limita la fiecare alegere la 1. Fiecare alegere constituie o dată și o oră a unei întâlniri disponibile.

În cadrul cursului POO, pe SMÎ Moodle, autorul, prin intermediul activității de alegere, a propus o listă de teme de cercetare, după cum este ilustrat în figura 2.17. În această activitate, studenții au fost împărțiți în grupuri și fiecare membru al grupului a avut opțiunea de a alege o singură temă. Temele propuse spre alegere țin de întregul curs, și nu de un modul anume. Lista temelor de cercetare este anexată (Anexa 4).



Fig. 2.17. Utilizarea activității *Alegere (Choice)* în Moodle

În momentul în care tema a fost aleasă, în dreptul ei se instalează opțiunea “full”, ceea ce înseamnă că tema nu mai este disponibilă pentru a fi selectată de către alt student din grupul respectiv; în schimb, această temă este disponibilă pentru un student din alt grup.

Ulterior, profesorul vede răspunsurile oferite de studenți, urmând linkul din dreapta sus al alegerii. În fereastra ce se deschide apare câte o coloană pentru fiecare alegere, în care se află numele și imaginea studenților care au ales acea variantă (figura 2.18).



Fig. 2.18. Vizualizarea activității *Alegere (Choice)* în Moodle din perspectiva profesorului

După alegerea temei, studenții au îndeplinit sarcina prin prezentarea temei în format Power Point, care ulterior a fost uploadat pe SMÎ Moodle.

Activitatea *Alegere*, permite profesorului vizualizarea opțiunilor studenților și se poate interveni, în caz de necesitate, cu informații și ajutor suplimentar. În cazul în care studentul vrea

să-și schimbe răspunsul după ce l-a trimis, în setările activității *Alegere* este prevăzută opțiunea de acceptare a actualizării, permițându-i studentului reluarea activității de alegere până la închiderea sesiunii de lucru.

Activitatea *Alegere* în Moodle este nu numai simplă, dar și foarte flexibilă. Interfața sa este mai puțin amenințătoare decât un test, iar caracteristicile sale de publicare le permit studenților să vadă progresul unei opțiuni, spre deosebire de un sondaj. Flexibilitatea acestei activități încurajează participarea studenților, îi face să simtă că opiniile lor sunt evaluate și că sunt parte a comunității de învățare.

Activitatea de alegere poate fi utilizată, de asemenea, pentru chestionare și sondaje, care ajută la identificarea preferințelor și stilurilor de învățare, la gestionarea timpului, precum și în alte domenii de autoreglementare, ce sunt esențiale pentru succesul online. Ori de câte ori trebuie de obținut un feedback, un consens sau de efectuat un sondaj, utilizarea unei activități de alegere nu este de neglijat.

Wiki. Obiectivele de învățare la nivel de unitate contribuie la organizarea cursului. În calitate de instrument organizatoric, obiectivele de învățare trebuie să fie suficient de largi, pentru a se aplica unui ansamblu de subiecte, permițându-i studentului să organizeze informații și să găsească relații între informațiile asimilate pe parcursul unui curs. Un wiki poate fi un instrument bun întru organizarea acestor informații.

Un wiki este unul dintre cele mai flexibile instrumente de colaborare, pe care îl conține setul de instrumente de e-Learning. În Moodle, wiki deschide o gamă largă de posibilități de colaborare [150]. Un wiki poate fi folosit:

- ✓ pentru a contribui împreună la prelucrarea și stocarea informației asemănătoare enciclopediilor (cum este Wikipedia);
- ✓ pentru proiecte și prezentări în grup;
- ✓ pentru a dezvolta module personalizate de învățare;
- ✓ pentru învățarea diferențiată.

Wiki permite crearea colectivă a documentelor, cu păstrarea unui istoric al tuturor modificărilor efectuate. La un wiki se poate lucra individual sau în grup. Denumirea de *wiki* provine de la termenul hawaiian ”wiki, wiki”, ceea ce înseamnă ”foarte repede”, deoarece wiki este într-adevăr o metodă foarte rapidă de creare a documentelor în grup.

Prin wiki pot fi abordate stilurile preferențiale de învățare, organizând un wiki personalizat, diferit de celelalte.

Avantajele unui wiki [151]:

- Poate fi accesat oriunde există o conexiune web.

- Într-un proiect comun, lucrul unui utilizator poate fi partajat cu toți utilizatorii la un moment dat.
- Toate lucrările unui document sunt salvate. Dacă un document a fost modificat de mai multe ori și apare necesitatea unei variante inițiale, se poate reveni cu ușurință la original.
- Documentele sunt editate într-un mod vizibil, ceea ce adaugă responsabilitate.
- Inițial se poate plasa doar ideea, iar mai târziu, când este necesar sau când permite timpul, această idee poate fi editată.

În cadrul cursului *POO*, pe platforma Moodle a fost creat un wiki de grup la care au colaborat studenții înregistrați la curs. Wiki începe cu o pagină de start, la care studenții au adăugat alte pagini, creând legături între ele. Pagina inițială este concepută ca un cuprins, cu linkuri către alte pagini (figura 2.19).

Astfel, wiki elaborat la cursul *POO* cuprinde o tematică ce este desfășurată pe larg în modulele acestei discipline, și anume: sunt descrise succint noțiunile de *încapsulare*, *clasă*, *metodă*, *moștenire*, *polimorfism*; de asemenea, sunt prezentate noțiunile de *constructor* și *destructor*, *modificatori de acces* ș.a.m.d. Nu există o persoană care să dețină controlul editorial asupra acestui wiki.

O atenție deosebită trebuie acordată faptului ca fiecare “piesă” din wiki să fie în concordanță cu conținutul de curs sau cu obiectivul de învățare.



Fig. 2.19. Activitatea Wiki la cursul *Programarea Orientată pe Obiecte*

După utilizarea atât a forumului, cât și a unui wiki, am putea face următoarele deosebiri între ele:

1. Forumul este o activitate colaborativă în mare parte, utilizată în grup, pe când wiki este mai mult o activitate individuală. Dacă se dorește ca fiecare student să gândească și să-și

înregistreze gândurile în mod independent, un forum nu este cea mai bună alegere. În acest scop se va utiliza wiki.

2. Dacă pe forum va fi lansat un subiect cu o tematică ce cuprinde mai mulți itemi, atunci la sfârșitul cursului vom avea o colecție de postări pe forum despre acest subiect. Însă dacă se va crea o pagină wiki, atunci la sfârșitul cursului vom avea un singur document dedicat acestui subiect. Pentru producerea de cunoștințe unificate, este recomandat mai mult un wiki, decât un forum.

3. Un wiki permite și încurajează editarea repetată a unei intrări. Înscrierea crește și se schimbă odată cu schimbarea înțelegerii studenților. Într-un forum, studentul ar trebui să creeze un nou răspuns de fiecare dată când înțelegerea sa se schimbă.

În concluzie am putea afirma că un wiki este o modalitate excelentă de a ghida studenții în procesul de învățare activă și de a raporta conținutul cu finalitățile de studiu, cu obiectivele de învățare și cu evaluările.

Glosare. Deseori glosarele sunt privite ca dicționare online cu scop special. În Moodle, glosarele pot fi utilizate în mai multe moduri, prin care se pot crea activități utile multifuncționale, ce stimulează studenții să atingă obiectivele de învățare ale cursului [149]. De exemplu, pot fi create reviste conceptuale, catalogul studenților și sarcinile grupului. Prin intermediul glosarului, studenții pot dezvolta o cale de organizare a cunoștințelor, creând legături între informația modulului curent și cunoștințele anterioare.

Glosarul poate fi totodată o activitate plăcută de colaborare și un instrument de predare.

Una dintre cele mai interesante utilizări ale glosarului este de a ajuta studenții să dezvolte categorii de cunoștințe, să facă legături cu experiența lor și să elaboreze concepte. În acest scop, glosarele îi ajută pe studenți să creeze o schemă ce reprezintă cunoștințele generale. Astfel, atunci când este creat un glosar, se va crea o bază de date cu intrări individuale.

După ce a fost elaborată structura glosarelor, se dezvoltă apoi activități care promovează învățarea interactivă și colaborativă, și nume:

- Activități de identificare și definire.
- Legătura cu cunoștințele și experiența anterioară.
- Elaborarea conceptelor.

Utilizarea metodei glosarului în Moodle este utilă prin faptul că dezvoltă activități instructive care ajută la realizarea finalităților cursurilor, în special ale celor care îi impun pe studenți să identifice termenii, să-i discute și să-și aplice propriile experiențe și idei, pentru a rezolva probleme sau pentru a oferi o analiză.

Instrumente de comunicare sincrone

SMÎ Moodle permite interacțiunea în timp real, simultană, între un număr finit de utilizatori, printr-un instrument de tip sincron.

Tipurile de interacțiune de tip sincron sunt:

- *Chatul de text*, în care interacțiunea dintre utilizatori se face numai prin text scris și transmis simultan la o cameră de chat publică sau privată.
- *Chatul de voce* – interacțiune similară celei anterioare, doar că utilizatorii comunică audio.
- *Videoconferința* – cel mai complex tip, permițând comunicarea video și audio.

Calitatea unui curs online este evaluată prin calitatea interacțiunii. Capacitatea SMÎ Moodle de a oferi multiple posibilități de interacțiune conduce la ideea că pot fi elaborate cursuri de înaltă calitate, cu aplicarea diferitor strategii interactive, pentru a obține rezultatele de învățare dorite. Printre aceste activități de interacțiune se numără și chatul, care le permite participanților să se implice într-o discuție sincronă în timp real.

Chatul. Activitățile de pe *Forum* și *Chat* deseori sunt privite doar ca instrumente interactive, dar nu și ca instrumente de colaborare. În Moodle, instructorul poate organiza activitățile de discuție ca o modalitate de colaborare între studenți în timp ce aceștia își postează lucrările, citesc și discută cu colegii lor. Toate interacțiunile din Moodle implică colaborarea, cel puțin la un nivel foarte simplu. Interacțiunile pozitive în forumuri pot să-i motiveze pe studenți și să-i ajute să învețe unii de la alții. Dar deoarece în forumuri se cere ceva timp pentru a obține răspunsuri la unele întrebări, studenții preferă să comunice în timp real sau prin chat.

De fapt, pentru studenții care sunt obișnuiți să comunice aproape instantaneu cu mesaje, comunicarea asincronă a unui forum poate părea foarte lentă. Ei preferă comunicarea rapidă și instantanee. Deci, pentru cei care preferă mesaje-text, chatul este o soluție bună. În Moodle, chatul online este foarte ușor de implementat și le permite studenților să comunice între ei în timp real [148].

Un avertisment privind chatul este faptul că studenții trebuie să fie programați să participe la chat în același timp. Cu o programare atentă, această problemă poate fi depășită.

Prin chat, studenții fac schimb de idei, își trimit reciproc fișiere, postează prezentări, grafice, video etc. Pentru studenții obișnuiți cu o lume îmbunătățită prin web, chatul este ideal.

Folosind chatul, se deține un control asupra instrumentelor Moodle și se poate de contactat în mod automat persoanele care sunt înregistrate la curs. În discuțiile din chatul Moodle pot intra doar persoanele care s-au înscris la curs. Este mult mai ușor de evaluat studenții înscriși la curs și de acordat note de participare prin chat cu integrarea convenabilă a SMÎ Moodle.

Una dintre condițiile unei sesiuni de chat reușite este limitarea numărului de participanți. Este bine de organizat sesiunile în grupuri mici. Membrii unui grup nu pot participa la activitățile altui grup. Grupurile pot fi complet separate, astfel încât membrii fiecăruia să nu poată vedea niciodată munca celorlalți colegi.

În cadrul experimentului pedagogic desfășurat de autor la cursul *Programarea Orientată pe Obiecte*, plasat pe SMÎ Moodle, a fost proiectat un chat de grup, la care au colaborat studenții înregistrați la curs. Acest chat a fost folosit ori de câte ori a fost nevoie, fiind stabilită în prealabil ora de întâlnire pe chat. Problemele puse în discuție au ținut de tematica orelor de curs la disciplina POO. De exemplu, o discuție interesantă a fost inițiată pe tema *Constructorii în limbajul C++*. Unii dintre studenți vroiau să clarifice diferențele dintre constructorii cu parametru și constructorii fără parametru; care sunt valorile inițiale în primul și în al doilea caz. Studenții care au înțeles mai bine această temă au putut interveni cu ajutorul de rigoare, pentru a explica aceste diferențe. Studenții au fost asistați permanent de către autor pe parcursul duratei sesiunii de chat. În cadrul acestor sesiuni au fost abordate și teme legate de modalitatea de utilizare a instrumentelor Moodle. Sesiunile de chat erau programate atât la cererea studenților, cât și la inițiativa profesorului (autorului).

Este de remarcat faptul că în cadrul sesiunilor de chat e important de a avea întrebări de rezervă, care i-ar include în discuție pe toți studenții participanți la discuție, deoarece cei care au însușit materialul mai bine își pierd interesul de comunicare pe tema respectivă. Pentru a-i ține tot timpul în „vizzor”, este bine de „aruncat” periodic întrebări „provocatoare”.

Analogic cu forumul individual se pot programa și chat-uri private cu fiecare student.

Teleconferința este încă o modalitate bună de stimulare a creativității, de motivare și implicare a studenților în procesul de învățare [149]. Teleconferința este produsul final al unei colaborări în grup, unde studenții lucrează împreună, pentru a investiga o anumită problemă sau pentru a realiza un proiect. Fiecare difuzare poate fi prezentată întregului grup și evaluată de colegi.

Instrumente externe

Pe sistemul Moodle se pot integra și softuri externe. În acest context, pot fi menționate următoarele softuri: Wiris, Hot Potatoes, Audacity, Geogebra.

Wiris este un soft integrat în Moodle. Facilitățile oferite de Wiris evidențiază un soft cu o utilitate extraordinară, oferit profesorilor care predau discipline precum matematica (în special), fizica și chimia. Wiris completează cursurile create pentru matematică, fizică sau chimie cu elemente atractive, cu caracter științific, editate matematic și cu algoritmi ușor de elaborat. Wiris pune la dispoziția utilizatorilor trei componente: Wiris Editor (pentru editare), Wiris CAS

(pentru crearea/rezolvarea problemelor), cu varianta off-line Wiris Desktop, și Wiris Quizzes (crearea întrebărilor și testelor) [152].

Hot Potatoes a fost creat la Universitatea Victoria din Canada [153]. Cu ajutorul soft-ului Hot Potatoes, orice utilizator poate crea și publica pe Internet mai multe tipuri de exerciții ca pagini html. Cu ajutorul acestui soft pot fi create chestionare și teste, care pot fi importate în Moodle. Este un soft ușor de utilizat, atât pentru profesori, cât și pentru studenți, totodată oferind o gamă largă de exerciții.

Audacity este un editor audio gratuit de tip open source, fiind o aplicație pentru realizarea înregistrărilor [154]. Proiectul „Audacity” a fost inițiat în anul 1999 de către Dominic Mazzoni și Roger Dannenberg de la Universitatea Carnegie Mellon, fiind lansat la 28 mai 2000.

Pe lângă posibilitatea de a înregistra din multiple surse, Audacity poate fi utilizat pentru post-procesarea tuturor tipurilor de semnale audio, inclusiv pentru podcasturi prin adăugarea unor efecte, precum normalizarea și ajustarea.

Datorită faptului că Audacity este o aplicație gratuită, ea a devenit foarte populară în mediul educațional, încurajând dezvoltatorii să creeze o interfață prietenoasă pentru studenți și profesori.

Prin integrarea în Moodle a acestor componente se pot obține cursuri atractive, interesante, complete, care pot să acopere într-o proporție foarte mare cerințele curriculumului, cu o varietate mare de cerințe de învățare și cu o multitudine de instrumente utilizate pentru evaluare.

Resurse de predare-învățare-evaluare

Într-un curs tradițional, conținutul include subiectele de studiu și materialele asociate acestora, oferite prin prelegeri sau tutoriale față în față.

Utilizarea TIC permite crearea unor resurse de învățare mult mai accesibile și mult mai flexibile. Plasarea documentelor pe SMÎ Moodle este o modalitate foarte eficientă de a oferi studenților acces permanent și imediat la o gamă largă de materiale și resurse.

Digitalizarea documentelor

Pentru disciplina POO există o serie de documente care, în mod tradițional, erau puse la dispoziția studenților pe hârtie. Actualmente, digitalizarea acestor documente a condus la accesibilitatea lor oricând și oriunde.

De ce este utilă digitalizarea?

- Pentru că permite distribuirea ușoară a documentelor.
- Oferă acces oricând și oriunde pentru studenți.

- Documentele sunt ușor arhivate și stocate pentru utilizarea ulterioară.

Există multe resurse online care pot fi utilizate pentru a susține atât dezvoltarea abilităților generice ale studenților, cât și cunoașterea și înțelegerea unor subiecte specifice cursului.

Mediul Moodle susține mai multe tipuri de resurse care pot fi adăugate la curs. Astfel, la elaborarea cursului *POO* au fost folosite resursele de tip pagină Moodle, care conține elemente minime de formatare. O resursă de pagină creează un link către un ecran ce afișează conținutul creat de profesor [82]. Prin editorul de text pe pagină pot fi afișate mai multe tipuri diferite de conținut, cum ar fi text simplu, imagini, audio, video, cod încorporat sau o combinație a acestora. În anumite cazuri, este preferabil de folosit resursa paginii în locul încărcării unui document.

În SMÎ Moodle pot fi create și pagini web prin intermediul unui editor HTML încorporat. Aceste pagini se integrează în conținuturile cursului creat.

Prin adăugarea, în lista de resurse, a unei referințe către un fișier extern sau către o pagină web, în cursul instructiv creat pot fi folosite și resursele externe. Astfel de referințe au fost utilizate în cursul *POO*, plasat pe SMÎ Moodle, pentru crearea modulelor: “Noțiuni generale despre programarea orientată pe obiecte”, “Clase și Obiecte”, “Moștenirea”, “Polimorfismul”.

Un instrument foarte puternic într-un curs Moodle sunt resursele video. Clipurile video pot fi folosite pentru a îmbunătăți procesul de învățare sau pentru a-l suplini. Avantajele clipurilor constau în repetarea lecției, în cazul în care studentul fie nu a fost prezent, fie nu a înțeles materialul. Un factor deloc neglijabil este și posibilitatea de distribuire unui număr nelimitat de studenți. Materialele didactice înregistrate elimină inconvenientele de tipul timp și spațiu. Secvențele video pot fi văzute ca o modalitate de contact între tutore și cursant. În cazul lucrărilor de laborator, secvențele video pot prezenta materialele și procesele ce urmează a fi desfășurate.

Este preferabil să se încarce videoclipurile în Moodle, stocându-le pe propriul server. Atunci, însă, când limitele de spațiu pentru server sau de încărcare sunt restricționate, este convenabil de încărcat videoclipurile într-un site online, cum ar fi youtube. În prezent sunt utilizate următoarele tipuri video: Talking Head, Prezentare de slide-uri cu acoperire voce, Picture-in-Picture, Text-Overlay, Khan-Style Tablet Capture, Udasity Style Tablet Capture, Classroom Lecture, Screencast, Animation, Actual Paper/Whiteboard, Seminar, Interview, Conversation, Live Video, Webcam Capture, Demonstration, On Location, Green Screen.

Videoclipurile realizate în cadrul experimentului pedagogic pentru cursul *Programarea Orientată pe Obiecte* la temele “Crearea de obiecte”, “Utilizarea constructorilor și destructorilor”,

”Moștenirea” sunt de tipul screencast și au fost elaborate cu ajutorul aplicației FastStone Capture.

Evaluarea

Evaluarea rezultatelor instruirii reprezintă o componentă fundamentală a procesului educațional. Orice proces educațional poate fi considerat eficient atunci când este evaluat prin performanțele sale. Evaluarea rezultatelor instruirii prin identificarea performanțelor dobândite de studenți reprezintă finalitățile întregului proces educațional.

Rezultatele instruirii, stabilite prin evaluare, oferă informații despre:

- a) procesul de instruire (sistemul de metode, suporturile de instruire, conținuturile, mijloacele, activitățile de învățare etc.);
- b) modul de realizare a finalităților (atingerea competențelor dorite).

Evaluarea este inseparabil legată de proiectare, deoarece, pe de o parte, programul de instruire și educare trebuie să prevadă criteriile, indicatorii de performanță, instrumentele pentru control, iar pe de altă parte, rezultatele evaluării constituie baza reluării procesului instructiv-educativ.

Strategiile didactice interactive de predare-învățare, corelate cu cele evaluative, conduc împreună la eficientizarea activității desfășurate de profesor cu studenții.

După definiția dată în *Dictionnaire de pédagogie*, evaluarea, folosită într-un sens cât mai larg, „nu se reduce la prezentarea unei simple constatări; relevând disfuncționalitățile, ea facilitează analizele comparate, permite înțelegerea diversității situațiilor și modificarea activităților, pentru îmbunătățirea activității” [155, p. 124]. Așadar, strategiile de evaluare permit determinarea gradului de îndeplinire a finalităților propuse.

În funcție de momentul evaluării, deosebim următoarele tipuri de evaluare [156]:

- *evaluarea inițială*, care se realizează la începutul procesului pedagogic, pentru stabilirea nivelului inițial de cunoștințe al studenților. Evaluarea inițială determină următorul demers pedagogic, în scopul atingerii obiectivelor propuse și finalităților de studiu;
- *evaluarea formativă (continuă)*, ce este efectuată pe parcursul întregului program de studii, fiind organizate verificări sistematice ale tuturor celor instruiți. Are ca scop înlăturarea lacunelor și stabilește dacă sunt atinse obiectivele preconizate, pentru a urma demersul pedagogic în scopul atingerii obiectivelor finale;
- *evaluarea sumativă (finală)*, care se realizează, de obicei, la sfârșitul procesului pedagogic. Această evaluare permite stabilirea nivelului de cunoștințe/competențe obținute în rezultatul procesului didactic.

În funcție de persoana care efectuează evaluarea, deosebim trei tipuri de evaluare:

- *evaluarea internă*, efectuată de persoana care a condus activitatea de învățare;
- *evaluarea externă*, desfășurată de o altă persoană/instituție decât cea care a condus activitatea învățării;
- *autoevaluarea*, efectuată de către student pentru a stabili propriul progres.

În literatură [157] sunt aduse în discuție încă două tipuri de evaluare: cea *repetată* și cea *temporizată*.

Evaluarea repetată realizează testarea cunoștințelor în paralel cu studierea materiei noi. Acest lucru contribuie la sporirea calității și eficacității finalităților de studiu.

Evaluarea temporizată presupune verificarea cunoștințelor și competențelor rămase în memorie după un anumit timp de la studierea subiectului, cursului (perioada poate varia de la trei la șase luni sau mai mult). Această evaluare îndeplinește cerința de apreciere a eficacității rezultatului final al procesului didactic.

Distincția dintre evaluarea formativă și cea sumativă este abordată de G. Meyer ca fiind una „teoretică și neoperatorie; instrumentele utilizate de evaluarea formativă și de evaluarea sumativă sunt aceleași” [158 p. 24]

Evaluarea trebuie privită ca un proces ce promovează învățarea, și nu ca un control extern, realizat de către profesor, asupra a ceea „ce face” și „cum face” studentul. Fiind inclusă în actul de învățare, evaluarea necesită o atenție sporită, atât din partea profesorului, cât și din partea studentului, asupra proceselor de învățare, asupra cauzelor care determină erorile și asupra factorilor favorizanți ai cunoașterii.

Așadar, în actul de evaluare este important faptul cum se implică studentul în optimizarea propriei învățări. Astfel evaluarea capătă valențele unui proces reflexiv, prin care cel ce învață devine conștient de propriile acte, de propriile capacități, iar profesorul îl orientează spre finalitățile de studiu.

Strategiile de evaluare centrate pe învățare se înscriu în rândul strategiilor de evaluare interactivă. *Evaluarea interactivă* presupune un parteneriat între profesor și studenți, la bază căruia stau procese de colaborare și de negociere constructivă și care ține de responsabilitatea studentului și implicarea acestuia în procesele de învățare și evaluare. Scopurile principale ale evaluării interactive sunt: stimularea capacității de autoevaluare a subiectului implicat în proces, conștientizarea necesității acesteia și sporirea încrederii celui instruit în forțele proprii.

Un argument foarte puternic în diversitatea de abordare a problematicii evaluării îl aduce SMÎ Moodle. În acest context, pot fi menționate:

- modalitățile variate de elaborare a testelor, care se pot afișa în ordine aleatoare în test și cu date inițiale diferite pentru utilizatori diferiți;
- modalitățile de combinare și structurare a testelor;
- modalitățile de aplicare, care permit definirea timpului de lucru (durata testului) – studentul știe cât timp mai are până la finalizare, data și ora când acest test poate fi susținut;
- aspectele legate de permisiunea de a repeta susținerea testului, pentru a-și îmbunătăți rezultatele;
- diverse rapoarte pentru utilizatorii care au susținut testul și care se obțin după aplicarea unui test, de la analiza în ansamblu, până la analiza pe itemi, cu obținerea unor diferite statistici.

Testele sunt corectate automat, fiind eliminat subiectivismul ce poate afecta nota acordată. Dar în acest caz este importantă modalitatea de definire a punctelor/notelor acordate pe itemi, raportat la ansamblul testului.

Prin utilizarea instrumentelor de test online, resursele au flexibilitate de acces, astfel încât studenții le pot experimenta într-un moment convenabil, atunci când au nevoie. Un test online posedă, de asemenea, longevitate, deoarece este ușor de reutilizat de la semestru la semestru.

La elaborarea testelor formative și sumative pentru cursul *Programarea Orientată pe Obiecte*, plasate în SMÎ Moodle, au fost folosite diverse tipuri de întrebări posibile, printre care [147,159]:

- multiplă;
- de potrivire (Matching);
- alegeri sau răspunsuri (*Da / Nu, Adevărat / Fals*)
- răspuns scurt sau eseu.

Aceste teste sunt anexate în teză (Anexele 2, 3).

Sarcinile de evaluare ar trebui concepute astfel încât ele mai degrabă să fie integrate în procesul de învățare, decât să fie legate de rezultate.

Ținând cont de cele expuse anterior, am putea conchide că evaluarea prin intermediul SMÎ Moodle are avantajele de:

- a monitoriza frecvent și mai ușor progresul studenților (de exemplu, chestionare online, contribuții individuale sau de grup la un forum de discuții, sau un wiki;
- a motiva studenții să se angajeze în procesul de învățare într-o manieră continuă, prin utilizarea activităților online programate, ca parte a curriculumului cursului;
- a face ca gestionarea și administrarea evaluării să fie mai eficiente și mai precise. De exemplu, prin plasarea sarcinilor online, studenții vor lucra în funcție de dată.

Obiectivele de evaluare și de învățare sunt strâns legate între ele. Obiectivele de învățare ale cursului trebuie aliniate la sarcinile de evaluare. Pentru aceasta ar trebui luate în considerație următoarele întrebări:

- Cum vor utiliza studenții în lumea reală cunoștințele și/sau abilitățile dobândite la orele de curs?
- Ce trebuie evaluat (cunoștințe, abilități, atitudini etc.)?

Din experiența autorului, în proiectarea evaluărilor este important de a furniza suficiente oportunități, pentru ca studenții să obțină abilitățile necesare de utilizare eficientă a tehnologiei în procesul evaluării. Astfel, dacă o activitate de învățare trebuie evaluată, este bine de permis studenților să experimenteze mai întâi un exercițiu introductiv, pentru a obține competențele necesare. Sau, atunci când sarcina de evaluare implică o colaborare în grup, este necesar de luat în considerare ce anume va fi evaluat (rezultatele/produsul sau procesul de grup, sau ambele) și acest lucru va fi clarificat cu studenții.

2.3. Concluzii la capitolul 2

Evoluția fulminantă în domeniul informatic din ultimul deceniu a determinat apariția și dezvoltarea numeroaselor instrumente TIC utile procesului didactic. Implicit, a sporit și numărul Sistemelor de Management al Învățării, care au devenit tot mai familiare în domeniul educației. SMÎ susțin învățarea într-un mediu educațional bazat pe colaborare, care îmbină metodele didactice tradiționale cu metode bazate pe mijloacele TIC. Instruirea prin utilizarea Sistemelor de Management al Învățării se bazează pe o predare modernă, diferită de cea clasică, fiind mult mai atractivă, în care consolidarea cunoștințelor și evaluarea au un rol important. Obiectivul de bază al SMÎ este creșterea performanțelor celor instruiți.

În baza celor relatate, putem formula următoarele rezultate ale cercetării:

1. A fost elaborat modelul pedagogic centrat pe utilizarea Sistemelor de Management al Învățării în procesul de studiu al *Programarea Orientată pe Obiecte*.
2. Pentru implementarea modelului menționat a fost creată o metodologie care include strategii interactive, metode didactice moderne, centrate pe student și orientate spre dezvoltarea gândirii critice. Implementarea SMÎ Moodle în procesul de studiere a cursului universitar *Programarea Orientată pe Obiecte* favorizează dezvoltarea unui mediu constructivist, interactiv și integrat, centrat pe învățare, care oferă noi oportunități, determinate de un șir de avantaje: accesul la o gamă largă de abordări formative și strategii de învățare flexibile (formarea participativă, metacogniția, învățarea axată pe probleme etc.), evaluarea diversificată (autoevaluare, evaluarea

formativă, cea sumativă etc.), abordarea asincronă în timp și spațiu, conținuturi dinamice, mediu colaborativ, feedback.

3. Utilizarea SMÎ Moodle a permis extinderea colecției de resurse și tehnologii educaționale la cursul universitar *Programarea Orientată pe Obiecte* cu resurse digitale și tehnologii axate pe instrumente SMÎ.

4. A fost elaborat curriculumul la cursul universitar *Programarea Orientată pe Obiecte*, adaptat la modelul pedagogic creat.

5. Rezultatele menționate permit să constatăm că pentru soluționarea problemei cercetării rămâne de validat prin experiment didactic eficiența modelului pedagogic elaborat.

3. DEMERSURI EXPERIMENTALE ÎN IMPLEMENTAREA SISTEMELOR DE MANAGEMENT AL ÎNVĂȚĂRII ÎN PROCESUL DE STUDIU AL PROGRAMĂRII ORIENTATE PE OBIECTE

Cercetarea pedagogică este o cercetare științifică ce are ca scop explicarea științifică, înțelegerea și optimizarea activității de instruire și educare. Experimentul este o observare provocată și constă în redarea sau schimbarea intenționată a unor evenimente sau procese educaționale. El constă în analiza acțiunii unor variabile independente asupra variabilelor dependente într-o situație controlată, în vederea verificării ipotezelor cauzale. Prin urmare, în știință experimentul este folosit pentru a dovedi anumite ipoteze prin observare deliberată. După cum menționează R. Duit și M. Tesch [160], experimentul este caracteristica esențială a metodelor științifice de investigare și este strâns legat de modelarea teoretică. M. Bocoș [161, p. 3] menționează că cercetarea pedagogică este definită ca „un tip special de cercetare științifică, un proces continuu ce are ca scop explicarea, înțelegerea, optimizarea, inovarea, reformarea și prospectarea activității de instruire și educare”.

Scopul experimentului pedagogic, desfășurat în cadrul cercetării noastre, a fost de a stabili dacă utilizarea SMÎ Moodle contribuie la creșterea performanțelor studenților în studierea cursului universitar *POO*.

Obiectivele experimentului realizat au fost:

- stabilirea nivelului de pregătire al studenților pentru utilizarea SMÎ Moodle în procesul de predare-învățare-evaluare a cursului universitar *POO*;
- elaborarea materialelor electronice calitative pentru cursul universitar *POO*;
- implementarea în procesul de instruire la cursul universitar *POO* a materialelor electronice elaborate și plasate pe SMÎ Moodle;
- organizarea activităților instructiv-colaborative ale studenților pe SMÎ Moodle (prin stabilirea sarcinilor de învățare colaborativă);
- identificarea aspectelor pozitive privind utilizarea SMÎ Moodle în procesul de predare-învățare-evaluare a cursului universitar *POO* în raport cu performanțele academice înregistrate de studenți.

Variabilele de cercetare. În experimentul pedagogic au fost identificate trei categorii de variabile: variabile independente, variabile-factor și variabile dependente.

Variabilele *independente* reprezintă factorii experimentali controlați. După cum se menționează în [161, p. 43], „o variabilă este independentă atunci când nivelurile/valorile sale sunt stabilite de experimentator înainte de a începe experimentul și, deci, sunt independente de

orice se întâmplă în experiment; astfel, variabila independentă precede și influențează, potențial, măsurările realizate în cadrul experimentului”.

Variabilele independente ale experimentului pedagogic sunt :

- Grup, două categorii: experimental, de control.
- Momentul testării: testare inițială, testare finală.

Variabilele *dependente* iau valori diferite în urma influenței asupra lor a variabilelor independente. Variabilele dependente, în experiment, le-au constituit rezultatele învățării (performanțele demonstrate de studenți).

3.1. Activități preexperimentale de utilizare ale modelului POO SMÎ

În etapa inițială a cercetării noastre a fost efectuat experimentul de constatare, care a avut ca obiectiv elucidarea abilităților de utilizare a mijloacelor TIC și a atitudinii studenților vizavi de tehnologiile informaționale. În acest scop a fost elaborat și aplicat un chestionar (Anexa 1) privind gradul de utilizare de către subiecți a mijloacelor TIC. Au fost chestionați 84 de respondenți – 44 de studenți la secția cu frecvență la zi, care studiau disciplina *POO* din cadrul programelor de studii *Informatică*, *Informatică și Matematică*, *Matematică și Informatică* și *Fizică și Informatică* (ciclul de licență), Facultatea *Fizică, Matematică și Tehnologii Informaționale*, UST (gr. 2I (2015), 3IM, 3MI (2015), 3FI, 2I (2016), 3MI (2016)), 23 de studenți la secția cu frecvență redusă din cadrul UST (gr. 31I (2015), 31I (2016)) și 17 studenți la secția de zi din cadrul UPS *Ion Creangă*.

În urma analizei răspunsurilor date la întrebările chestionarului, au fost atestate următoarele rezultate:

- Tehnica modernă și învățământul centrat pe necesitățile, dorințele și posibilitățile studentului impun desfășurarea de activități diferențiate și folosirea diverselor dispozitive digitale. 77 (85%) de persoane utilizează în activitatea academică calculatorul, 68% utilizează laptopul, 80% – telefonul, 45% – proiectorul video, 46% – imprimanta și 39% – scannerul. Comunicarea prin Internet, de cele mai multe ori, este folosită pentru schimb de mesaje prin poșta electronică (87%), 64% folosesc Internetul pentru discuții prin mesagerie instantanee, aplicația Skype este folosită de 77% dintre cei chestionați, discuțiile în rețelele de socializare sunt utilizate de 64%, iar convorbirile audio și video sunt folosite în proporție de 54% și 55% respectiv. Acest lucru ne demonstrează că utilizarea dispozitivelor digitale și a mijloacelor TIC nu este un act pasiv, ci din contra, se atestă o utilizare activă a acestora

- Având în vedere faptul că implementarea TIC în procesul educațional induce necesitatea utilizării calculatorului în intervale nelimitate de timp, prin aplicarea chestionarului s-a constatat că 62% din respondenți folosesc calculatorul acasă, 35% – la lucru și 39% din numărul total folosesc calculatorul fără restricții de locație. Totodată, 29% din intervievați nu au dificultăți în utilizarea TIC în activitatea academică, 42% au dificultăți minore, 23% – dificultăți moderate și doar 6% întâmpină dificultăți mari. Posibilitatea de a accesa calculatorul acasă și fără restricții în proporție destul de mare induce ideea că subiecții își pot organiza timpul de acces la informație după necesitate. Prin urmare, plasarea cursurilor pe platformele LMS este perfectă pentru individualizarea predării-învățării-evaluării materialului didactic, iar procesul de predare-învățare este destul de accesibil.

- Utilizarea Internetului și a softurilor educaționale determină formarea unor abilități bine definite în domeniul TIC. Pentru a stabili gradul de deținere a unor astfel de abilități, subiecții au fost chestionați despre instrumentele TIC folosite în diferite activități de instruire, cu specificarea tipului de activitate. Astfel, 96% din persoanele chestionate au utilizat editoare de texte, 90% din intervievați – aplicații de prezentări electronice, 60% – platforme de învățare, și 76% au utilizat aplicații de calcul tabelar. Aceste instrumente au fost folosite pentru: elaborarea referatelor – 90%, căutarea informației în Internet – 89%, elaborarea prezentărilor – 83%, rezolvarea problemelor cu ajutorul softurilor specializate – 60%. Aici trebuie de menționat că 80% din cei chestionați au menționat că inițiativa de a folosi TIC le aparține și doar 20% au afirmat că au utilizat instrumentele TIC la îndemnul profesorilor. Luând în considerație faptul că 80% din respondenți au folosit instrumentele TIC la inițiativa proprie și tot 80% au utilizat platforme de învățare, și știind că utilizarea calculatorului și a Internetului favorizează apariția unor activități de învățare care implică procese cognitive superioare, precum dezvoltarea creativității, implicarea în procesele de cercetare, participarea la diverse dispute pe probleme științifice, am presupus că folosirea unui SMÎ la predarea-învățarea-evaluarea cursului POO va fi acceptat cu lejeritate și va avea în impact pozitiv.

- Dintre instrumentele Google utilizate de către intervievați, circa 96% au menționat Gmail-ul, 83% – Google translate, 65% – Google maps, 58% – Google drive, 39% – Google docs, 29% – Google sites, Google forms – 19%, Google books și Google calendar au acumulat respectiv 17% și 15%. Aceste abilități de utilizare a instrumentelor Google sunt salutare în cercetarea noastră, deoarece instrumentele respective se regăsesc în compartimentul „Instrumentele externe” ale modelului pedagogic prezentat în capitolul 2.

- Fiind întrebați dacă au un site sau un blog personal, 50% au răspuns că au un blog, 36% au menționat că planifică să-și creeze unul și doar 14 % din cei chestionați au susținut că nu au blog și nici nu intenționează să-și creeze. Știind că blogul face parte din instrumentele asincrone ale unui SMÎ, am rămas la ideea că utilizarea acestor platforme ar da un impuls creativității studenților.

- Este de menționat și faptul că gradele de interes „mare” și „foarte mare” al intervievaților pentru tehnologiile informaționale și de comunicație este de 85%, interesul „redus” constituie 15%, iar cele „foarte redus” și „deloc interesat” nu au acumulat nici un vot. Acest lucru demonstrează încă o dată că societatea modernă, dar mai ales generația tânără, manifestă un interes sporit față de TIC care, la rândul lor, stimulează capacitatea de învățare inovatoare, ajută indivizii să se adapteze la condițiile de schimbare socială rapidă.

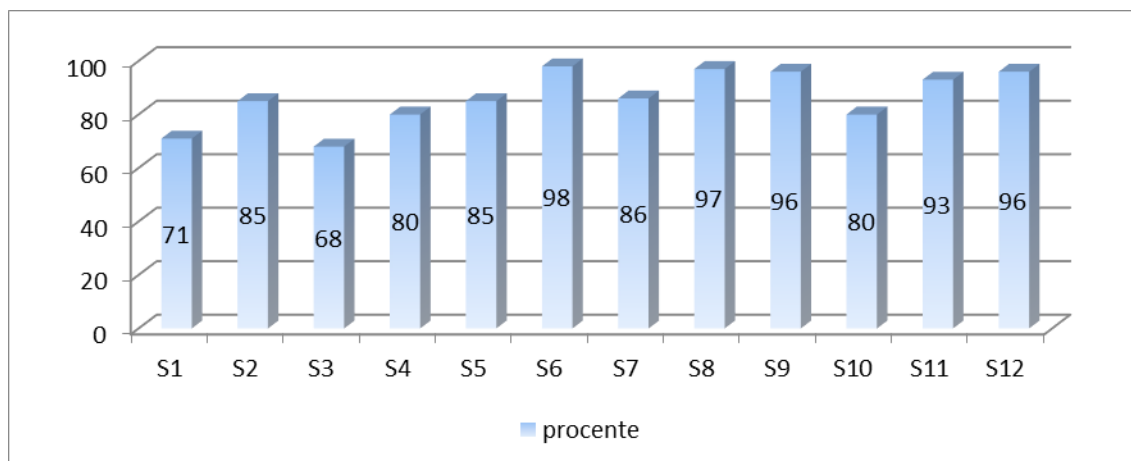
- În cazul evaluării asistate de calculator se elimină subiectivitatea umană, se poate folosi autoevaluarea, se reduce starea de stres. Acest lucru a reieșit și din răspunsurile date de către studenți în chestionar. Astfel, 93% din respondenți au menționat gradul de obiectivitate a evaluării asistate de calculator ca fiind unul „foarte obiectiv” sau „obiectiv”. Iar importanța utilizării TIC în procesul de învățare–evaluare a fost apreciată de 98% ca fiind una „foarte importantă” sau „importantă”.

Rezultatele chestionarului desfășurat asupra subiecților participanți la experimentul pedagogic este reflectat în lucrarea [162] și în mod sumar în figura 3.1.

Prin analiza răspunsurilor date la întrebările acestui chestionar am reușit să determinăm rolul TIC în procesul instructiv-educativ și să stabilim oportunitatea de utilizare a platformei SMÎ Moodle în predarea cursului universitar *POO*. Totodată, am constatat că utilizarea TIC facilitează achiziționarea unor cunoștințe și formarea unor deprinderi ce îi permit individului să se adapteze cerințelor actuale ale societății evoluționiste.

Tot în etapa de constatare a fost analizat standardul curricular la disciplina *POO*, a fost studiată Strategia de dezvoltare a educației pentru anii 2014–2020 *Educația–2020* [163] cu privire la modernizarea curriculumului universitar din perspectiva tehnologiilor informaționale și comunicaționale, a centrării pe student, cu privire la asigurarea accesului la o educație de calitate prin implementarea modelelor de învățământ la distanță, prin elaborarea și implementarea programelor de instruire pentru utilizarea TIC în educație.

În prima fază a experimentului de constatare a fost examinată și literatura de specialitate, atât cea psihopedagogică, cât și cea din domeniul platformelor educaționale și din domeniul *POO*.



S1 – gradul de dificultate în utilizarea TIC în activitatea academică (“fără dificultăți” și “dificultăți minore”)
 S2 – dispozitive digitale pe care le utilizează în activitatea academică (calculator)
 S3 – laptop
 S4 – telefon
 S5 – gradul de interes pentru TIC (“mare” și “foarte mare”)
 S6 – importanța utilizării TIC în procesul de învățare–evaluare (“foarte importantă” și “importantă”)
 S7 – posesia sau intenția de creare a blogului
 S8 – instrumentele TIC utilizate în activitatea academică (editoare de texte)
 S9 – instrumentele TIC folosite în activitatea de instruire (aplicații de prezentări electronice)
 S10 – instrumentele TIC utilizate în activitatea academică (platforme de învățare)
 S11 – gradul de obiectivitate a evaluării asistate de calculator (“foarte obiectivă” și “obiectivă”)
 S12 – instrumentele Google (Gmail)

Fig. 3.1. Gradul de utilizare a TIC de către studenții antrenați în experimentul pedagogic

În aceeași etapă de constatare am efectuat un test inițial (Anexa 2), care a evaluat nivelul de pregătire inițială a studenților antrenați în experimentul pedagogic la disciplinele informatice studiate până la acel moment. Acest test a fost aplicat pe parcursul a doi ani de studii: 2015–2016 și 2016–2017. Pentru acești ani au fost selectate eșantioanele experimentale și de control, verificând omogenitatea acestor eșantioane, iar criteriul de selectare fiind unul aleatoriu.

În anul de studii 2015–2016, în experimentul pedagogic au fost implicați 31 de studenți de la Facultatea *Fizică, Matematică și Tehnologii Informaționale*, specialitățile *Matematică* și *Informatică*, UST: 17 studenți au format eșantionul de control (grupele 3FI și 3MI) și 14 studenți – eșantionul experimental (grupele 3IM și 2I).

În anul 2016–2017, în experiment au fost antrenați și 23 de studenții de la secția cu frecvență redusă, grupa 31I (anul de studii 2015–2016) și grupa 31I (anul de studii 2016–2017), Facultatea *Fizică, Matematică și Tehnologii Informaționale*, specialitatea *Informatică*: 11 studenți în eșantionul de control și 12 studenți în eșantionul experimental.

În anul de învățământ 2016–2017, experimentul a fost desfășurat atât la UST, cât și la UPS *Ion Creangă* pe un grup de 30 de studenți, secția cu frecvență la zi: eșantionul experimental a fost format din 13 studenți din grupele 2I și 3MI ale Facultății *Fizică, Matematică și Tehnologii Informaționale*, specialitățile *Matematică* și *Informatică*, UST, și 17 de la UPS *Ion Creangă*, care au constituit eșantionul de control. Reprezentarea numerică a studenților pentru fiecare eșantion este precizată în tabelul 3.1.

În scopul prelucrării statistice a datelor experimentale a fost folosită aplicația SPSS (Statistical Package for the Social Sciences), una dintre cele mai utilizate programe statistice pentru analiza datelor în științele sociale. Aplicația SPSS reprezintă un pachet de programe interactiv, dedicat analizelor de date, ce conține multiple facilități și tehnici statistice. Acest pachet oferă modalități de prezentare a rezultatelor conform standardelor APA (Asociația Psihologilor Americani) [164] – forul cu prestigiul cel mai înalt în domeniul științelor sociale. În afară de analizele statistice posibile, programul are componente puternice pentru managementul și documentarea datelor.

Tabelul 3.1. Numărul de studenți implicați în experimentul pedagogic

Anul de studii	Eșantionul experimental (numărul de studenți)	Eșantionul de control (numărul de studenți)
2015–2016	14 (UST)	17(UST)
2016–2017 (secția cu frecvență redusă)	12 (UST)	11 (UST)
2016–2017 (secția cu frecvență la zi)	13 (UST)	17 (UPS)
Total	39	45

Pentru început am creat baza de date, definind variabilele și proprietățile lor (figura 3.2), coplețând-o ulterior cu valorile respective. Variabilele pot fi cel puțin de patru tipuri [165]:

- *Nominale* – sunt caracteristici sub formă de nume sau simboluri, care nu pot fi ordonate între ele.
- *Ordinale* – sunt acele caracteristici care presupun o ordine naturală (intrinsecă) a valorilor.
- *De intervale* – variabile ordinale, dar cu diferență egală între valori.
- *De rapoarte* – variabile pur numerice.

	Name	Type	Width	Decimals	Label	Values	Missing	Columns	Align	Measure	Role
1	Id_pers	String	24	0	Nume_P	None	None	8	Left	Nominal	Input
2	Initial	Numeric	8	2	T_initial	None	None	8	Right	Scale	Input
3	eval_1	Numeric	8	2	Test_1	None	None	8	Right	Nominal	Input
4	eval_2	Numeric	8	2	Test_2	None	None	8	Right	Nominal	Input
5	examen	Numeric	8	2	examen	None	None	8	Right	Nominal	Input
6	Nota_fin	Numeric	8	2	Nota_finala	None	None	8	Right	Nominal	Input

Fig.3.2. Variabilele bazei de date în SPSS

Eșantioanele de control și experimental sunt independente, deoarece un subiect participant la experiment a fost inclus doar într-un singur eșantion (experimental sau de control), excluzând posibilitatea de selectare a acestui element în alte eșantioane.

Inițial au fost calculați indicatorii statistici de bază pentru fiecare eșantion implicat în experiment (anii 2015–2016 și 2016-2017). Rezultatele sunt reflectate în Anexele 6, 7, 8 și în tabelul 3.2.

Tabelul 3.2. Indicatorii statistici de bază, calculați pentru fiecare eșantion din experiment

Anul	Eșantionul	Numărul de subiecți (n)	Media (m)	Abaterea-standard (s)
2015-2016	experimental	14	6,30	1,40712
	de control	17	6,22	1,14114
2016-2017, frecvență la zi	experimental	13	6,12	0,60160
	de control	17	6,04	0,72376
2016-2017, frecvență redusă	experimental	12	6,85	1,42092
	de control	11	6,60	0,67676

Atunci când a fost aplicat testul inițial, s-a urmărit un nivel de pregătire apropiat al subiecților din eșantioanele experimental și de control. Pornind de la ideea că lotul experimental și cel de control, la începutul cercetării noastre, au avut același nivel de pregătire, am formulat următoarele ipoteze de cercetare:

H_0 : $m_1=m_2$ – nu există diferențe semnificative între media eșantionului experimental și media eșantionului de control;

H_1 : $m_1 \neq m_2$ – există diferențe semnificative între media eșantionului experimental și media eșantionului de control.

Pentru a verifica dacă există diferențe semnificative între mediile eșantioanelor experimental și de control, am aplicat testul **t-Student**. Acest test este utilizat pentru a testa ipoteza nulă precum că mediile a două populații sunt aceleași, $H_0: m_1=m_2$ atunci când este disponibil un eșantion de observații din fiecare populație.

Condițiile aplicării testului t-Student pentru eșantioane independente sunt [166, p. 97]:

- 1) independența grupurilor – fiecare subiect face parte doar dintr-un grup, iar aceste grupuri sunt independente;
- 2) variabila dependentă este cantitativă, măsurată pe scale de interval sau proporții;
- 3) variabila dependentă este normal distribuită;
- 4) omogenitatea varianțelor – grupurile trebuie să facă parte din populații cu varianțe egale.

Asupra datelor obținute în cadrul experimentului a fost aplicat testul t-Student. Rezultatele furnizate de softul SPSS, pentru eșantioanele implicate în experiment, anul de studii 2015–2016, sunt prezentate în tabelul 3.3.

Tabelul 3.3. Rezultatele testului t-Student, anul de studii 2015–2016, grupele cu frecvență la zi

T_initial	Levene's Test for Equality of Variances		t-test for Equality of Means			
	F	Sig.	t	df	Sig. (2-tailed) p	Std. Error Difference
Equal variances assumed	0,460	0,503	-0,154	29	0,878	0,45737
Equal variances not assumed			-0,151	24,949	0,881	0,46694

Din tabelul 3.3 se observă că valoarea testului Levene de egalitate a varianțelor este nesemnificativă: $F(29) = 0,460$ cu pragul de semnificație $0,503 > 0,05$. Prin urmare, se constată că varianțele sunt egale ($s_C = 1,14114, s_E = 1,40712, s_C \approx s_E$), iar rezultatele se citesc din linia de sus (*Equal variances assumed*) a tabelului. Rezultatul testului t-Student este: $t = -0,154$ la un prag de semnificație $p = 0,878 > 0,05$. Comparând t cu $t_{cr} = 2,045$ (Anexa 4 [167, p. 358]), obținem: $t < t_{cr}, 0,154 < 2,045$. Astfel, se poate afirma că se menține ipoteza nulă H_0 , adică nu există diferențe semnificative între mediile eșantioanelor experimental și de control.

Pentru datele experimentale din anul de studii 2016–2017, grupele cu frecvență la zi, de asemenea a fost aplicat testul t-Student, iar rezultatele sunt reflectate în tabelul 3.4.

Tabelul 3.4. Testul t-Student, anul de studii 2016–2017, grupele cu frecvență la zi

T_initial	Levene's Test for Equality of Variances		t-test for Equality of Means			
	F	Sig.	t	df	Sig. (2-tailed) p	Std. Error Difference
Equal variances assumed	0,093	0,763	-0,330	28	0,744	0,24837
Equal variances not assumed			-0,338	27,759	0,738	0,24218

Deoarece deviația-standard, calculată pentru eșantioanele de control și experimental, este $s_C = 0,72376$, $s_E = 0,60160$, se constată că varianțele sunt egale: $s_C \approx s_E$, fapt confirmat și de valoarea testului Levene, care este nesemnificativă statistic: $F(28)=0,093$ cu pragul de semnificație $0,756 > 0,05$. Rezultatul testului t-Student, afișat în tabelul 3.4, este $t = -0,330$. Din Anexa 4 [167, p. 358] citim $t_{cr} = 2,048$, pentru $p < 0,05$ și $df = 28$. Constatăm că $t < t_{cr}$, $0,330 < 2,0048$ și $p = 0,744 > 0,05$. Așadar, concluzia este că și pentru aceste două eșantioane se menține ipoteza nulă.

Testul t-Student a fost aplicat și în eșantioanele de control și experimental din anul de studii 2016–2017, grupele cu frecvență redusă, pentru a identifica existența sau lipsa diferențelor dintre mediile acestor două eșantioane. Rezultatele obținute sunt reflectate în tabelul 3.5.

Tabelul 3.5. Testul t-Student, anul de studii 2016–2017, grupele cu frecvență redusă

T_initial	Levene's Test for Equality of Variances		t-test for Equality of Means			
	F	Sig.	t	df	Sig. (2-tailed) p	Std. Error Difference
Equal variances assumed	3,833	0,064	-0,548	21	0,590	0,47146
Equal variances not assumed			-0,564	16,038	0,581	0,45813

Deoarece testul Levene, reflectat în tabelul 3.5, este nesemnificativ – $F(21) = 3,833$, cu pragul de semnificație $0,064 > 0,05$, putem face concluzia că se asumă varianțe egale. Din tabelul 3.5, $t = -0,548$ și $p = 0,590 > 0,05$. Din Anexa 4 [167, p. 358] citim: $t_{cr} = 2,080$. Observăm că, $t < t_{cr}$, $0,548 < 2,080$. Prin urmare, și în acest caz nu sunt diferențe semnificative între mediile eșantioanelor experimental și de control.

Pentru confirmarea rezultatelor obținute prin aplicarea testului t-Student, se va aplica **testul neparametric Mann-Whitney**.

Calea de parcurgere a testului Mann-Whitney pusă la dispoziție de aplicația SPSS este redată în figura 3.3. Fișierele de input și output sunt identice cu cele utilizate pentru aplicarea testului t-Student.

Pentru a aplica testul Mann-Whitney, inițial se selectează variabila asupra căreia se va aplica acest test. În cazul nostru, variabila este $T_initial$, iar variabila după care se face gruparea este *Grup-eșantion*.

Selectarea variabilelor asupra cărora se va aplica testul Mann-Whitney este reflectată în figura 3.4.

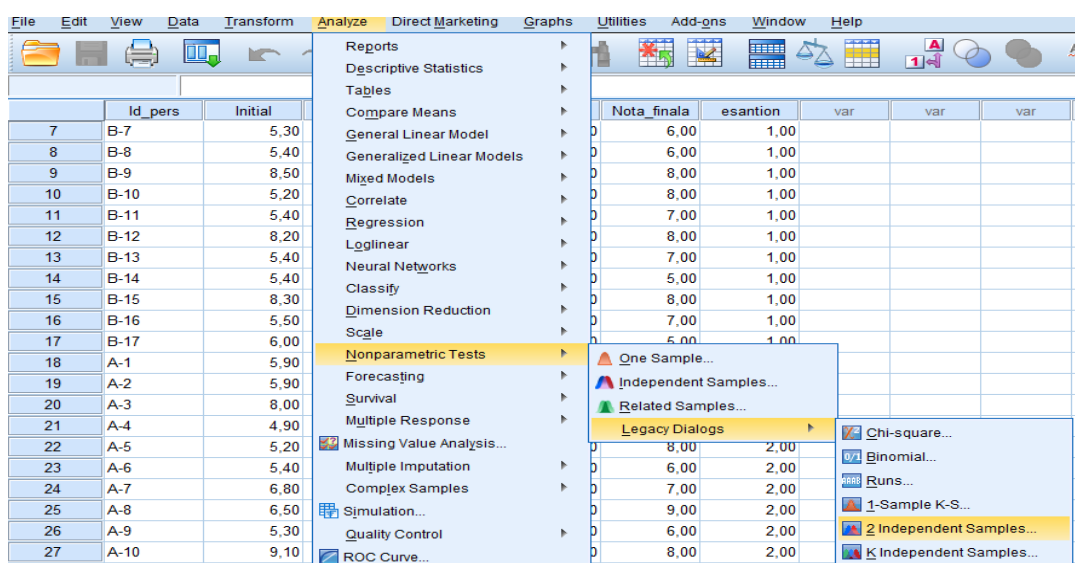


Fig. 3.3. Calea pentru aplicarea testului Mann-Whitney în SPSS pe doua eșantioane independente

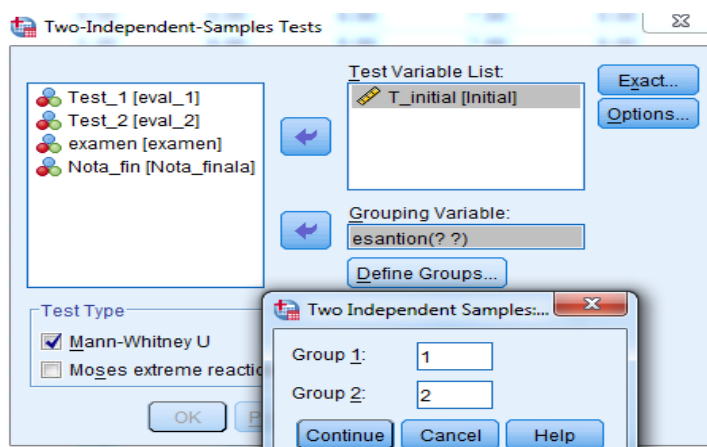


Fig. 3.4. Selectarea variabilelor pentru aplicarea testului Mann-Whitney

Rezultatele aplicării testului Mann-Whitney pe eșantioanele de control și experimental, anul de studii 2015–2016, grupele cu frecvență la zi, sunt afișate în tabelul 3.6.

Tabelul 3.6. Testul Mann-Whitney, anul de studii 2015–2016, grupele cu frecvență la zi

T_initial	Eșantion de control	Eșantion experimental
<i>n</i>	17	14
Mean Rank	15,74	16,32
Sum of Ranks	267,50	228,50
Mann-Whitney U	114,000	
Asymp. Sig., <i>p</i>	0,857	

Pentru anul de studii 2015–2016, datele obținute cu ajutorul softului SPSS, pentru eșantioanele implicate în experimentul pedagogic, sunt: $\sum R_C = 267,50$ și $\sum R_E = 228,50$, $U = 114$, $p = 0,857 > 0,05$. Comparăm U cu valoarea critică $U_{cr} = 67$, din Anexa 8 [167, p. 364], și obținem $U > U_{cr}$. Prin urmare, și testul Mann-Whitney ne confirmă justetea ipotezei nule H_0 .

Aplicăm testul Mann-Whitney pe eșantioanele implicate în experimentul pedagogic din anul de studii 2016-2017, secția cu frecvență la zi. Rezultatele obținute sunt oglindite în tabelul 3.7.

Tabelul 3.7. Testul Mann-Whitney, anul de studii 2016–2017, secția cu frecvență la zi

T_initial	Eșantion de control	Eșantion experimental
<i>n</i>	17	13
Mean Rank	14,82	16,38
Sum of Ranks	252,00	213,00
Mann-Whitney U	99,000	
Asymp. Sig., <i>p</i>	0,630	

Pentru aceste eșantioane, $U = 99 > U_{cr} = 63$ la un prag de semnificație $p = 0,630 > 0,05$. Aceste date demonstrează faptul că nu există diferențe semnificative între mediile acestor două eșantioane. Astfel, se confirmă ipoteza nulă.

Pentru eșantioanele din anul academic 2016–2017, secția cu frecvență redusă, valorile calculate în SPSS sunt afișate în tabelul 3.8: $\sum R_C = 122,00$ și $\sum R_E = 154,00$, $U = 56$, $p = 0,537 > 0,05$.

Tabelul 3.8. Testul Mann-Whitney, anul de studii 2016–2017, secția cu frecvență redusă

T_initial	Eșantion de control	Eșantion experimental
<i>n</i>	11	12
Mean Rank	11,09	12,83
Sum of Ranks	122,00	154,00
Mann-Whitney U	56,000	
Asymp. Sig., <i>p</i>	0,537	

$\sum R_C = 122,00$ și $\sum R_E = 154,00$, $U = 56$, $p = 0,537 > 0,05$. Analizând rezultatele obținute pentru $U_{cr} = 33$, constatăm că $U > U_{cr}$, fapt ce confirmă și menține ipoteza nulă. Prin urmare, am demonstrat faptul că nu există diferențe semnificative între mediile acestor două eșantioane.

3.2. Valorificarea experimentală a modelului POO SMÎ

Formularea *ipotezei* cercetării a pornit de la observațiile că prin implementarea cursului POO pe SMÎ Moodle se vor propune studenților situații de învățare care vor crește performanțele lor și interesul pentru această disciplină.

Eșantionul de participanți. Experimentul de formare a fost desfășurat pe parcursul a doi ani de studii: 2015–2016 și 2016–2017.

În anul 2015–2016, procesul didactic la disciplina POO, atât în eșantionul experimental (14 studenți), cât și cel de control (17 studenți), a fost dirijat de autor, lecțiile teoretice fiind predate în serii comune pentru ambele eșantioane. Lecțiile de laborator ale cursului POO în eșantionul experimental au fost predate având la bază modelul pedagogic elaborat prin implementarea SMÎ Moodle în procesul didactic.

În cadrul experimentului pedagogic au fost aplicate următoarele metode:

- *Metoda testelor*, care a fost folosită în etapa inițială a experimentului pentru stabilirea nivelului inițial al studenților, în etapa desfășurării experimentului pedagogic pentru verificarea realizării obiectivelor și în etapa finală pentru a stabili rezultatele obținute de către studenți și eficiența modului de predare.
- *Metoda problematizării și învățarea prin descoperire*, în care studentul, având un rol principal, activ, își activează intens mecanismele cognitive, în special prin mecanismele disonanței cognitive. Această metodă dezvoltă abilitățile de cooperare în situații de grup.
- *Metoda instruirii reciproce (peer-to-peer instruction)*, unde studenții, utilizând forumul și chatul, au participat activ la lecții și au împărtășit cunoștințele lor cu alți studenți.

Această metoda de predare a fost extrem de eficientă, în special la orele de laborator, sporind gradul de învățare și de interes față de cursul POO.

- *Metoda clasei inversate* este o metodă în care elementele tipice ale cursului sunt inversate. Studenții studiau materialul la domiciliu, înaintea sesiunii de lucru, iar timpul orelor de curs este dedicat exercițiilor, proiectelor sau discuțiilor. Astfel, lucrul în clasă a fost transformat într-o interacțiune în stil atelier, în care studenții puneau întrebări asupra conținutului lecturii, își testau aptitudinile în aplicarea cunoștințelor dobândite și desfășurau activități practice (rezolvarea de probleme). În timpul sesiunilor de curs, studenții au fost încurajați să-și dezvolte aptitudinile de cercetare individuală și abilitățile de colaborare.

- *Metoda studiului de caz*, folosită în experimentul didactic, a avut ca scop angajarea activă și interactivă a studenților în desfășurarea activităților de analizare și dezbateri colectivă a unui „caz”, încurajând învățarea prin cooperare și colaborare. Această metodă i-a încurajat pe studenți să manifeste o atitudine critică față de diferite variante de soluționare a cazului și să-și dezvolte capacitățile decizionale.

Utilizând SMÎ Moodle în timpul lecțiilor, studenții au avut posibilitatea de a revizui temele învățate anterior și oportunitatea de a-și aprofunda cunoștințele. De asemenea, studenții au manifestat interes pentru prelucrarea materialului nou. Prin utilizarea mijloacelor oferite de SMÎ Moodle s-a contribuit la îmbunătățirea creativității studenților, antrenându-i activ în crearea materialului studiat prin prezentări și Wiki. Promovând predarea constructivistă, autorul a încurajat studenții să participe activ în dezvoltarea cursului, folosind wiki. La lecțiile de laborator, studenții aveau posibilitatea de a urmări metodele de rezolvare a problemelor de POO la diferite module, prin accesarea fișierelor în format video, create de autor și plasate pe Moodle. Acest lucru oferea studenților libertatea de a studia materialul predat și modalitățile de programare orientată pe obiecte ori de câte ori era nevoie.

Aceste metode au fost aplicate de autor pe parcursul desfășurării experimentului pedagogic, pentru a dezvolta anumite competențe, în special competența-cheie de *a învăța să înveți* (*Learning to learn*), care face parte din cele opt domenii de competențe definite de Comisia Europeană și recomandate de Parlamentul European [168]. Competența de a învăța să înveți reprezintă abilitatea de a se implica și de a persevera în învățare, de a organiza propria învățare, permițând individului să gestioneze și să reglementeze eficient timpul și informația. Aceasta include conștientizarea procesului individual de învățare, identificarea oportunităților disponibile și a abilităților de a depăși obstacolele pentru realizarea unei învățări de succes; de a dobândi și a asimila noi cunoștințe, de a aplica aceste cunoștințe în diferite domenii. Utilizarea

acestor metode a fost facilitată prin implementarea tehnologiilor SMÎ Moodle în predarea cursului *POO*.

Pentru motivarea studenților și implicarea lor activă în procesul de instruire, s-a decis aplicarea **metodei proiectelor**, descrisă în capitolul 2. Pentru rezolvarea unor probleme complexe este bine de recurs la metoda proiectelor, care implică o participare colaborativă activă a studenților [169]. Prin participarea la proiecte, studenții realizează importanța elaborării unui cod de program eficient, precum și necesitatea de a efectua comentarii explicite la codul de program.

La lecțiile de laborator, studenților li s-a propus să dezvolte o bibliotecă de clase – geometria. Această bibliotecă trebuie să conțină obiecte cum ar fi: cerc, trapez isoscel, paralelogram, triunghi, pătrat, dreptunghi și așa mai departe. Sarcina de bază a proiectului a fost descompusă în subsarcini: crearea de clase separate. Distribuirea sarcinilor a fost efectuată în concordanță cu nivelul de pregătire al fiecărui student în baza unui test de verificare a cunoștințelor la disciplinele studiate anterior și a testului inițial (crearea de obiecte constituie tematica primelor lecții de laborator). Studenții cu rezultate mai bune au primit sarcini complexe. În această etapă s-au utilizat intens așa instrumente ale SMÎ Moodle precum forumul și chatul, pentru a concretiza sarcinile și a face schimb de informație între profesor–studenți și studenți–studenți. Lista metodelor obligatorii din clase a fost aprobată de către profesor, iar fiecare student a stabilit în mod independent atributele și metodele specifice clasei.

În etapa inițială, o lecție de laborator a fost axată pe explicarea modului de muncă în echipe, pe organizarea proiectelor și explicarea metodelor eventual necesare pentru aplicare în cadrul proiectului, cum ar fi: metoda de colectare a informației, metoda de analiză și cea de gândire critică. Proiectul a fost planificat pe o perioadă de trei săptămâni. În fiecare săptămână, la orele de laborator, se repartizau 20 de minute pentru a discuta problemele de lucru ale proiectului, problemele de programare și modul de utilizare a metodelor. Întâlnirile pe forum și chat în Moodle erau programate de două ori în săptămână la ore fixe, stabilite în prealabil. Pentru a se asigura că toți studenții primesc aceleași informații, a fost creat un serviciu de asistență, unde toate întrebările erau afișate împreună cu răspunsurile. Am efectuat acest lucru deoarece am dorit ca studenții să aibă acces la informații cât mai repede posibil și să reducă la minimum decalajul de informații.

Următoarea etapă a implementării metodei proiectului a fost testarea claselor, obținute în rezultatul elaborării de către studenți a sarcinilor individuale. În această etapă, lucrul efectuat a fost evaluat după mai multe criterii: termenul-limită, caracterul complet al descrierii clasei, calitatea comentariilor la codul de program etc. Pentru eficientizarea testării codului, studenții au

făcut schimb de programe elaborate, prin intermediul e-mailului sau mesajelor prin chat ori forum. În cadrul acestei etape, studenții își formează abilități de testare. Ultima etapă a proiectului a fost asamblarea tuturor claselor elaborate în bibliotecă și testarea bibliotecii create. În cadrul acestei etape, studenții au primit sarcina de a scrie metodele lipsă pentru orice clasă. Astfel, pe parcursul derulării întregului proiect au fost folosite intens instrumentele de colaborare Moodle.

Pe lângă sarcinile pregătite pentru fiecare student, autorul a folosit sistemul Moodle pentru crearea și aplicarea testelor, ceea ce a provocat interesul studenților pentru revizuirea lecțiilor studiate anterior și a sporit interesul față de conținutul noilor unități de lecții. Studenții au acceptat cu plăcere utilizarea Moodle, deoarece li s-a permis accesul suplimentar la conținutul cursului *POO*, creat de autor, în orice moment și în orice loc, plus condițiile moderne de lucru, care au constituit un factor deloc neglijabil.

Prin intermediul chatului creat pe SMÎ Moodle, a fost realizat feedbackul cu studenții implicați în experiment. Lucrările de laborator și sarcinile individuale au fost elaborate în format electronic și expediate profesorului, utilizând aplicația *Google mail*. În cadrul lucrului individual, fiecare student a prezentat produsele elaborate, în format electronic, demonstrând gradul de implicare în realizarea lor. Portofoliul prezentat la sfârșitul semestrului de către fiecare student a constatat din lucrările de laborator și sarcinile individuale, acesta fiind lucrul său individual.

Pe durata predării cursului *POO* au avut loc două evaluări curente sumative, realizate sub formă de test în sistemul Moodle pentru grupele experimentale. Înainte de testare, studenții au fost instruiți de către profesor asupra particularităților de parcurgere a testelor în SMÎ Moodle. Aceste evaluări au acoperit conținutul curriculumului la disciplina *POO* în totalitate, accentul fiind pus pe testarea competențelor specifice, formate atât în cadrul orelor de curs, cât și în cadrul orelor de laborator.

Media aritmetică a notelor înregistrate de student la aceste două evaluări plus lucrul individual al studentului a format 60% din nota finală la curs. Nota de la evaluarea finală (testul final) a constituit 40% din nota finală.

În scopul respectării corectitudinii experimentului, autorul a aplicat aceleași teste de evaluare sumativă și testul final (în format imprimat) în grupul de control. În urma unei analize calitative a rezultatelor obținute în prima etapă a experimentului de formare (anul de studii 2015–2016), s-a intervenit cu unele ajustări în modelul pedagogic elaborat.

Deoarece în etapa următoare se preconiza aplicarea modelului pedagogic și a metodologiei elaborate și la secția cu frecvență redusă, unde aplicarea metodelor colaborative sunt mai eficiente, am venit cu ideea de a intensifica utilizarea forumului și chatului.

A doua etapă a experimentului de formare a demarat în anul de studii 2016–2017, la care au participat 12 studenți de la secția cu frecvență redusă, anul de studii 2015–2016, care a constituit eșantionul experimental, și 11 studenți, la fel de la secția cu frecvență redusă, anul de studiu 2016–2017, care a format eșantionul de control, ambele eșantioane făcând parte din cadrul Facultății *MFTI* a UST, cu scopul de a demonstra eficacitatea modelului elaborat și a metodologiei propuse, în condițiile instruirii la distanță. O particularitate a utilizării SMÎ Moodle în grupele cu frecvență redusă a fost folosirea sporită a forumului și a chatului în eșantionul experimental.

Tot în acest an de studii au fost antrenați în experiment încă 13 studenți de la secția cu frecvență la zi din cadrul UST și 17 studenți de la secția cu frecvență la zi din cadrul UPS *Ion Creangă*, pentru a demonstra repetat eficacitatea modelului elaborat.

3.3. Validarea rezultatelor experimentale

După implementarea modelului pedagogic creat și a metodologiei propuse, au fost colectate mai multe serii de date: rezultatele (notele) studenților la două teste de evaluare sumativă și notele de la evaluarea finală. Aplicând metodologia propusă de Ministerul Educației al Republicii Moldova privind calcularea notei finale la o unitate de curs, a fost calculată nota finală pentru fiecare student la unitatea de curs *Programarea Orientată pe Obiecte*. Analiza statistică a fost efectuată în baza notelor finale obținute de studenți la disciplina *POO*.

În figura 3.5 este reprezentată analiza comparativă a mediilor înregistrate de studenți la două evaluări sumative și nota finală la *POO* pentru eșantionul experimental și cel de control din experimentul pedagogic realizat pe parcursul a doi ani de studii: 2015–2016 și 2016–2017.

În scopul elucidării unor legități sau dependențe, colecția de date obținută a fost supusă unor operații de clasificare, ordonare, analiză și interpretare [170, p. 136].

Asupra testelor de evaluare sumativă și testului final pentru fiecare eșantion (experimental și de control), autorul a aplicat atât analiza statistică cantitativă, cât și cea calitativă.

Astfel, analiza statistică a fost realizată prin calcularea valorilor testului parametric *t*-Student, testului neparametric Mann-Whitney, indicatorilor mărimii efectului, indicelui de asimetrie Fisher, testului Wilcoxon și testului Pearson [171, p. 15-24].

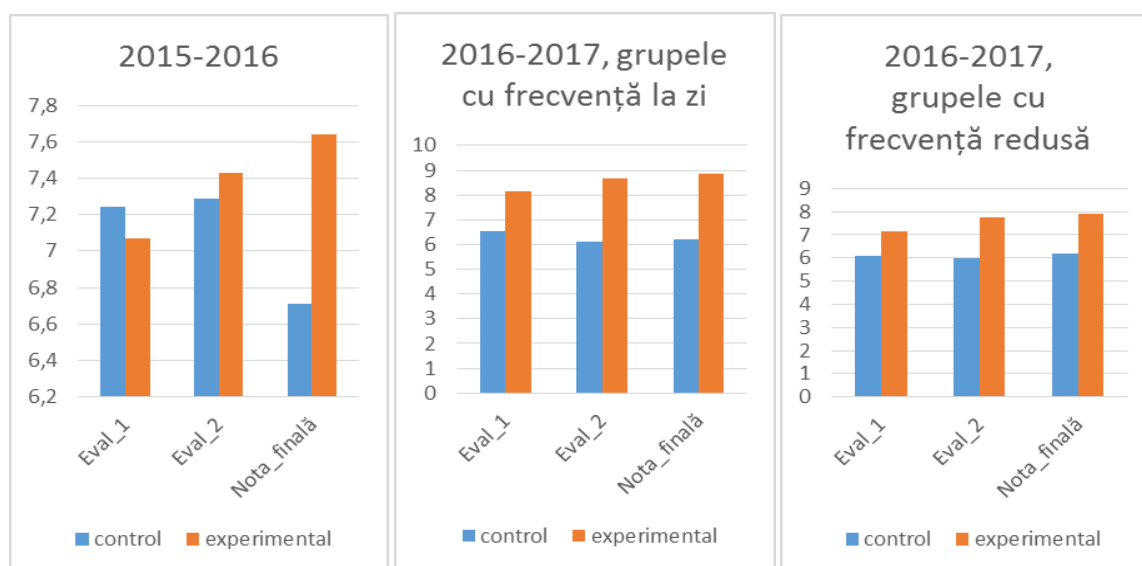


Fig. 3.5. Analiza comparativă a mediilor înregistrate de studenți la două evaluări sumative și nota finală la POO pentru eșantionul experimental și cel de control

Indicatorii statistici pentru fiecare eșantion implicat în experiment (anii de studii 2015–2016 și 2016–2017), utilizând notele finale la unitatea de curs *POO*, sunt reflectate în tabelul 3.9.

Tabelul 3.9. Indicatorii statistici calculați pentru fiecare eșantion implicat în experimentul pedagogic

Anul	Eșantion	Număr de subiecți	Media	Abaterea-standard
2015–2016	EE	14	7,6429	0,92878
	EC	17	6,7059	1,10480
2016–2017, frecvență la zi	EE	13	8,8769	0,72819
	EC	17	6,2235	0,73103
2016–2017, frecvență redusă	EE	12	7,9167	1,25831
	EC	11	6,1818	0,75076

Pentru a vedea distribuția notelor de la cele două teste sumative și de la testul final ale subiecților din eșantioanele implicate în experimentul pedagogic, folosim distribuția de frecvențe. Reprezentarea grafică a unui tabel de frecvențe absolute se va face cu ajutorul unei histogramme prin intermediul softului SPSS, unde pe axa verticală sunt reprezentate frecvențele, iar pe cea orizontală – valorile variabilei analizate.

În figura 3.6 sunt reflectate acțiunile necesare pentru construirea histogrammei de distribuție în aplicația SPSS.

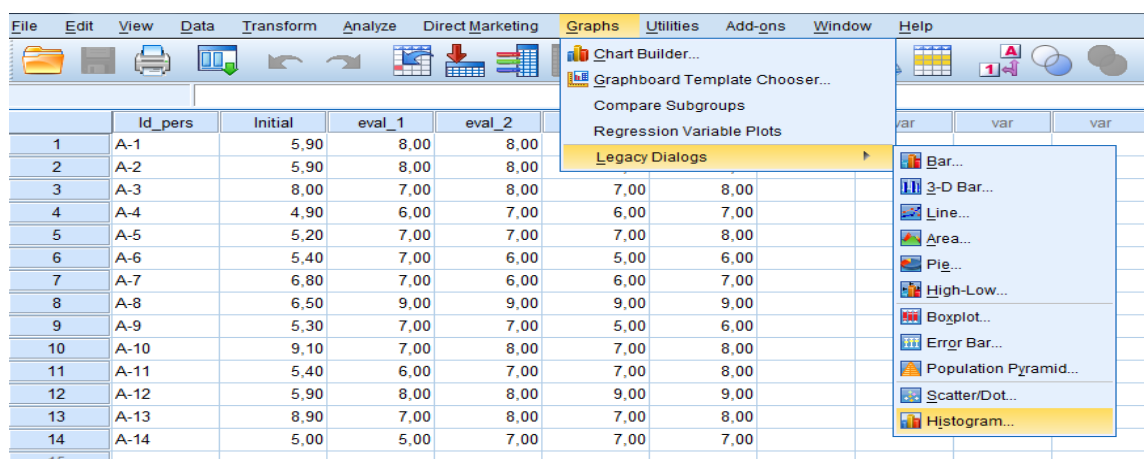
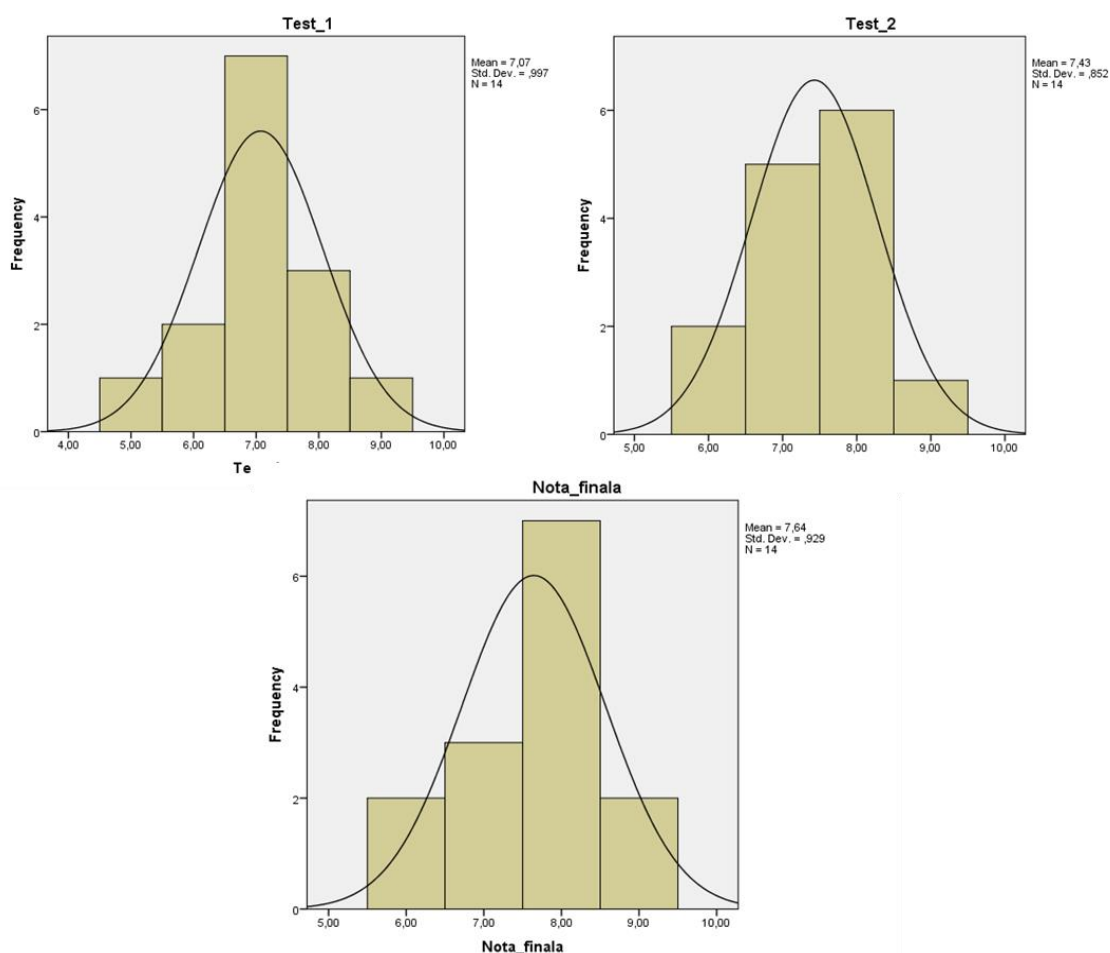


Fig. 3.6. Călea în SPSS pentru construirea histogramei de distribuție a variabilei cantitative

Histogramele pentru eșantionul experimental, anul de studii 2015–2016, sunt afișate în figura 3.7.



Notă. Mean – media (m); Std.Dev – abaterea-standard (s); N – numărul de studenți din eșantion (n); pe axa x sunt notele luate de studenți la test, iar pe axa y – numărul de studenți.

Fig. 3.7. Histogramele eșantionului experimental, anul de studii 2015-2016

Analizând graficele din figura 3.7, se poate face concluzia că la testul final (Nota_finală, figura 3.7) au fost înregistrați 9 studenți cu media mai mare decât media generală 7,64, ceea ce reprezintă 64% din numărul de subiecți ai eșantionului experimental, anul 2015–2016.

Asimetria unei serii se definește în raport cu dispunerea unităților într-o parte sau alta a valorii medii.

Coeficientul de asimetrie indică tipul de asimetrie. Indicele de asimetrie A al lui Fisher se poate calcula după următoarea formulă:

$$A = \frac{\sum_{i=1}^n d_i^3}{n \cdot S^3} \quad (3.1.)$$

unde $d_i = x_i - m$ – abaterea valorii x_i de la media m , S – abaterea-standard, n – numărul de subiecți.

În cazul în care indicele de asimetrie A al lui Fisher este pozitiv, vom spune că avem o distribuție asimetrică la dreapta și vor predomina scorurile mici. Modul este valoarea situată cel mai la stânga în lotul de date, iar mediana este mai mică decât media. Într-o distribuție asimetrică la stânga, indicele de asimetrie A al lui Fisher este negativ și predomină scorurile mari. În acest caz, mediana este mai mare decât media, iar modul este valoarea situată cel mai la dreapta. În cazul în care media, mediana și modul au aceleași valori, vorbim despre o distribuție perfect simetrică [171, p. 15-24].

În tabelul 3.10 sunt prezentate valorile mediei, abaterii-standard și indicatorului de asimetrie Fisher pentru două teste sumative și testul final pentru eșantionul experimental, anul de studii 2015–2016.

Tabelul 3.10. Indicele A al lui Fisher și abaterea-standard, eșantionul experimental, anul 2015–2016

	Test_1	Test_2	Nota_finală
Media	7,0714	7,4286	7,6429
Abaterea-standard (s)	0,99725	0,85163	0,92878
Indicatorul lui Fisher (A)	0,21	-0,910	-1,160

Deoarece indicele de asimetrie A al lui Fisher pentru două teste este negativ, se poate face concluzia că aceste două variabile (notele la testul 2 sumativ și la testul final) au o distribuție asimetrică la stânga. Prin urmare, în acest eșantion, la testele respective, predomină scorurile mari. Indicele de asimetrie A al lui Fisher pentru testul 1 are valoare pozitivă, ceea ce arată că

valorile notelor la primul test au scoruri mai mici. Acest lucru se explică prin faptul că testul respectiv a fost desfășurat la începutul experimentului și studenții se aflau în etapa de familiarizare cu conceptul POO – o disciplină cu un grad sporit de dificultate. Dar și în acest caz 28% au avut note cu mult mai mari decât media (note de 8 și 9), media generală fiind 7.07; nota 7 au avut 50%, ceea ce este foarte aproape de medie, și doar 21% au avut note mai mici decât media.

Pentru eșantionul de control, anul de studii 2015–2016, indicatorul lui Fisher pentru testul final $A = 0,81$, ceea ce denotă faptul că scorurile notelor mici în acest eșantion sunt în procent destul de mare (42%). Histograma eșantionului de control este prezentată în figura 3.8.

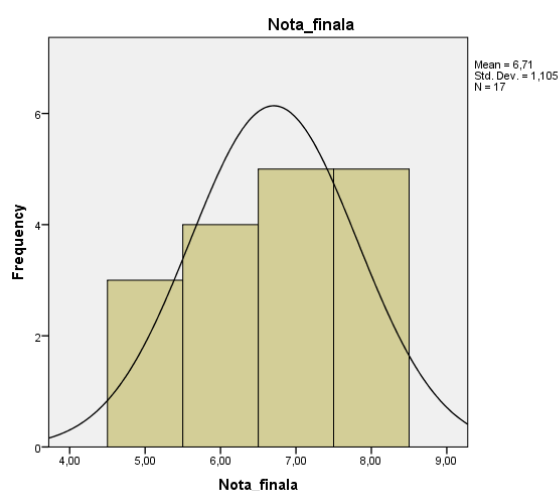


Fig. 3.8. Histograma eșantionului de control, testul final, anul 2015-2016

Indicatorul lui Fisher pentru eșantioanele experimental și de control, anul de studii 2016–2017, secția cu frecvență la zi, la testul final a avut, respectiv, valorile: $A = -0,507$ pentru eșantionul experimental și $A = 0,082$ pentru cel de control. Acest fapt ne demonstrează că avem o distribuție asimetrică la stânga, adică în direcția scorurilor mari, pentru eșantionul experimental, și o distribuție asimetrică la dreapta, adică în direcția scorurilor mici, în cazul eșantionului de control. Din punct de vedere cantitativ, în eșantionul experimental sunt 54% note mai mari decât media generală de 8,87 și 46% note mai mici decât media generală, iar în eșantionul de control, 63% din note sunt mai mici decât media generală de 6,22 și doar 35% sunt mai mari de medie.

Aceeași tendință pentru indicatorul lui Fisher (Anexele 12, 13) s-a menținut și în cazul eșantioanelor experimental și de control, anul de studii 2016–2017, secția cu frecvență redusă. Indicatorul lui Fisher pentru eșantionul experimental este $A = -0,215$ și pentru eșantionul de control $A = 0,399$. Astfel, pentru eșantionul experimental avem o distribuție asimetrică la stânga,

iar pentru eșantionul de control – o distribuție asimetrică la dreapta. În cazul eșantionului experimental, media generală e 7,91 și sunt înregistrate note mai mari decât media în proporție de 58% și 42% note mai mici de medie, iar în cazul eșantionului de control cu media 6,21, doar 36% au fost note mai mari decât media generală și 64% note mai mici de medie.

În continuare vor fi aplicate testele statistice t-Student, (U) Mann-Whitney și Wilcoxon, prin intermediul softului SPSS, în eșantioanele incluse în experimentul pedagogic, în scopul de a confirma ipoteza de cercetare H_1 și a respinge ipoteza nulă H_0 . Rezultatele vor fi analizate conform etapelor de desfășurare a experimentului.

Pentru fiecare grup de eșantioane implicate în experimentul pedagogic a fost calculată valoarea testului t-Student. Rezultatele furnizate de softul SPSS, pentru anul de studii 2015–2016, sunt afișate în tabelul 3.11.

Tabelul 3.11. Testul t-Student, anul de studii 2015–2016

Nota_finală	Levene's Test for Equality of Variances		t-test for Equality of Means						
	F	Sig.	t	df	Sig. (2-tailed) p	Mean Difference Δ_m	Std. Error Difference	95% CID*	
								Lower	Upper
Equal variances assumed	0,981	0,330	-2,521	29	0,017	-0,93697	0,37160	-1,69697	-0,17697
Equal variances not assumed			-2,565	28,978	0,016	-0,93697	0,36526	-1,68404	-0,18991

Notă. * – CID (Confidence Interval of the Difference).

Analizând datele din tabelul 3.11, vedem că valorile testului Levene sunt nesemnificative: $F(29) = 0,981$ și $p = 0,330 > 0,05$. În acest caz se satisface condiția de omogenitate a varianțelor. Valoarea lui $t = 2,521$ și a lui $p = 0,017 < 0,05$. Din Anexa 4 [167, p. 358] avem: $t_{cr} = 2,024$. Prin urmare, există diferențe semnificative între mediile eșantioanelor experimental și de control, deoarece $t > t_{cr}$, $2,521 > 2,024$, adică se afirmă ipoteza H_1 , iar ipoteza nulă H_0 se respinge.

Se dovedește că relația $\alpha_1 < \Delta_m < \alpha_2$, $-1,69697 < -0,93697 < -0,17697$ se îndeplinește, unde α_1, α_2 sunt limitele intervalului de încredere, cu o probabilitate de 95%, iar Δ_m – diferența dintre mediile eșantioanelor examinate. Întrucât $0 \notin (-1,69697; -0,17697)$, se demonstrează că diferența dintre mediile eșantionului experimental și media eșantionului de control este semnificativă.

Același test t-Student a fost aplicat și asupra eșantioanelor din anul de studii 2016–2017, pentru grupele (experimentală și de control) cu frecvență la zi. Rezultatele acestui test, obținute cu ajutorul aplicației SPSS, sunt reprezentate în tabelul 3.12.

Tabelul 3.12. Testului t-Student, anul de studii 2016–2017, grupele cu frecvență la zi

Nota_finală	Levene's Test for Equality of Variances		t-test for Equality of Means						
			t	df	Sig. (2-tailed) p	Mean Difference Δ_m	Std. Error Difference	95% CID*	
	F	Sig.						Lower	Upper
Equal variances assumed	0,048	0,829	-9,868	28	0,000	-2,65339	0,26889	-3,20419	-2,10259
Equal variances not assumed			-9,873	26,029	0,000	-2,65339	0,26875	-3,20578	-2,10101

Notă. * – CID (Confidence Interval of the Difference)

Conform datelor din tabelul 3.12, rezultatele testului Levene sunt nesemnificative: $F(28) = 0,048$ și $p = 0,829 > 0,05$, ceea ce demonstrează omogenitatea varianțelor. S-au obținut valorile: $t = 9,868$, iar $p = 0,00 < 0,05$. Pentru gradele de libertate $df = 28$, din Anexa 4 [167, p. 358] se citește: $t_{cr} = 2,048$. Prin faptul că $t > t_{cr}$, $9,868 > 2,048$, se confirmă existența unor diferențe semnificative între mediile eșantioanelor experimental și de control.

Ca și în cazul precedent, se satisface relația $\alpha_1 < \Delta_m < \alpha_2$. deoarece nici în acest caz $0 \notin (-3,20419; -2,10259)$, se confirmă o diferență semnificativă între mediile eșantionului experimental și eșantionului de control.

Rezultatele testului t-Student pentru anul de studii 2016–2017, secția cu frecvență redusă, sunt prezentate în tabelul 3.13.

Tabelul 3.13. Testul t-Student, anul de studii 2016–2017, grupele cu frecvență redusă

Nota_finală	Levene's Test for Equality of Variances		t-test for Equality of Means						
			t	df	Sig. (2-tailed) p	Mean Difference Δ_m	Std. Error Difference	95% CID*	
	F	Sig.						Lower	Upper
Equal variances assumed	1,548	0,227	-3,967	21	0,001	-1,73485	0,43735	-2,64437	-0,82532
Equal variances not assumed			-4,053	18,186	0,001	-1,73485	0,42800	-2,63339	-0,83631

Rezultatul testului Levene, prezentat în tabelul 3.13, este nesemnificativ: $F(21) = 1,548$, iar $p = 0,227 > 0,05$, prin urmare este demonstrată omogenitatea varianțelor și rezultatele citite

vor fi din linia întâi. Valoarea lui $t = 3,967$, iar a lui $p = 0,001 < 0,05$. Astfel, pentru $df = 21$, conform Anexei 4 [167, p. 358], $t_{cr} = 2,080$. Deoarece $t > t_{cr}$, $3,967 > 2,080$, și în acest caz se stabilește existența diferențelor semnificative între mediile eșantioanelor experimental și de control. La fel ca și în cazurile precedente, se păstrează relația: $\alpha_1 < \Delta_m < \alpha_2$, iar $0 \notin (-2,64437; -0,82532)$. Astfel, se adevărește că diferența dintre medii este semnificativă.

Prin testarea ipotezelor se verifică doar semnificația statistică, care arată în ce măsură rezultatul eșantionului depășește nivelul de probabilitate, dar nu se verifică dacă efectul există sau nu. Testul de semnificație arată doar dacă există o diferență semnificativă, pe când mărimea efectului arată cât de mare este această diferență. Prin urmare, mărimea efectului indică magnitudinea efectului studiat și este un instrument important pentru interpretarea eficacității. Mărimea efectului se referă la amploarea efectului în ipoteza de alternativă. Ținând cont de acest lucru, Asociația Americană pentru Psihologie (APA) și Society for Industrial and Organizational Psychology (Asociația de Psihologie Industrială și Organizațională) au recomandat ca toate rapoartele statistice publicate să includă și mărimea efectului [164, p.116].

Mărimea efectului conține o serie de indicatori care măsoară amploarea efectului. Spre deosebire de testele de semnificație, acești indici sunt independenți de dimensiunea eșantionului.

Indicatorii mărimii efectului pot fi clasificați în două categorii:

1. Indicatori care se bazează pe diferența standardizată dintre medii, de exemplu indicatorul d al lui J. Cohen, indicatorul delta (δ) al lui Glass etc.;
2. Indicatori care se bazează pe gradul de asociere dintre variabile, de exemplu, r , r^2 , η^2 [166, p. 82-83].

Pentru a calcula mărimea efectului, de cele mai dese ori este folosit indicatorul r sau indicatorul d al lui Cohen [167, p. 238-249].

Pentru a demonstra că diferența semnificativă dintre mediile înregistrate de studenții din grupul experimental la unitatea de curs POO față de mediile obținute de studenții din grupul de control, înregistrate la aceeași disciplină de studiu, nu este una aleatorie, întâmplătoare, se va calcula mărimea efectului, care este un indicator statistic ce cuantifică mărimea diferenței dintre aceste medii sau intensitatea asocierii dintre variabilele independente supuse testării.

Formulele de calcul pentru indicatorii d și r , pentru testul t-Student aplicat pe eșantioane independente, sunt:

$$d = \frac{t(n_1 + n_2)}{\sqrt{df} \cdot \sqrt{n_1 n_2}} \quad (3.2)$$

$$r = \frac{t(n_1 + n_2)}{\sqrt{t^2(n_1 + n_2)^2 + 4df \cdot n_1 n_2}} \quad (3.3)$$

unde t – rezultatul testului t-Student; n_1, n_2 – numărul de subiecți din fiecare eșantion; df – gradele de libertate.

Rezultatele obținute prin aplicarea formulelor 3.2 și 3.3 sunt reflectate în tabelul 3.14.

Tabelul 3.14. Mărimea efectului, pentru eșantioanele (control și experimental) implicate în experimentul pedagogic, în baza rezultatelor testului t-Student

Anul de studii	t	d	r
2015-2016 (EE – 14, EC – 17)	2,521	0,940691	0,425617
2016-2017, frecvența la zi (EE – 13, EC – 17)	9,868	3,763355	0,883046
2016-2017, frecvență redusă (EE – 12, EC – 11)	3,967	1,732979	0,654854

Analizând rezultatele testului t-Student din punctul de vedere al mărimii efectului, se poate face următoarea concluzie: efectul produs asupra performanțelor academice ale studenților din eșantioanele experimentale, în urma implementării modelului pedagogic elaborat și a metodologiei propuse, axate pe aplicarea SMÎ Moodle în procesul didactic la unitatea de curs POO, este:

1. „puternic” (2015–2016), „foarte puternic” (2016–2017) în baza mărimii indicatorului d al lui Cohen;
2. „mediu” (2015–2016), „foarte puternic” (2016–2017, grupele cu frecvență la zi) și „puternic” (2016–2017, grupele cu frecvență redusă) în baza mărimii indicatorului r .

Asupra datelor stocate în cadrul experimentului de formare a fost aplicat și testul U Mann-Whitney pe aceleași eșantioane. Rezultatele testului respectiv aplicat pe eșantioanele (control și experimental), participante la experimentul pedagogic desfășurat în anul de studii 2015–2016, sunt redate în tabelele 3.15 și 3.16.

Tabelul 3.15. Suma rangurilor, anul de studii 2015–2016

Ranks				
	Grup	N	Mean Rank	Sum of Ranks
Nota_fin:	control	17	12,76	217,00
	experimental	14	19,93	279,00
	Total	31		

În tabelul 3.15 se indică numărul de studenți încadrați în experimentul pedagogic, suma rangurilor $\sum R_E = 279,00$ și $\sum R_C = 217,00$ și media rangurilor. Analizând rezultatele furnizate,

se poate conchide că se satisface relația $\sum R_E > \sum R_C$. Deci, concluzia este că notele studenților din grupul experimental sunt mai mari decât notele înregistrate de studenții grupului de control.

Tabelul 3.16. Testul Mann-Whitney, anul 2015–2016

Test Statistics	
	Nota_fin
Mann-Whitney U	64,000
Z	-2,279
Asymp. Sig. (2-tailed), p	0,023

Valoarea U a testului Mann-Whitney este $U = 64$ pentru un prag de semnificație $p = 0,023 < 0,05$ (tabelul 3.16). Această valoare, în comparație cu mărimea critică $U_{cr} = 67$, este mai mică, fapt ce confirmă existența unor diferențe semnificative între eșantionul experimental și eșantionul de control. Astfel, a fost respinsă ipoteza H_0 și demonstrată veridicitatea ipotezei H_1 .

Testul Mann-Whitney a fost aplicat, repetat, și pe eșantioanele de control și experimental, secția cu frecvență la zi, anul de studii 2016–2017, iar rezultatele sunt oglindite în tabelele 3.17 și 3.18.

Tabelul 3.17. Suma rangurilor, anul 2016–2017, grupele cu frecvență la zi

Ranks				
	Grup	N	Mean Rank	Sum of Ranks
Nota_fin:	control	17	9,00	153,00
	experimental	13	24,00	312,00
	Total	30		

Tabelul 3.18. Testul Mann-Whitney, anul 2016-2017, grupele cu frecvență la zi

Test Statistics	
	Nota_fin
Mann-Whitney U	0,000
Z	-4,641
Asymp. Sig. (2-tailed), p	0,000

Adevărul relației $\sum R_E > \sum R_C$ asigurat de datele furnizate de SPSS prin aplicarea testului Mann-Whitney, aplicat pe eșantioane independente, anul de studii 2016–2017, secția cu frecvență la zi, dovedește faptul că notele obținute de studenții din grupul experimental sunt mai ridicate decât notele obținute de cei din grupul de control. Datele din tabelul 3.18: $U = 0,00$; $p = 0,000$ demonstrează, la un $U_{cr} = 63$, relația fiind $U < U_{cr}$, că se înregistrează diferențe semnificative între mediile acestor două eșantioane: experimental și de control.

În mod analogic, testul Mann-Whitney a fost aplicat și asupra rezultatelor grupurilor de control și experimental din anul de studii 2016–2017, secția cu frecvență redusă. Rezultatele sunt incluse în tabelele 3.19 și 3.20.

Tabelul 3.19. Suma rangurilor, anul 2016–2017, grupele cu frecvență redusă

Ranks				
	Grup	N	Mean Rank	Sum of Ranks
Nota_fin:	control	11	7,27	80,00
	experimental	12	16,33	196,00
	Total	23		

Tabelul 3.20. Testul Mann-Whitney, anul 2016-2017, grupele cu frecvență redusă

Test Statistics	
	Nota_fin
Mann-Whitney U	14,000
Z	-3,236
Asymp. Sig. (2-tailed), p	0,001

Din analiza rezultatelor furnizate de softul SPSS, se atestă faptul că și în cazul dat suma rangurilor și media rangurilor calculate pentru grupul experimental sunt mai mari decât cele calculate pentru grupul de control (tabelul 3.19). Analogic, datele din tabelul 3.20 ($U = 14,00$; $p = 0,001$ și $U < U_{cr} = 33$) ne permit să concluzionăm referitor la existența diferențelor semnificative între notele finale, la unitatea de curs *POO*, ale studenților din eșantionul experimental vizavi de notele finale înregistrate de studenții din eșantionul de control.

În continuare se va calcula indicatorul mărimii efectului în funcție de rezultatele obținute în urma aplicării testului Mann-Whitney pe datele furnizate de eșantioanele implicate în experimentul pedagogic. Se va aplica formula:

$$r = \sqrt{\frac{z^2}{n}} \quad (3.4)$$

unde: $n = n_1 + n_2$; n_1, n_2 – numărul de subiecți din fiecare eșantion; z – scorul z al testului U Mann-Whitney.

Scorul z al testului Mann-Whitney este furnizat de softul SPSS în fișierele de output în același tabel în care se indică și valoarea U a testului respectiv, iar rezultatele obținute sunt indicate în tabelul 3.21.

Tabelul 3.21. Mărima efectului în eșantioanele cercetate, conform testului Mann-Whitney

Anul de studii	U	z	r
2015-2016 (EE – 14, EC – 17)	64,000	2,279	0,40932
2016-2017, frecvența la zi (EE – 13, EC – 17)	0,000	4,641	0,847327
2016-2017, frecvență redusă (EE – 12, EC – 11)	14,000	3,236	0,674753

Astfel, în funcție de indicatorul r , noua metodologie propusă, axată pe implementarea SMÎ Moodle în procesul de instruire la unitatea de curs *POO*, a produs un efect mediu asupra performanțelor academice obținute de studenții din eșantionul experimental în anul de studii 2015-2016; foarte puternic – în anul 2016–2017, asupra eșantionului experimental cu studii la zi, și puternic pentru eșantionul experimental de la secția cu frecvență redusă.

În scopul determinării gradului de corelare a notelor obținute la testele sumative și la testul final de către studenții din eșantionul experimental, anul de studii 2015–2016, secția cu frecvență la zi, a fost aplicat testul Pearson.

Corelarea rezultatelor obținute la testele sumative și la testul final de către studenții din eșantionul experimental, anul de studii 2015-2016, este afișată în tabelul 3.22.

Tabelul 3.22. Valoarea coeficientului de corelare Pearson pentru eșantionul experimental, anul de studii 2015–2016, secția cu frecvență la zi

		Test_1	Test_2	Nota_finala
Test_1	Correlația Pearson	1	0,595	0,528
	Sig. (2-tailed)		0,025	0,052
	N	14	14	14
Test_2	Correlația Pearson	0,595	1	0,792
	Sig. (2-tailed)	0,025		0,001
	N	14	14	14
Nota_finală	Correlația Pearson	0,528	0,792	1
	Sig. (2-tailed)	0,052	0,001	
	N	14	14	14

Analizând datele afișate în acest tabel, putem conchide că gradul de corelare între testul_1 și testul_2 este 0,595, iar gradul de corelare între testul_2 și testul_final este 0,792. Cu cât valoarea coeficientului Pearson este mai aproape de 1, cu atât corelarea dintre aceste variabile este mai mare [165, p. 13].

Chiar dacă impactul testului unu a fost mai mic asupra testului final (a se vedea explicația la descrierea asimetriei distribuției, testul Fisher), impactul testului al doilea asupra testului final a fost unul foarte mare. Prin urmare, în cazul dat putem vorbi despre o influență pozitivă a testului doi asupra scorurilor înregistrate la testul final.

Testul Pearson a fost aplicat și asupra eșantionului experimental, anul de studii 2016–2017, secția cu frecvență la zi, iar valorile coeficientului de corelare Pearson sunt afișate în tabelul 3.23

Tabelul 3.23. Valoarea coeficientului de corelare Pearson pentru eșantionul experimental, anul de studii 2016–2017, secția cu frecvență la zi

		Test_1	Test_2	Nota_finala
Test_1	Correlația Pearson Sig. (2-tailed) N	1 13	0,744 0,000 13	0,772 0,000 13
Test_2	Correlația Pearson Sig. (2-tailed) N	0,744 0,000 13	1 13	0,961 0,000 13
Nota_finală	Correlația Pearson Sig. (2-tailed) N	0,772 0,000 13	0,961 0,000 13	1 13

Din tabelul 3.23 se poate face concluzia despre gradul de corelare Pearson pentru eșantionul experimental, anul de studii 2016–2017, secția cu frecvență la zi. Coeficientul de corelare Pearson între testul unu și testul doi are valoarea 0,744, astfel atestându-se un impact pozitiv destul de mare a primului test asupra performanțelor atestate la testul doi. Gradul de corelare între testul_1 și testul_final este 0,772, iar gradul de corelare între testul_2 și testul_final este 0,961. Aceste valori denotă faptul că ambele teste sumative au avut un impact semnificativ asupra testului final, iar testul doi a influențat puternic scorurile înalte obținute la testul final.

Asupra eșantionului experimental, anul de studii 2016–2017, secția cu frecvență redusă, de asemenea a fost aplicat testul Pearson, iar rezultatele lui sunt prezentate în tabelul 3.24.

Tabelul 3.24. Valoarea coeficientului de corelare Pearson pentru eșantionul experimental, anul de studii 2016–2017, secția cu frecvență redusă

		Test_1	Test_2	Nota_finala
Test_1	Correlația Pearson Sig. (2-tailed) N	1 12	0,894 0,000 12	0,922 0,000 12
Test_2	Correlația Pearson Sig. (2-tailed) N	0,894 0,000 12	1 12	0,985 0,000 12
Nota_finală	Corelația Pearson Sig. (2-tailed) N	0,922 0,000 12	0,985 0,000 12	1 12

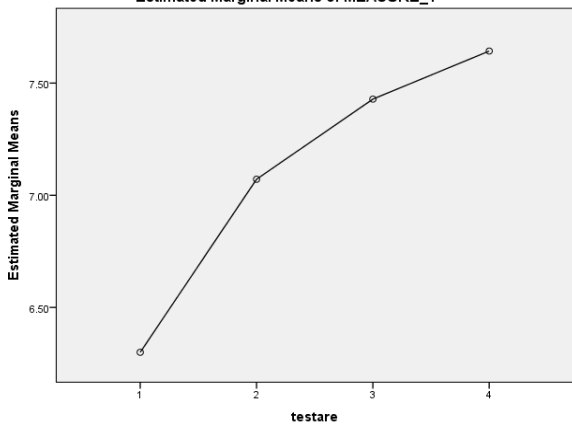
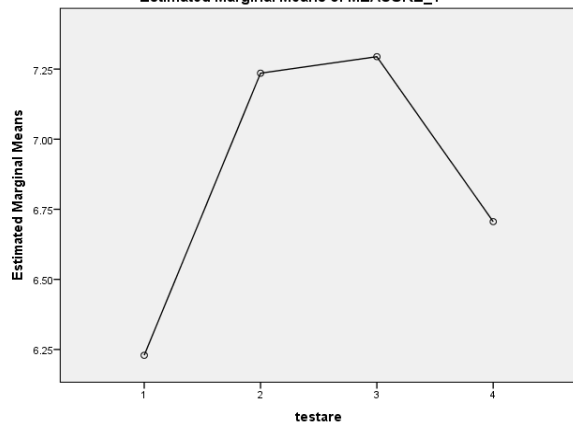
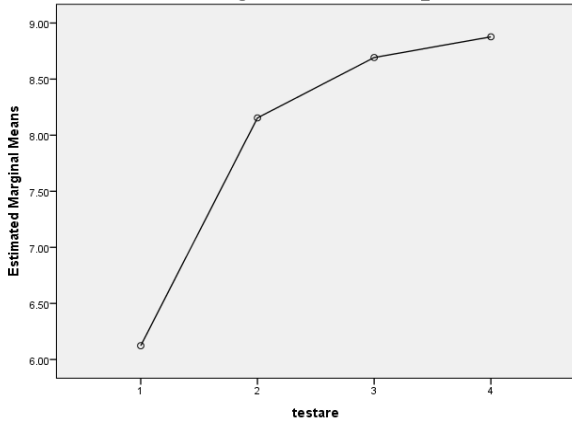
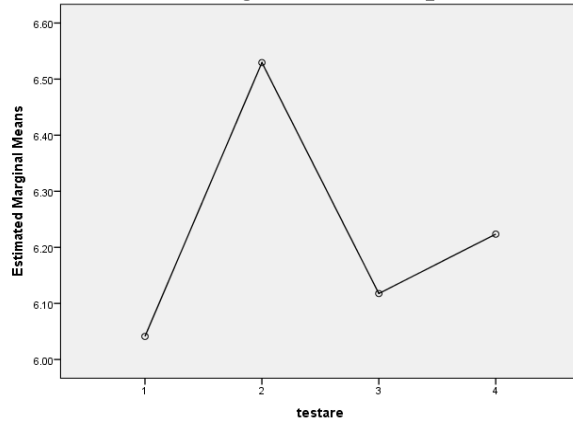
În acest caz, coeficientul de corelare Pearson între testul unu și testul doi are valoarea 0,894, astfel atestându-se un impact foarte mare al rezultatelor primului test asupra testului doi. În același timp, gradul de corelare între testul doi și cel final ajunge la valoarea 0,985. Ținând cont de faptul că atunci când valoarea coeficientului Pearson este mai aproape de 1 gradul de corelare este unul foarte mare, putem afirma cu siguranță că rezultatele testului doi au influențat cu desăvârșire performanțele testului final.

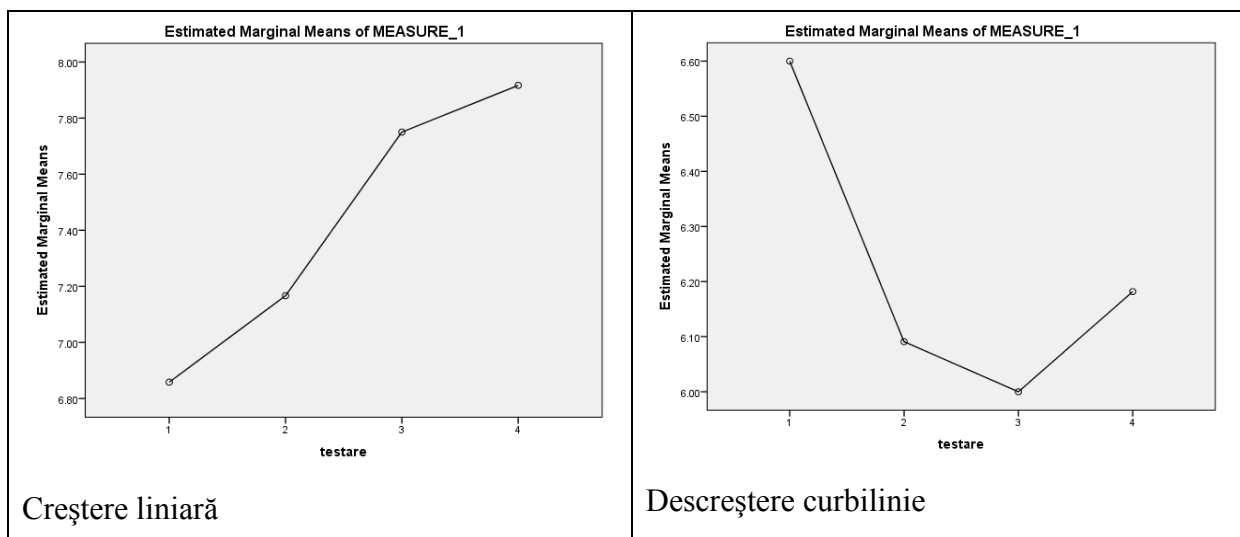
Așadar, potrivit rezultatelor testului Pearson, se poate face concluzia că eșantioanele experimentale au obținut performanțe academice semnificative ca urmare a aplicării modelului pedagogic orientat pe implementarea SMÎ Moodle în procesul de studiere a cursului universitar *Programarea Orientată pe Obiecte*.

Asupra eșantioanelor experimentale și de control implicate în experimentul pedagogic a fost aplicat și testul ANOVA.

În continuare, prin aplicarea acestui test, realizat cu ajutorul softului SPSS, au fost construite graficele de tendință pentru fiecare eșantion implicat în experimentul pedagogic, reflectate în tabelul 3.25.

Tabelul 3.25. Graficele de tendință pentru eșantioanele independente,
implicate în experimentul pedagogic

Eșantionul experimental	Eșantionul de control
2015–2016	
Estimated Marginal Means of MEASURE_1  Estimated Marginal Means testare Creștere liniară	Estimated Marginal Means of MEASURE_1  Estimated Marginal Means testare Creștere curbilinie
2016–2017, grupele cu frecvență la zi	
Estimated Marginal Means of MEASURE_1  Estimated Marginal Means testare Creștere liniară	Estimated Marginal Means of MEASURE_1  Estimated Marginal Means testare Creștere cubică
2016–2017, grupele cu frecvență redusă	



Pentru a demonstra că tendința de creștere a performanțelor academice ale studenților este una semnificativă în grupurile experimentale, a fost aplicat testul Wilcoxon pe același eșantion, fiind comparate statistic media înregistrată de studenți la testul inițial și nota finală la unitatea de curs *POO*. Rezultatele testului Wilcoxon sunt concentrate în tabelul 3.26.

Tabelul 3.26. Rezultatele testului Wilcoxon, în baza mediilor înregistrate de studenți la testul inițial și nota finală la unitatea *POO* din eșantioanele experimental și de control

Anul de studiu	Eșantion	Media Test inițial	Media Nota finală	Testul Wilcoxon <i>Z</i>	Pragul de semnificație <i>p</i>	Interpretarea rezultatelor
2015–2016	experimental	6,30	7,6429	-2,554	0,011	creștere semnificativă
	de control	6,22	6,7059	-1,540	0,124	creștere nesemnificativă
2016–2017, frecvență la zi	experimental	6,12	8,8769	-3,185	0,001	creștere semnificativă
	de control	6,04	6,2235	-1,163	0,245	creștere nesemnificativă
2016–2017, frecvență redusă	experimental	6,85	7,9167	-2,512	0,012	creștere semnificativă
	de control	6,60	6,1818	-2,940	0,073	descreștere nesemnificativă

Analizând rezultatele din tabelul 3.26 privind aplicarea testului Wilcoxon, se poate afirma că studenții din eșantioanele experimentale au obținut performanțe academice semnificative în urma implementării modelului pedagogic orientat pe aplicarea SMÎ Moodle în procesul de studiere a cursului universitar *Programarea Orientată pe Obiecte*, iar performanțele academice ale studenților din grupurile de control au înregistrat creșteri nesemnificative.

Astfel, în urma prelucrării datelor statistice, s-a demonstrat că utilizarea TIC în procesul didactic la unitatea de curs *POO*, în special a SMÎ Moodle, s-a soldat cu efecte pozitive în grupul experimental, iar studenții au obținut rezultate mai înalte, comparativ cu studenții grupului de control.

Prin rezultatele obținute a fost confirmată complet ipoteza cercetării.

3.4. Concluzii la capitolul 3

Rezultatele experimentului pedagogic desfășurat în perioada anilor de studii 2015–2016 și 2016–2017, ne-au permis să facem următoarele concluzii:

Rezultatele experimentului pedagogic desfășurat în perioada anilor de studii 2015–2016 și 2016–2017, ne-au permis să facem următoarele concluzii:

1. Prin experimentul de constatare, realizat în cadrul cercetării, s-a reliefat oportunitatea elaborării unui model pedagogic și a unei metodologii de implementare eficace a Sistemelor de Management al Învățării, în scopul formării inițiale a viitorilor specialiști de informatică.

2. În experimentul de formare s-a demonstrat eficiența utilizării modelului pedagogic elaborat în studierea Programării Orientate pe Obiecte și a metodologiei de implementare a acestui model.

3. Metodologia propusă a permis valorificarea metodelor de individualizare a învățării, ceea ce a dus la creșterea semnificativă a performanțelor academice ale studenților.

4. Grație unui feedback *profesor-student* eficient, susținut de SMÎ Moodle, au fost identificate și eliminate dificultățile studenților în studierea *Programării Orientate pe Obiecte*, acest fapt fiind decisiv la elaborarea curriculumului la cursul universitar *Programarea Orientată pe Obiecte*.

5. Datele înregistrate în experimentul de formare au confirmat ipoteza cercetării prin consemnarea unor performanțe academice semnificative, înregistrate de studenții eșantionului experimental, comparativ cu rezultatele obținute de studenții eșantionului de control, fapt demonstrat de testele statistice, astfel fiind dovedită eficiența modelului propus.

Așadar, experimentul pedagogic desfășurat de autor a validat eficiența modelului pedagogic de studiere a Programării Orientate pe Obiecte și a metodologiei de implementare a acestui model. Astfel, a fost complet soluționată problema cercetării: fundamentarea teoretico-praxiologică a eficientizării procesului de studiere a *Programării Orientate pe Obiecte*, prin elaborarea unui model pedagogic de predare-învățare-evaluare în baza cursului universitar *Programarea Orientată pe Obiecte*, axat pe utilizarea Sistemelor de Management al Învățării și orientat spre formarea competenței de Programare Orientată pe Obiecte la viitorii specialiști în informatică (inclusiv în domeniul *Științe ale Educației*).

CONCLUZII GENERALE ȘI RECOMANDĂRI

Cercetarea efectuată vizează elaborarea metodologiei de predare-învățare-evaluare a disciplinei *Programarea Orientată pe Obiecte* prin utilizarea Sistemelor de Management al Învățării (SMÎ).

Analiza detaliată a rezultatelor cercetărilor în domeniul utilizării TIC în educație și a platformelor de instruire au pus în evidență importanța modernizării procesului educațional prin utilizarea eficientă a SMÎ în studierea informaticii, în special a programării orientate pe obiecte.

În urma cercetării efectuate se pot face următoarele **concluzii generale**:

1. Au fost determinate avantajele oferite de Sistemele de Management al Învățării și argumentată necesitatea implementării lor în procesul didactic universitar. Cele mai importante dintre ele se referă la: promovarea învățării active, a interacțiunii, a gândirii critice; creșterea motivației studenților; oferirea instantă a feedback-ului; furnizarea conținuturilor de învățare în diferite formate (multimedia, video, text); extinderea timpului și a locației de acces; individualizarea conținuturilor și traseelor de învățare; reutilizarea activităților de învățare; diversificarea metodelor și tehnologiilor de predare-învățare [14].

2. S-a precizat că Sistemul de Management al Învățării (open source) cu caracteristici optime pentru sistemul educațional universitar este SMÎ Moodle. Totodată s-a demonstrat că implementarea SMÎ Moodle în procesul de studiu al Programării Orientate pe Obiecte oferă noi oportunități, determinate de un șir de avantaje: accesul la o gamă largă de abordări formative și strategii de învățare flexibile (formarea participativă, metacogniția, învățarea axată pe probleme etc.), evaluarea diversificată (autoevaluare, evaluarea formativă, cea sumativă etc.), abordarea asincronă în timp și spațiu, conținuturi dinamice, mediu colaborativ, feedback etc. [14, 40, 96, 135, 136, 159].

3. În urma analizei comparative a limbajelor orientate pe obiecte s-a identificat limbajul orientat pe obiecte adecvat studierii conceptului de Programare Orientată pe Obiecte [19].

4. Modelul pedagogic elaborat este axat pe utilizarea Sistemelor de Management al Învățării și aplicarea lui crește eficiența procesului de studiu al Programării Orientate pe Obiecte [136].

5. Eficiența utilizării modelului pedagogic elaborat a fost argumentată din punct de vedere științifico-metodologic prin metodologia de implementare a modelului, care include strategii interactive, metode didactice moderne, centrate pe student, care favorizează dezvoltarea gândirii critice [96, 99, 169, 159].

6. Curriculumul la cursul universitar *Programarea Orientată pe Obiecte*, elaborat de

autor, reflectă conceptul modelului SMÎ POO și pune accent pe corelarea dintre unitățile de învățare și instrumentele SMÎ, necesare pentru buna organizare a procesului de studiu al POO.

7. Resursele digitale și tehnologiile educaționale axate pe instrumente ale SMÎ Moodle, elaborate de autor pentru cursul universitar *Programarea Orientată pe Obiecte*, sunt centrate pe student și extind substanțial parametrii *timp de studiu, acces la resurse, locație* de instruire.

8. Prin experimentul de constatare, realizat în cadrul cercetării, s-a validat modelul pedagogic și eficacitatea metodologiei de implementare a Sistemelor de Management al Învățării, în scopul formării inițiale a viitorilor specialiști de informatică. Experimentul de formare a demonstrat eficiența utilizării modelului pedagogic elaborat în studierea Programării Orientate pe Obiecte și a metodologiei de implementare a acestui model [162].

Rezolvarea problemei de cercetare și realizarea obiectivelor propuse sunt confirmate de rezultatele obținute, publicate de autor în lucrările [14, 19, 40, 96, 99, 135, 136, 159, 162, 169].

În conformitate cu rezultatele obținute, putem face următoarele **recomandări practice**:

- Implementarea modelului pedagogic, centrat pe utilizarea Sistemelor de Management al Învățării în procesul de studiu al Programării Orientate pe Obiecte, și a metodologiei de implementare a acestui model prin aplicarea strategiilor interactive, a metodelor didactice moderne, centrate pe student și orientate spre dezvoltarea gândirii critice, în procesul de predare a disciplinelor Informatică.
- Aplicarea rezultatelor obținute pentru elaborarea resurselor educaționale moderne, preponderent digitale.
- Studierea materialelor publicate la tematica tezei pentru eficientizarea activității cadrelor didactice din învățământul universitar.
- Integrarea rezultatelor obținute de autor în procesul de formare inițială a specialiștilor în domeniul Informaticii (inclusiv în domeniul *Științe ale Educației*).
- Elaborarea unui curs de formare profesională continuă a cadrelor didactice universitare în vederea utilizării Sistemelor de Management al Învățării în procesul didactic.
- Extinderea modelului elaborat (în context de perspectivă) asupra altor discipline din domeniul Științe exacte.
- Ajustarea modelului la fiecare 3-5 ani în corespundere cu realizările tehnologice.

BIBLIOGRAFIE

1. Tarik M., & Karim A. The use of Web 2.0 Innovations on Education and Training. In ICCC'11 Proceedings of the 2011 International Conference on Computers and Computing, 2011, p. 140-145.
<http://www.wseas.us/e-library/conferences/2011/Lanzarote/COMCOM/COMCOM-23.pdf>
(vizitat 12.08.2015)
2. Rosenberg M.J. The eLearning Guild's Handbook of e-Learning Strategy.
Published by The eLearning Guild http://theelearningcoach.com/wp-content/uploads/downloads/2010/06/strategy_ebook.pdf (vizitat 10.08.2015)
3. Recesso A. Prospect of a Technology-Based Learner Interface for Schools. Educational Technology and Society, 2001, vol. 4(1), ISSN 1436-4522.
http://www.ifets.info/journals/4_1/discuss_summary_nov2000.pdf (vizitat 10.08.2015)
4. Paulsen M.F. Online Education Systems: Discussion and Definition of Terms, 2002.
https://www.edutubebd.com/file_resource/1368197236online%20education%20system.pdf
(vizitat 05.08.2015)
5. Filippidi A., Tselios N., Komis V. Impact of Moodle usage practices on students' performance in the context of a blended learning environment. In: Proceedings of Social Applications for Life Long Learning, 2010, p. 2-7.
[https://www.academia.edu/425665/Impact of Moodle usage practices on students performance in the context of a blended learning environment](https://www.academia.edu/425665/Impact_of_Moodle_usage_practices_on_students_performance_in_the_context_of_a_blended_learning_environment) (vizitat 05.07.2015)
6. Brandon-hall.com.E-Learning Guidebook Six Steps to Implementing E-Learning.
<http://www.cedma-europe.org/newsletter%20articles/Brandon%20Hall/e-Learning%20Guide%20Book.pdf> (vizitat 05.08.2015)
7. Ellis R.K. A Field Guide to Learning Management Systems. ASTD: Learning Circuits, 2009. http://cgit.nutn.edu.tw:8080/cgit/PaperDL/hclin_091027163029.PDF (vizitat 20.06.2015)
8. Priyavahani S. et al. A Study of Comparison between Moodle and Blackboard based on Case Studies for Better LMS In: Journal of information systems research and innovation. p.26-33
http://www.moodlebites.com/pluginfile.php/26295/mod_resource/content/1/Pub4_ComparisonBetweenMoodleAndBlackboard.pdf (vizitat 15.08.2015)

9. Poulova P. et al. Which One, or Another? Comparative Analysis of Selected LMS In: Procedia - Social and Behavioral Sciences 186, 2015 p.1302 – 1308
<https://core.ac.uk/download/pdf/82462066.pdf> (vizitat 29.10.2015)
10. Sharma A., Vatta S. Role of Learning Management Systems in Education In: International Journal of Advanced Research in Computer Science and Software Engineering Volume 3, Issue 6, June 2013 p.997-1002
https://www.ijarcsse.com/docs/papers/Volume_3/6_June2013/V3I6-0456.pdf (vizitat 05.09.2015)
11. Al Naddabi Z. A Moodle Course: Design and Implementation in English for Academic Purposes Instruction. In: T. Bastiaens & S. Carliner (Eds.). Proceedings of World Conference on E-Learning in Corporate, Government, Healthcare, and Higher Education, 2007,p1371-1376
<http://research.moodle.net/98/1/Al%20Naddabi%20%282007%29%20A%20Moodle%20Course-%20Design%20and%20Implementation%20in%20English%20for%20Academic%20Purposes%20Instruction.pdf> (vizitat 24.08.2015)
12. Kulshrestha T. et al. Benefits of Learning Management System (LMS) in Indian Education. In: International Journal of Computer Science & Engineering Technology (IJCSET) Vol. 4 No. 08 Aug 2013 p. 1153-1164 <http://www.ijcset.com/docs/IJCSET13-04-08-036.pdf> (vizitat 18.08.2015)
13. <http://blogs.butler.edu/lms/files/2012/04/Executive-Summary-and-Recommendation.pdf> (vizitat 11.08.2015)
14. **Gasnaș A. Analiza comparativă a sistemelor de management al învățării. În: Materialele Conferinței științifice naționale cu participare internațională “Învățământul superior din Republica Moldova la 85 de ani”. Chișinău: UST, 2015, p. 122-128.**
15. Ajlan S. Al-Ajlan. A Comparative Study Between E-Learning Features. In: Elvis Pontes et al. Methodologies, Tools and New Developments for E-Learning, 2012, p. 191-214.
<http://cdn.intechweb.org/pdfs/27926.pdf> (vizitat 1.08.2015)
16. Armstrong D.J. The quarks of object-oriented development. In: Communications of the ACM, 2006, nr. 49(2), p. 123–128.
<https://ru.scribd.com/document/124176460/Armstrong-The-quarks-of-Objet-oriented-develop> (vizitat 2.03.2015)

17. Nierstrasz O. A Survey of Object-Oriented Concepts. In: Object-Oriented Concepts, Databases and Applications, 1989, p. 3-22.
<http://scg.unibe.ch/archive/osg/Nier89aSurveyOfOOConcepts.pdf> (vizitat 13.07.2015)
18. Объектно-ориентированное программирование.
http://ru.wikipedia.org/wiki/Объектно-ориентированное_программирование Сравнение ООП языков. <http://www.interface.ru/home.asp?artId=3567> (vizitat 02.02.2015)
19. **Gasnaş A. Studiu comparativ al caracteristicilor de bază ale limbajelor de programare orientate pe obiecte. În: Studia Universitatis Moldaviae (revistă științifică categoria B), 2015, nr. 9(89), p. 114-121. ISSN 1857-2103.**
20. Полиморфизм. [https://ru.wikipedia.org/wiki/Полиморфизм_\(информатика\)](https://ru.wikipedia.org/wiki/Полиморфизм_(информатика)) (vizitat 14.04.2015)
21. Fundamental Concepts in Programming Languages.
<http://www.itu.dk/courses/BPRD/E2009/> (vizitat 14.03.2015)
22. Параметрический полиморфизм как он есть. <http://fediq.ru/parametric-polymorphism/> (vizitat 13.03.2015)
23. Programare orientată obiect. <http://www.cs.ubbcluj.ro/~lauras/test/docs/school/oop/2013/lectures/curs07.pdf> (vizitat 13.03.2015)
24. Ellis M.A., Stroustrup B. The annotated C++ reference manual. Boston, MA, USA: Addison-Wesley Longman Publishing Co. Inc., 1990.
25. Gosling J. et al. Java (TM) Language Specification. The Java (Addison-Wesley). Addison-Wesley Professional, 2005. <https://docs.oracle.com/javase/specs/jls/se7/jls7.pdf> (vizitat 18.04.2015)
26. CSharp_Overview. http://en.csharponline.net/CSharp_Overview#A_Brief_History_of_C.23 (vizitat 17.03.2015)
27. Pascal_object.http://www.tutorialspoint.com/pascal/pascal_object_oriented.htm (vizitat 17.04.2015)
28. PHP: Hypertext Preprocessor. PHP Manual: Chapter 19 – Classes and Objects.
<http://php.net/manual/en/language.oop5.php>. (vizitat 12.05.2015)
29. A Touch of Class: Inheritance in JavaScript. <http://markdalgleish.com/2012/10/a-touch-of-class-inheritance-in-javascript/> (vizitat 3.06.2015)
30. POO și Java. Programarea Orientată pe Obiecte.<http://www.arh.pub.ro/lab/poo/lab2.html> (vizitat 5.05.2015)
31. CSharp. <http://www.math.uaic.ro/~cgales/csharp/Curs5.pdf> (vizitat 13.05.2015)
32. Delphi Language Reference.

- [http://docwiki.embarcadero.com/RADStudio/XE5/en/Operator_Overloading_\(Delphi\)](http://docwiki.embarcadero.com/RADStudio/XE5/en/Operator_Overloading_(Delphi))
(vizitat 22.06.2015)
33. Interfete, Delegari, Evenimente. <http://www.math.uaic.ro/~cgales/csharp/Curs6.pdf> (vizitat 22.07.2015)
 34. Comparing ObjectOriented Features. <https://docslide.com.br/documents/comparing-object-oriented-features-of-delphi-c-c-and-java.html> (vizitat 10.06.2015)
 35. PHP Manual. <http://php.net/manual/en/language.oop5.overloading.php> (vizitat 3.06.2015)
 36. Filippidi A. et al. Impact of Moodle usage practices on students' performance in the context of a blended learning environment. In: Social Applications for Life Long Learning, Patras, 2010, p. 2-7.
https://www.academia.edu/425665/Impact_of_Moodle_usage_practices_on_students_performance_in_the_context_of_a_blended_learning_environment (vizitat 7.07.2016)
 37. Tarik M., Karim A. The use of Web 2.0 Innovations on Education and Training. In: ICCC'11 Proceedings of the 2011 International Conference on Computers and Computing. 2011, p. 140-145. http://elearning-conf.org/wp-content/uploads/2014/11/EL_2014.pdf (vizitat 7.07.2016)
 38. Concannon F. et al. What campus-based students think about the quality and benefits of e-Learning. In: British Journal of Educational Technology, vol. 36(3), 2005, p. 501-512. <http://files.eric.ed.gov/fulltext/EJ1071365.pdf> (vizitat 8.07.2016)
 39. Pedro F. Comparing Traditional and ICT-Enriched University Teaching Methods: Evidence from Two Empirical Studies. In: Higher Education in Europe, 2005, vol. 30(3-4), p. 399-411. <http://www.tandfonline.com/doi/abs/10.1080/03797720600625937> (vizitat 8.07.2016)
 40. **Gasnaș A. Tendințe de aplicare a sistemelor LMS în predarea-învățarea-evaluarea informaticii. În: Didactica Pro. Revistă de teorie și practică educațională a Centrului Educațional PRO DIDACTICA, nr.1 (101), 2017, p. 17-23. ISSN: 1810-6455.**
 41. Martens R. et al. The impact of intrinsic motivation on e-learning in authentic computer tasks. In: Journal of Computer Assisted learning, 2004, vol. 20, p. 368-376. <http://citeseerx.ist.psu.edu> (vizitat 8.07.2016)
 42. Reiser R. A history of instructional design and technology. Part 2: a history of instructional design. In: Educational Technology, Research and Development, 2001, vol. 49, p. 57-67. http://www.speakeasydesigns.com/SDSU/student/SAGE/compsprep/History_of_Instructional_Design.pdf (vizitat 10.07.2016)
 43. Duffy T. et al. Designing Environments for Constructive Learning. Heidelberg: Springer-Verlag Berlin, 1993, p. 232-247. <https://books.google.md/>

http://www.speakeasydesigns.com/SDSU/student/SAGE/compsprep/History_of_Instructional_Design.pdf (vizitat 10.07.2016)

44. Wolters C.A., Pintrich P.R. Contextual differences in student motivation and self-regulated learning in mathematics, English and social studies classrooms. In: Instructional Science, 1998, vol. 26, p. 27-47. <https://deepblue.lib.umich.edu/bitstream/handle/2027> (vizitat 10.07.2016)
45. Bastiaens Th., Martens R. Conditions for web-based learning with real events. In: Instructional and Cognitive Impacts of Web-Based Education. London: Idea Group Publishing, p. 1-32. <https://www.ou.nl/bastiaens/tba.pdf> (vizitat 10.07.2016)
46. Herrington J., Oliver R. An instructional design framework for authentic learning environments. In: Educational Technology Research and Development, 2000, vol. 48, p. 23-48. <http://ro.uow.edu.au/cgi/viewcontent.cgi?article=1031&context=edupapers> (vizitat 10.07.2016)
47. Cabac V. Centrarea pe student și orientarea pe finalități de studii – piloni ai formării universitare moderne. În: Formarea universitară în medii digitale: cercetări teoretico-experimentale. Bălți, 2015, p. 11-36.
48. Steel C., Levy M. Creativity and constraint: Understanding teacher beliefs and the use of LMS technologies. In: Same places, different spaces. Proceedings ascilite Auckland, 2009. <http://www.ascilite.org.au/conferences/auckland09/procs/steel.pdf> (vizitat 12.07.2016)
49. Ryan S. et al. The virtual university: the internet and resource-based learning. London: Kogan Page, 2000. 218 p.
50. Islas E. et al. E-Learning Tools Evaluation and Roadmap Development for an Electrical Utility. In: Journal of Theoretical and Applied Electronic Commerce Research, 2007, vol. 2(1), p. 63-75. <http://www.redalyc.org/html/965/96520106> (vizitat 11.07.2016)
51. Douglas E., Van Der Vyver G. Effectiveness of E-Learning Course Materials For Learning Database Management Systems: an Experimental Investigation. In: Journal of Computer Information System, 2004, vol. 41(4), p. 41-48. [https://www.researchgate.net/publication/285683646_Effectiveness_of_E-learning_course_materials_for_learning_database_management_systems_\(vizitat_9.07.2016\)](https://www.researchgate.net/publication/285683646_Effectiveness_of_E-learning_course_materials_for_learning_database_management_systems_(vizitat_9.07.2016))
52. Arbaugh J.B., Fich R.B. The importance of participant interaction in online environments. In: Decision Support Systems, 2007, vol. 43(2), p. 853-865. <http://www.sciencedirect.com/science/article/pii/> (vizitat 2.08.2016)

53. Gilbert J. E-Learning: The Student Experience. In: British Journal of Educational Technology, 2007, vol. 38(4), p. 560-573. <https://pdfs.semanticscholar.org/7079/a686cc33e91e8141d9887de0225e87929f58.pdf> (vizitat 2.08.2016)
54. We Hu et al. Collaborative Web – Based E-Learning Environment for Information Security Curriculum. In: World Academy of Science Engineering and Technology, 2009, vol. 53. <http://www.cs.zju.edu.cn/people/ctz/203.pdf> (vizitat 2.08.2016)
55. Doberkat E.-E. et al. Software Engineering and eLearning: The MuSoft Project. eLeed, Iss. 2005. <https://eLeed.campussource.de/archive/2/201> (vizitat 2.08.2016)
56. Corbera F. et al. Development of a new MOODLE module for a basic course on computer architecture. In: ACM SIGCSE Bulletin-ITiCSE '08, 2008, vol. 40(3), p. 349-349. <http://dl.acm.org/citation.cfm?id=1384391> (vizitat 8.08.2016)
57. Riabov V.V., Higgs B.J. Software Tools and Virtual Labs in Online computer-Science Classes. In: Learning Management System Technologies and Software Solutions for Online Teaching: Tools and Applications. Ed. Yefim Kats, IGI Global, 2010, p. 332-350. https://www.researchgate.net/publication/260038613_Software_Tools_and_Virtual_Labs_in_Online_Computer-Science_Classes (vizitat 8.08.2016)
58. Huan X. et al. Teaching computer science courses in distance learning. In: Journal of Instructional Pedagogies, 2011, vol. 6, p. 47-60. <https://eric.ed.gov/?id=EJ1097042> (vizitat 2.08.2016)
59. Palumbo M., Verga F. Creation of an integrated environment to supply e-learning platforms with Office Automation features. In: Interactive Learning Environment, 2013, vol. 23(6), p. 766-777. <http://www.tandfonline.com/doi/abs/10.1080/10494820.2013> (vizitat 2.08.2016)
60. Ramos J. et al. E-assessment of Matlab assignments in Moodle: Application to an introductory programming course for engineers. In: Computer Applications in Engineering Education, 2013, vol. 21(4), p. 728-736. https://www.researchgate.net/publication/227726027_E-Assessment_of_Matlab_Assignments_in_Moodle_Application_to_an_Introductory_Programming_Course_for_Engineers (vizitat 12.08.2016)
61. Sevgi O., Koseler R. Multi-Dimensional Evaluation of E-Learning Systems in the Higher Education Context: An Empirical Investigation of a Computer Literacy Course. In: 39th ASEE/IEEE Frontiers in Education Conference W1C-1, 2009. <http://archive.fie-conference.org/fie2009/papers/1200.pdf> (vizitat 12.08.2016)

62. Tripathi A., Mishra A. A Web-based E-Learning Environment for Information Security. In: International Journal of Computer Applications, 2012, vol. 45(4), p. 50-54. <http://research.ijcaonline.org/volume45/number4/pxc3879058.pdf> (vizitat 12.08.2016)
63. Martin F. Blackboard as the Learning Management System of a Computer Literacy Course. In: MERLOT Journal of Online Learning and Teaching, 2008, vol. 4(2). <http://jolt.merlot.org/vol4no2/martin0608.pdf> (vizitat 15.08.2016)
64. Pat P.W., Chan M. A study on the methods of assessment and strategy of knowledge sharing in computer course. In: The materials of the International Conference e-Learning, 2014, p. 141-148. <http://files.eric.ed.gov/fulltext/ED557251.pdf> (vizitat 15.08.2016)
65. Akingbade A. et al. JAWAA: easy web-based animation from CS 0 to advanced CS courses. In: ACM SIGCSE Bulletin, 2003. <http://citeseerx.ist.psu.edu> (vizitat 15.08.2016)
66. Al-Khanjari Z. et al. An extended e-Learning system architecture: integrating software tools within the e-Learning portal. The International Arab Journal of Information Technology, 2006, vol. 3(1). <http://ccis2k.org/iajit/PDF/vol.3,no.1/12-Zhoor.pdf> (vizitat 5.08.2016)
67. Schwieren J. et al. Using software testing techniques for efficient handling of programming exercises in an e-learning platform. In: The Electronic Journal of e-Learning, 2006, vol. 4(1), p. 87-94. <http://ejel-volume4-issue1-article17.pdf> (vizitat 5.08.2016)
68. Sodoké K. et al. Extending Moodle functionalities to adaptive testing framework. In: The materials of the World Conference on E-Learning in Corporate, Government Healthcare and Higher Education, 2007, p. 476-482. [https://www.academia.edu/11731103/Extending Moodle functionalities to adaptive testing framework](https://www.academia.edu/11731103/Extending_Moodle_functionalities_to_adaptive_testing_framework) (vizitat 15.08.2016)
69. Georgantaki S., Retalis S. Using educational tools for teaching object oriented design and programming. In: Journal of Information Technology Impact, 2007, vol. 7(2), p. 111-130. [https://www.academia.edu/3687655/Using Educational Tools for Teaching Object Oriented Design and Programming](https://www.academia.edu/3687655/Using_Educational_Tools_for_Teaching_Object_Oriented_Design_and_Programming) (vizitat 15.08.2016)
70. Queirós R. et al. Integration of e-Portfolios in Learning Management Systems. In: Murgante B. et al. (Eds.). Computational Science and Its Applications – ICCSA 2011. International Conference, Santander, Spain, June 20-23, 2011, p. 500-510. <https://pdfs.semanticscholar.org/2471/2bb8e60e47d4e254ffac817cb9b2b61f6f30.pdf> (vizitat 25.08.2016)
71. Rößling G., Kothe A. Extending Moodle to Better Support Computing Education. In: The materials of the 14th annual ACM SIGCSE Conference on Innovation and Technology in

- Computer Science Education, New York, 2009, p. 146-150.
[https://pdfs.semanticscholar.org/2471/\(vizitat 25.08.2016\)](https://pdfs.semanticscholar.org/2471/(vizitat%2025.08.2016))
72. Красильникова В.А. Оценка учебных достижений в условиях компетентностного подхода. http://conference.osu.ru/assets/files/conf_info/conf8/s15.pdf (vizitat 5.09.2016)
73. Кобылкин Д.С. Информационно-коммуникационные технологии при изучении информатики как средство мотивации учебной и научной деятельности у бакалавров технических специальностей. http://conference.osu.ru/assets/files/conf_info/conf8/s15.pdf (vizitat 5.09.2016)
74. Гладышева Ю.А. Особенности применения информационных технологий в образовательном процессе вуза.
http://conference.osu.ru/assets/files/conf_info/conf8/s15.pdf (vizitat 5.09.2016)
75. Заводчикова Н.И., Плясунова У.В. Особенности методики обучения информатике с использованием дистанционной среды Moodle. В: Ярославский педагогический вестник, 2015, № 5, с. 134-138. http://vestnik.yspu.org/releases/2015_5/28.pdf (vizitat 5.09.2016)
76. Позняк Ю.В. и др. Возможности системы Moodle и актуальность ее применения в сфере образования. В: Материалы Республиканской научно-практической конференции «Инновационные технологии в образовании, науке и производстве», Минск, 6-7 декабря 2007 г., Минск: БНТУ, с. 156-157.
http://elib.bsu.by/bitstream/123456789/3591/1/Vozmozn_Moodle.pdf (vizitat 5.09.2016)
77. Bleandură N. Situații de învățare în medii digitale: proiectare și utilizare. În: Formarea universitară în medii digitale: cercetări teoretico-experimentale. Bălți, 2015, p. 81-107.
78. Scutelnic O. Diferențierea instruirii studenților în procesul studierii cursului universitar de informatică. Autoreferatul tezei de doctor în pedagogie. Chișinău, 2013. 24 p.
<http://dspace.usarb.md:8080/jspui/bitstream/123456789/1732/1/Autoreferat%20Scutelnic.pdf>
79. Vovnenciuc O. Formarea și dezvoltarea la studenți a competenței de învățare în medii digitale. În: Formarea universitară în medii digitale: cercetări teoretico-experimentale. Bălți, 2015, p. 108-127.
80. Deinego N. Formarea și dezvoltarea competențelor de programare: dirigarea învățării prin evaluări computerizate. În: Formarea universitară în medii digitale: cercetări teoretico-experimentale. Bălți, 2015, p. 128-154.

81. Corlat S. Integrarea portofoliilor de predare și învățare. Metaportofoliul. În: Didactica Pro..., revistă de teorie și practică educațională a Centrului Educațional PRO DIDACTICA. 2010, nr. 6(64), p. 8-12.
82. Corlat S. ș.a. Metodologia utilizării tehnologiilor informaționale și de comunicație în învățământul superior. Chișinău: UST, 2011. 204 p.
83. Gîncu S. Dezvoltarea competenței de programare orientată pe obiecte din perspectiva instruirii în medii digitale. În: Formarea universitară în medii digitale: cercetări teoretico-experimentale. Bălți, 2015, p. 155-171.
84. Braicov A., Velicova T. Organizarea evaluării cu ajutorul LCMS Moodle și a altor produse soft integrabile cu el. În: The materials of the 20th Conference on applied and industrial mathematics: Dedicated to Academician Mitrofan M. Ciobanu, Chișinău, 22-25 august, 2012, p. 138-142.
85. Gremalschi A. Modernizarea învățământului preuniversitar prin implementarea pe scară largă a tehnologiei informației și a comunicațiilor. În: Didactica Pro..., revistă de teorie și practică educațională, nr. 6(64), 2010, p. 2-5.
86. Chiriac L., Globa A., Bobeică N. Procedee metodice privind pregătirea și desfășurarea olimpiadelor de informatică. Mathematics & Information Technologies: Research and Education (MITRE 2011) dedicated to the 65th anniversary of the Moldova State University. Chișinău, august 22-25, 2011, p. 168-169.
87. Cabac V. Cabac E. Experiența de evaluare a cursurilor electronice. În: Învățământul superior din Republica Moldova la 85 de ani. Materialele conferinței științifice naționale cu participare internațională. V.III. Probleme actuale ale didacticii științelor. Chișinău: Universitatea de Stat din Tiraspol, 24-25 septembrie, 2015, p. 111-116.
88. Mihălache L. Abordări metodice privind aplicarea complexă a tehnologiilor computaționale în procesul de predare-învățare a compartimentului „Modelare și metode de calcul” în cursul liceal de informatică. Autoreferatul tezei de doctor în pedagogie. Chișinău: UST, 2013. 26 p.
89. Globa A. Abordări metodice privind implementarea noilor tehnologii informaționale în procesul de studiere a disciplinei universitare „Tehnici de programare”. Autoreferatul tezei de doctor în pedagogie. Chișinău: UST, 2016. 26 p.
90. Fojtík R., Habiballa H. E-Learning in Computer Science Education. In: The materials of the Conference “E-Learning and the Knowledge Society”. Brusel, 09.05.2004, p. 49-53. ISBN 90-70963-89-2. <http://www1.osu.cz/~fojtik/doc/ecet04.pdf> (vizitat 27.01.2017).

91. Fjuk A., Karahasanovic A., Kaasboll J. Comprehensive Object Oriented Learning: the Learner's Perspective. In: Informing Science Press, Santa Rosa, CA, 2006. 226 p. <https://books.google.md/Comprehensive+Object+Oriented+Learning> (vizitat 27.04.2017).
92. Fleury A.E. Encapsulation and reuse as viewed by java students. In: ACM SIGCSE Bulletin, 2001, nr. 33(1), p. 189-193.
<http://ftp.math.utah.edu/pub/tex/bib/sigcse2000.html> (vizitat 03.05.17)
93. Ragonis N., Ben-Ari M. A Long-Term Investigation of the Comprehension of OOP Concepts by Novices. In: Computer Science Education, 2005, nr. 5(3), p. 203-221.
<http://cormack.uwaterloo.ca/papers/t4v336x210720108.pdf>. (vizitat 2.05.2017)
94. Schulte C., Niere J. Thinking in Object Structures: Teaching Modeling in Secondary Schools. In: The materials of the 16th European Conference on Object-Oriented Programming, 6th Workshop on Pedagogies and Tools for the Teaching and Learning of Object Oriented Concepts, Malaga, Spain, 2002.
http://www.fujaba.de/uploads/tx_sibibtex/PTLOOC2002.pdf. (vizitat 2.05.2017)
95. Luker P.A. There's more to oop than syntax! In: ACM SIGCSE Bulletin, 1994, nr. 26(1), p. 56-60. <https://www.deepdyve.com/lp/association-for-computing-machinery> (vizitat 03.05.2017)
96. **Gasnaș A. Programarea orientată pe obiecte: particularități de predare. În: Acta et Commentationes. Științe ale Educației. Revistă științifică, nr. 2(5), 2014, p. 41-45. ISSN 1857-0623. Chișinău: Universitatea de Stat din Tiraspol.**
97. Brusilovsky P. et al. Teaching programming to novices: a review of approaches and tools. In: Proceedings of ED-MEDIA 94-World Conference on Educational Multimedia and Hypermedia. Vancouver, Canada, 1994, 25-30 June, p. 103-110.
<http://files.eric.ed.gov/fulltext/ED388228.pdf> (vizitat 03.05.2017)
98. Brusilovsky P. et al. Minilanguages: a way to learn programming principles. In: Education and Information Technologies, 1997, nr. 2(1), p. 65-83.
https://www.researchgate.net/publication/227186529_Mini-languages (vizitat 03.05.2017)
99. **Gasnaș A., Chiriac L. Reflecții asupra particularităților și strategiilor de predare a limbajelor orientate pe obiect. Probleme actuale ale didacticii științelor reale. Conferința științifico-didactică națională cu participare internațională consacrată aniversării a 80-a de la nașterea profesorului universitar Andrei Hariton. Chișinău, 4-6 octombrie, 2013, p. 113-116.**
100. Bennedsen J. Teaching and Learning Introductory Programming: A Model-Based Approach, 227 p. <https://www.duo.uio.no/bennedsen.pdf?sequence=1> (vizitat 04.05.2017)

101. Cerghit I. Sisteme de instruire alternative și complementare: structuri, stiluri și strategii. Ediția a II-a, revăzută și adăugată. Iași: Editura Polirom, 2008. 395 p.
102. Hubscher-Younger T., Narayanan N.H. Constructive and collaborative learning of algorithms. In: Proceedings of the 34th SIGCSE technical symposium on computer science education, 1-58113-648-X, 6-10, Reno, NV, 2003. <http://citeseerx.ist.psu.edu/viewdoc/> (vizitat 04.05.2017)
103. Lupu I., Cabac V., Gîncu S. Formarea și dezvoltarea competenței de programare orientată pe obiecte la viitorii profesori de informatică. Chișinău: UST, 2013. 150 p.
104. Bîrzea C. Artă și știința educației. București: Editura Didactică și Pedagogică, 1995. 218 p.
105. Дахин А.Н. Моделирование в педагогике: попытка осмысления. 2007. <http://www.bestreferat.ru/referat-78582.html> (vizitat 19.04.2017)
106. Iucu R.B. Instruirea școlară – perspective teoretice și aplicative. Iași: Polirom, 2001.
107. Teoria și metodologia instruirii. http://www.upm.ro/metodologia_instr.pdf (vizitat 08.05.2017)
108. Conole G. et al. Mapping pedagogy and tools for effective learning design. In: Computers and Education, 2004. <https://pdfs.semanticscholar.org/be05/e40c05977b81a873f5abd03e5216a5a20141.pdf> (vizitat 19.04.2017)
109. Mayes T., De Freitas S. Review of e-Learning frameworks, models and theories. In: JISC e-learning models desk study, 2004. <http://www.jisc.ac.uk/epedagogy> (vizitat 15.05.2017).
110. Merrill D.M. First Principles of Instruction, In: ETR&D, 2002, vol. 50, no. 3, p. 43–59. <http://mdavidmerrill.com/Papers/firstprinciplesbymerrill.pdf> (vizitat 05.05.2017).
111. Wikipedia.org. [https://ro.wikipedia.org/wiki/Schema_\(psihologie_cognitiv%C4%83\)](https://ro.wikipedia.org/wiki/Schema_(psihologie_cognitiv%C4%83)) (vizitat 05.05.2017).
112. Jonassen D.H., Peck K.L. Wilson B.G. Learning with Technology: A Constructivist Perspective. In: Upper Saddle River, NJ: Merrill Publishing, 1999. <http://www.aect.org/edtech/ed1/pdf/24.pdf> (vizitat 15.04.2017)
113. Salmon G. E-moderating: the key to teaching and learning online. London: Kogan Press, 2003. <http://www.ijede.ca/index.php/jde/article/viewFile/439/381> (vizitat 09.05.2017)
114. Tait A. Reflections on student support in open and distance learning. In: International Review of Research in Open and Distance Learning, 2003, vol. 4, no. 1 Editorial. <http://oro.open.ac.uk/1017/1/604.pdf> (vizitat 16.05.2017)
115. EduTechWiki, http://edutechwiki.unige.ch/en/Instructional_design (vzitat 03.06.2017)

116. Design instructional. http://consilr.info.uaic.ro/uploads_lt4el/resources/htmlrom (vizitat 05.06.2017)
117. Backwards Design. <https://canvas.instructure.com/courses/803402/pages/backwards-design> (vizitat 06.06.2017)
118. Deinego N. Testarea adaptivă ca factor de optimizare a procesului de instruire în învățământul universitar. Teză de doctor în pedagogie. Teoria și metodologia instruirii (informatica). Bălți, 2009. 173 p.
119. Cerghit I. Sisteme de instruire alternative și complementare: structuri, stiluri și strategii. Ediția a II-a, revăzută și adăugată. Iași: Editura Polirom, 2008. 395 p.
120. Cristea S. Dicționar de termeni pedagogici. București: Editura Didactică și Pedagogică, 1998. 422 p.
121. Nicola I. Tratat de pedagogie școlară. București: Editura Aramis, 2003. 442 p.
122. Mialaret G. Introducere în pedagogie. București: Editura Didactică și Pedagogică, 1979. 442 p.
123. Ianovici N., Frenț A.O. Metode didactice de predare, învățare și evaluare la biologie. Timișoara: Mirton, 2009. 167 p.
124. Oprea C.L. Strategii didactice interactive. București: Editura Didactică și Pedagogică, 2006. 414 p.
125. Pedagogie II. Teoria și metodologia instruirii. Teoria și metodologia evaluării. <http://www.umfcv.ro/files/p/e/Pedagogie%202.pdf> (vizitat 29.05.2017)
126. Neacșu Ioan. Instruire și învățare. Ediția a II-a, revizuită. București: Editura Didactică și Pedagogică, 1999. 283 p.
127. Cerghit I. Metode de învățământ. București: Editura Didactică și Pedagogică, 1980. 247 p.
128. Ionescu M. Instrucție și educație. Ediția a II-a. Arad: "Vasile Goldiș" University Press, 2005. 232 p.
129. Metode interactive de predare, învățare, evaluare. <https://www.academia.edu/11825268> (vizitat 19.06.2017)
130. Sorensen O.J. Învățarea bazată pe probleme. <http://www.pblmd.aau.dk/fileadmin/> (vizitat 06.06.2017)
131. Instructional Design Models and Theories: Inquiry-based Learning Model. <https://elearningindustry.com/inquiry-based-learning-model> (vizitat 08.06.2017)
132. Studiul de caz – metodă eficientă de dezvoltare a competențelor comunicative. <http://limbaromana.md/index.php?go=articole&n=1666> (vizitat 06.06.2017)

133. Silistraru N., Golubișchi S. Pedagogia învățământului superior: ghid metodologic. Chișinău: Universitatea de Stat din Tiraspol, 2013. 206 p.
134. Cucoș C. Pedagogie. Ediția a II-a, revăzută și adăugată. Iași: Polirom, 2006. 462 p.
135. **Gasnaș A. Rolul LMS Moodle în predarea-evaluarea cursului „Programarea orientată pe obiecte”. În: Materialele Conferinței „Mahematics & Information Technologies: Research and Education” (MITRE 2016), dedicated to the 70th anniversary of the Moldova State University, Chișinău, 23-26 iunie, 2016, p. 101.**
136. **Gasnaș A., Braicov A. Un model pedagogic centrat pe implementarea Sistemelor de Management al Învățării în procesul de studiu al Programării Orientate pe Obiecte. The conference on Applied and Industrial Mathematics. Iași, România, 15-16 septembrie 2018. p.93-95**
137. Skinner B.F. Revoluția științifică a învățământului. București: EDP, 1971. 224 p.
138. Panțuru S., Păcurar D. Didactica. Brașov: Editura Universității “Transilvania”, 1999.
139. Spady W.G. Outcome-Based Education: Critical Issues and Answers. Arlington: American Association of School Administrators. 212 p.
140. Rusu M. Modelul clasic și modern în sistemul de învățământ românesc. Influența mijloacelor TIC asupra procesului de predare-învățare-evaluare <http://ccdmures.ro/cmsmadesimple/uploads/file/rev8sp/lbrom6.pdf> (vizitat 23.06.2017)
141. Bonk C.J., Graham C.R. The handbook of blended learning: global perspectives, local designs. Pfeiffer, 2006.
142. Blended Learning – Wikipedia. https://en.wikipedia.org/wiki/Blended_learning (vizitat 25.06.2017)
143. Biggs J.B. Teaching for quality learning at university. Buckingham: Society for Research in Higher Education, 1999.
144. Corlat S. Aspecte generale de instruire la distanță în învățământul superior. https://ibn.idsi.md/ro/vizualizare_articol/48032 (vizitat 23.06.2017)
145. www.edutechie.ws/2007/10/09/cone-of-experience-media (vizitat 23.06.2017)
146. Dougiamas M. Social constructionism or social constructivism forum. <https://moodle.org/mod/forum/discuss.php?d=198733&mode=3> (vizitat 23.06.2017)
147. Bates A.W. Managing Technological Change: Strategies for College and University. San Francisco: Leaders Jossey-Bass, 2000. 320 p.
148. CarteMoodle. https://edu.moodle.ro/pluginfile.php/16465/mod_forum/intro/ (vizitat 9.06.2017).

149. Moodle 1.9 Teaching Techniques. <http://moodle.eibfs.ae/pluginfile.php> (vizitat 10.06.2017)
150. wikipedia.org. <https://en.wikipedia.org/wiki/> (vizitat 30.06.2017)
151. Bhat A. LMS Wiki: Unharnessed Potential. <https://www.upsidelms.com/blog/lms-wikis-unharnessed-potential/> (vizitat 23.06.2017)
152. WIRIS. <https://en.wikipedia.org/wiki/> (vizitat 01.06.2017)
153. Hot_Potatoes. https://en.wikipedia.org/wiki/Hot_Potatoes (vizitat 01.06.2017)
154. Audacity. [https://en.wikipedia.org/wiki/Audacity_\(audio_editor\)](https://en.wikipedia.org/wiki/Audacity_(audio_editor)) (vizitat 01.06.2017)
155. Dictionnaire de pédagogie. Paris: Larousse, Bordas, 1996.
156. Potolea D., Manolescu M. Teoria și practica evaluării educaționale. București: MEN-PIR, 2005. 248 p.
157. www.studfiles.ru/preview/5183124/page:73/ (vizitat 01.06.2017)
158. Meyer Genevieve. De ce și cum evaluăm. Iași: Editura Polirom, 2000.
159. **Gasnaș A. Rolul evaluării în procesul de studiu al programării orientate pe obiecte. Conferința științifico-didactică națională cu participare internațională ediția a II-a, *Probleme actuale ale didacticii științelor reale*, consacrată aniversării a 80-a a profesorului universitar Ilie Lupu. Chișinău, 11–12 mai, 2018.**
160. Duit R., Tesch M. On the role of the experiment in science teaching and learning – Visions and the reality of instructional practice. (Eds.) In: Proceedings of the 7th International Conference on Hands-on Science, 25-31 July 2010, Rethymno-Crete, p. 17–30. <http://www.clab.edc.uoc.gr/HSci2010> (vizitat 12.03.2017)
161. Bocos M. Metodologia cercetării pedagogice. <http://www.bp-soroca.md/pdf/cercetare%20pedag.pdf> (vizitat 12.03.2017).
162. **Gasnaș A. Determinarea nivelului de pregătire al studenților pentru utilizarea platformelor LMS. În: Materialele Conferinței științifice internaționale “The Fourth Conference of Mathematical Society of the Republic of Moldova. Dedicated to the centenary of Vladimir Andrunachevici”. Chișinău, 28 iunie –2 iulie, 2017, p. 135-143.**
163. Hotărârea nr. 944 din 14.11.2014 cu privire la aprobarea Strategiei de dezvoltare a educației pentru anii 2014–2020 „Educația–2020”. <http://lex.justice.md/viewdoc.php?action=view&view=doc&id=355494&lang=1> (vizitat 15.03.2015)
164. Publication Manual of the American Psychological Association, Sixth edition. Washington, DC: American Psychological Association, 2010. 272 p.
165. Statistică socială și SPSS – ghid pentru curs practic.

http://fspac.ubbcluj.ro/comunicare/wp-content/uploads/2013/02/Statistic_social_-_i-SPSS.pdf (vizitat 27.03.2017)

166. Labăr A.V. SPSS pentru științele educației. Iași: Polirom, 2008. 347 p.
167. Opariuc-Dan C. Statistica aplicată în științele socioumane. Analiza asocierilor și a diferențelor statistice. Cluj-Napoca: Editura ASCR, 2011. 373 p.
168. Recommendation of the european parliament and of the council<http://eur-lex.europa.eu/legal-content/RO/TXT/?uri=celex:32006H0962>, (vizitat 15.03.2015)
169. **Braicov A., Gasnaș A. Metode didactice interactive pentru studierea programării orientate pe obiecte. În: Didactica Pro. Revistă de teorie și practică educațională a Centrului Educațional PRO DIDACTICA. Nr.1 (107), 2018. p.32-37.**
170. Dumitriu C. Introducere în cercetarea psihopedagogică. București: Editura Didactică și Pedagogică, R.A., 2004. 230 p.
171. Сидоренко Е. Методы математической обработки в психологии. Москва: Речь, 2010. 350 с.
172. UNESCO. ICT in Education, Performance Indicators on ICT Use in Education project. URL. <http://www.unescobkk.org/education/ict/ict-in-education-project/monitoring-andmeasuring-change> (vizitat 30.05.2017)
173. Cadrul Strategic ET 2020. http://ec.europa.eu/education/policy/strategic-framework_ro
174. Strategia „Educația 2020”. Hotărârea Guvernului Republicii Moldova nr. 944 din 14.11.2014. În: Monitorul Oficial al Republicii Moldova, nr. 345-351 din 21.11.2014, art. nr. 1014 (vizitat 30.05.2017).
175. Strategia Națională de dezvoltare a societății informaționale „Moldova Digitală 2020”, aprobată prin Hotărârea Guvernului Republicii Moldova nr. 857 din 31.10.2013. În: Monitorul Oficial al Republicii Moldova, nr. 252-257 din 08.11.2013. <http://lex.justice.md/md/350246> (vizitat 01.06.2017).
176. Programul național de securitate cibernetică a Republicii Moldova pentru anii 2016–2020. <http://www.mtic.gov.md/ro/projects/programul-de-securitate-cibernetica> (vizitat 01.06.2017)

ANEXE

Anexa 1. Chestionar

Chestionar privind opiniile studenților vizavi de utilizarea TIC, inclusiv a SMÎ, în procesul de predare-învățare-evaluare

Citiți atent enunțurile și încercați variantele corecte de răspuns.

1. Folosiți calculatorul:

- ☐ acasă
- ☐ la locul de muncă
- ☐ fără restricții de locație (calculator portabil)
- ☐ nu folosesc deloc

2. Selectați gradul de dificultate în utilizarea TIC în activitatea academică:

- ☐ nu am dificultăți
- ☐ dificultăți minore
- ☐ dificultăți moderate
- ☐ dificultăți mari
- ☐ nu utilizez TIC

3. Indicați tipurile de dispozitive digitale pe care le utilizați în activitatea academică:

- ☐ calculator (desktop)
- ☐ laptop
- ☐ tabletă
- ☐ proiector video
- ☐ tabla interactivă
- ☐ telefon
- ☐ imprimantă
- ☐ scaner
- ☐ cameră video
- ☐ nu utilizez dispozitive digitale
- ☐ altele (*enumerați*): _____

4. Folosiți comunicarea prin Internet pentru:

1. schimb de mesaje prin poșta electronică
2. discuții prin mesagerie instantanee (text)
3. convorbiri audio
4. convorbiri video
5. discuții în rețelele de socializare
6. Skype
7. altele (*enumerați*): _____

5. Ați folosit vreun mediu virtual de predare-învățare-evaluare?
1. Da
 2. Nu
 3. Nu știu/Nu pot răspunde
6. În cazul în care la întrebarea 5 ați răspuns *Da*, scrieți denumirea mediului/mediilor virtual(e) utilizat(e).
1. _____
 2. _____
 3. _____
7. Care este gradul Dvs. de interes pentru tehnologiile informaționale și de comunicație (TIC)?
1. foarte mare
 2. mare
 3. redus
 4. foarte redus
 5. deloc interesat
8. Selectați, prin bifare, activitățile în care ați utilizat TIC:
- ☐ elaborarea referatelor
 - ☐ elaborarea prezentărilor
 - ☐ căutarea informației pe Internet
 - ☐ rezolvarea problemelor cu ajutorul softurilor specializate
 - ☐ realizarea proiectelor investigaționale
 - ☐ n-am utilizat
 - ☐ altele (*enumerați*): _____
9. Inițiativa de a folosi TIC:
1. îmi aparține
 2. aparține profesorului
10. Cât de importantă considerați utilizarea TIC în procesul de învățare-evaluare?
1. foarte importantă
 2. importantă
 3. mai puțin importantă
 4. deloc importantă
11. Aveți blog sau site personal?
1. Nu, nici nu cred că e necesar.
 2. Nu, dar planific să-mi creez.
 3. Am blog/site.
12. Selectați instrumentele TIC pe care le-ați utilizat în activitatea academică.

- ☐ editoare de texte (de exemplu, Microsoft Word)
- ☐ aplicații de prezentări electronice (de exemplu, Microsoft Power Point)
- ☐ aplicații de calcul tabelar (de exemplu, Microsoft Excel)
- ☐ softuri specializate (de exemplu, Maple, AutoCad, 1C Contabilitate)
- ☐ platforme de învățare (de exemplu, MOODLE)
- ☐ instrumente Web (poșta electronică, bloguri, rețele de socializare, portofolii electronice, site-uri wiki etc.)
- ☐ nu utilizez
- ☐ altele (*enumerați*): _____

13. Selectați instrumentele Google pe care le-ați utilizat:

- a) Gmail
- b) Google docs
- c) Google sites
- d) Google translate
- e) Google forms
- f) Google maps
- g) Google photo
- h) Google drive
- i) Google calendar
- j) Google books
- k) Google scholar
- l) nu utilizez
- m) altele (*enumerați*): _____

14. Selectați gradul de obiectivitate (în opinia Dvs) al evaluării asistate de calculator:

- 1. foarte obiectiv
- 2. obiectiv
- 3. puțin obiectiv
- 4. deloc obiectiv

15. Scrieți care sunt, în opinia Dvs., *avantajele* utilizării TIC în educație:

16. Scrieți care sunt, în opinia Dvs., *dezavantajele* utilizării TIC în educație:

Test de evaluare inițială

(Numele și prenumele studentului)

(Grupa)

(Data)

Citiți atent enunțurile și alegeți variantele corecte de răspuns.

1. Selectați (bifând) cuvintele-cheie cu ajutorul cărora în limbajul C se declară constantele:

- ☐ `define`
- ☐ `const`
- ☐ `volatile`
- ☐ `include`
- ☐ `default`

2. Alegeți declarațiile scrise corect în limbajul C:

- ☐ `int a[100];`
- ☐ `char a[1..100];`
- ☐ `float a[1.10..10.00];`
- ☐ `char a[100];`
- ☐ `float a[12,5];`

3. Elementele unui tablou pot fi de tip:

- ☐ `integer`
- ☐ `real`
- ☐ `char`
- ☐ `boolean`

4. În ce cazuri este necesar de a utiliza instrucțiunea `return` în corpul funcției?

- ☐ întotdeauna
- ☐ în cazul în care e necesar ca funcția să returneze o valoare
- ☐ pentru a asigura ieșirea din funcție într-o poziție arbitrară
- ☐ în cazul în care este indicat tipul valorii returnate, inclusiv `void`

5. Rolul funcției `strlen(char *sir)` este:

- a. de a returna lungimea unui șir
- b. de a converti valoarea șirului la un număr
- c. de a șterge un caracter din șir
- d. de a introduce un caracter în șirul de caractere

6. Reprezentarea corectă a expresiei $0 < x < 10$ în limbajul C este:

- ☐ $x > 0, x \leq 10$
- ☐ $0 < x \leq 10$
- ☐ $x > 0 \text{ AND } x \leq 10$
- ☐ $(x > 0) \text{ AND } (x < 10)$

7. Selectați tipurile de date simple:

- ☐ întreg
- ☐ structură
- ☐ real (float, double)
- ☐ caracter
- ☐ pointer
- ☐ tablou
- ☐ enumerare

8. Fie instrucțiunile:

```
strncpy(s, strstr("examen", "am"), 4);  
s[4]='\0';
```

Ce valoare are variabila *s* de tip șir de caractere după executarea lor?

- ☐ exam
- ☐ amen
- ☐ menn
- ☐ men

9. Subprogramul *minim* primește prin parametrii *x* și *y* două numere întregi și returnează, prin parametrul *m*, cea mai mică dintre cele două valori *x* și *y*. Antetul corect al acestui subprogram este:

- ☐ `int minim(int x, int y, int m)`
- ☐ `void minim(int x, int y, int &m)`
- ☐ `int minim(int x, int y)`
- ☐ `void minim(int x, int y, int m)`

Barem de notare

Item	1	2	3	4	5	6	7	8	9
Punctaj maxim	1	1	1	1	1	1	1	1	2

Itemul 1: se acordă 1 punct pentru răspunsul corect și 0,1 puncte penalizare pentru încă un alt răspuns greșit. În cazul răspunsului greșit se vor acorda 0 puncte.

Itemul 2: se acordă 1 punct pentru răspunsul corect și 0,1 puncte penalizare pentru încă un alt răspuns greșit. În cazul răspunsului greșit se vor acorda 0 puncte.

Itemul 3 se acordă 1 punct pentru răspunsul corect și 0,1 puncte penalizare pentru încă un alt răspuns greșit. În cazul răspunsului greșit se vor acorda 0 puncte.

Itemul 4: se acordă 1 punct pentru răspunsul corect și 0,1 puncte penalizare pentru încă un alt răspuns greșit. În cazul răspunsului greșit se vor acorda 0 puncte.

Itemul 5: se acordă 1 punct pentru răspunsul corect și 0,1 puncte penalizare pentru încă un alt răspuns greșit. În cazul răspunsului greșit se vor acorda 0 puncte.

Itemul 6: se acordă 1 punct pentru răspunsul corect și 0,1 puncte penalizare pentru încă un alt răspuns greșit. În cazul răspunsului greșit se vor acorda 0 puncte.

Itemul 7: se acordă 1 punct pentru răspunsul corect și 0,1 puncte penalizare pentru încă un alt răspuns greșit. În cazul răspunsului greșit se vor acorda 0 puncte.

Itemul 8: se acordă 1 punct pentru răspunsul corect și 0,1 puncte penalizare pentru încă un alt răspuns greșit. În cazul răspunsului greșit se vor acorda 0 puncte.

Itemul 9: se acordă 2 puncte pentru răspunsul corect și 0,1 puncte penalizare pentru încă un alt răspuns greșit. În cazul răspunsului greșit se vor acorda 0 puncte.

Notă	4	5	6	7	8	9	10
Punctaj	3,1-4,0	4,1-5,0	5,1-6,0	6,1-7,0	7,1-8,0	8,1-9,0	9,1-10

Teste de evaluare sumativă Programarea Orientată pe Obiecte

Testul 1

1. În scopul utilizării instrucțiunilor `cin` și `cout`, selectați fișierul header potrivit:
 - a) `string.h`
 - b) `<string.h>`
 - c) `"string.h"`
 - d) `<iomanip.h>`
 - e) `<iostream.h>`
2. Alegeți cuvântul-cheie utilizat pentru a indica faptul că o funcție nu returnează nicio valoare:
 - a) `void`
 - b) `null`
 - c) `undefined`
 - d) `empty`
 - e) `nothing`
3. Selectați mecanismul de inițializare a obiectelor în C++:
 - a) Constructor
 - b) `ObjBuild`
 - c) Funcții membre
 - d) `InitObj`
 - e) `ObjInit`
4. Alegeți propoziția care face diferența dintre o funcție membru și un constructor.
 - a) O funcție membru poate defini valori, pe când un constructor nu poate.
 - b) Un constructor poate declara valori, iar o funcție membru nu poate.
 - c) O funcție membru poate returna valori, iar un constructor nu poate.
 - d) Un constructor poate defini valori, iar o funcție membru nu poate.
 - e) Un constructor poate returna valori, iar o funcție membru nu poate.
5. Examinați codul de mai jos și alegeți propoziția corectă

```
main() {
    int *p;
    int q;
    p = new int;
    delete q;
    return 0 ;
}
```

 - a) Pentru ca acest cod să funcționeze corect, `p` trebuie să fie declarat de tip `float`.
 - b) `p` nu poate fi inițializat ca `new int`, deoarece `p` este un pointer de tip întreg.
 - c) `int q;` este o declarație incorectă.
 - d) `q` nu poate fi șters, deoarece `q` nu este pointer.
 - e) Pentru ca acest cod să funcționeze corect, `p` trebuie să fie declarat de tip `double`.

Selectați declarația corectă pentru un pointer constant la date nonconstante.

- a) `const char * pc`
- b) `char * pc`
- c) `char * const pc`
- d) `const char * const pc`
- e) Este imposibil de a avea pointer "const".

6. Ce reprezintă aceste trei modalități de scriere a constructorului în codul de mai jos? Alegeți afirmația corectă.

```
class comun {  
    int a, b, c;  
    public:  
        comun() {a=b=c=0;}  
        comun(int x) {a=b=c=x;}  
        comun(int x, int y, int z) {  
            a=x;  
            b=y;  
            c=z;  
        }  
};
```

- a) Supraîncărcarea constructorului
- b) Încapsularea constructorului
- c) Polimorfismul constructorului
- d) Moștenirea constructorului

7. Selectați afirmațiile false.

- a) Într-o clasă, e necesar de declarat în mod explicit constructorul și destructorul.
- b) Într-o clasă specificată, constructorii sunt întotdeauna necesari, iar destructorul nu.
- c) Constructorii pot avea parametri, iar destructorii nu.
- d) Este incorect de definit un constructor sau destructor ca fiind virtual.
- e) Atât constructorul, cât și destructorul pot avea parametri.

8. Selectați afirmațiile adevărate.

- a) Apelul constructorului se realizează la declararea unui obiect.
- b) Un constructor nu poate returna valoare.
- c) Un constructor poate fi supraîncărcat.
- d) În cazul în care clasa nu are un constructor definit de programator, va fi furnizat unul implicit.
- e) Constructorii pot fi metode virtuale.

9. Examinați următorul cod și stabiliți ce fel de funcție este `set()`.

```
class numar {  
    int I;  
    public:  
        void set(int v) {I = v;}  
        int read() {return I;}  
};
```

- a) Funcție locală
- b) Funcție de tip void
- c) Funcție nulă
- d) Funcție de tip class
- e) Funcție membru

10. Examinați următorul cod și stabiliți ce va fi afișat în rezultatul executării lui.

```
class X {
    private:
        int a;
    public:
        X(){cout<<"A fost apelat constructorul X "<<endl;}
        ~X(){cout<<" A fost apelat destructorul X "<<endl;}
};
main() {
    X x;
    return 0 ;
}
```

- a) A fost apelat destructorul ~X()
- b) A fost apelat constructorul X
- c) A fost apelat constructorul X()
- d) A fost apelat destructorul X
- e) A fost apelat constructorul X

11. Examinați următorul cod și stabiliți de ce constructorul `numar` poate avea două definiții diferite.

```
class numar{
    int I;
    public:
        numar() {I=0;}
        numar(int j) {I=j;}
};
```

- a) Este posibil din cauza supraîncărcării destructorului.
- b) Este posibil datorită funcției supraîncărcării.
- c) Este posibil datorită moștenirii de clasă.
- d) Nu este posibil.
- e) Este posibil din cauza supraîncărcării obiectului.

12. Examinați următorul cod și stabiliți ce va fi afișat în rezultatul executării lui.

```
class X {
    private:
        int a;
    public:
        X(){cout<<" A fost apelat constructorul X "<<endl;}
        X(int n){cout<<"A fost apelat constructorul X(int)"<<endl;}
        ~X(){cout<<" A fost apelat destructorul X "<<endl;}
};
```

```
};
main() {
    X x(3);
}
```

- a) A fost apelat constructorul X
A fost apelat constructorul X(int)
A fost apelat destructorul X
- b) A fost apelat constructorul X
A fost apelat destructorul X
- c) A fost apelat destructorul X(int)
A fost apelat destructorul X
- d) A fost apelat constructorul X(int)
A fost apelat destructorul X
- e) A fost apelat constructorul X
A fost apelat constructorul X(int)

13. Examinați atent următorul cod și selectați din următoarele înscrisuri pe cele care sunt echivalente cu: Cuvint Search ("Ana").

```
class Cuvint {
public:
    Cuvint(const char*, int=0);
private:
    int var;
    char *string;
};
```

- a) Cuvint Search{}="Ana"
- b) Cuvint.Search("Ana")
- c) Cuvint Search("Ana",0)
- d) Cuvint->Search("Ana")
- e) Cuvint Search(0,"Ana")

Barem de notare (Testul 1)

Item	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Punctaj maxim	5	5	5	5	10	10	10	10	10	10	10	10	10	10

Itemul 1: se acordă 5 puncte pentru răspunsul corect și 1 punct penalizare pentru încă un alt răspuns greșit. În cazul răspunsului greșit se vor acorda 0 puncte.

Itemul 2: se acordă 5 puncte pentru răspunsul corect și 1 punct penalizare pentru încă un alt răspuns greșit. În cazul răspunsului greșit se vor acorda 0 puncte.

Itemul 3: se acordă 5 puncte pentru răspunsul corect și 1 punct penalizare pentru încă un alt răspuns greșit. În cazul răspunsului greșit se vor acorda 0 puncte.

Itemul 4: se acordă 5 puncte pentru răspunsul corect și 1 punct penalizare pentru încă un alt răspuns greșit. În cazul răspunsului greșit se vor acorda 0 puncte.

- Itemul 5:** se acordă 10 puncte pentru răspunsul corect și 2 puncte penalizare pentru încă un alt răspuns greșit. În cazul răspunsului greșit se vor acorda 0 puncte.
- Itemul 6:** se acordă 10 puncte pentru răspunsul corect și 2 puncte penalizare pentru încă un alt răspuns greșit. În cazul răspunsului greșit se vor acorda 0 puncte.
- Itemul 7:** se acordă 10 puncte pentru răspunsul corect și 2,5 puncte penalizare pentru încă un alt răspuns greșit. În cazul răspunsului greșit se vor acorda 0 puncte.
- Itemul 8:** se acordă 10 puncte pentru răspunsul corect și 2 puncte penalizare pentru încă un alt răspuns greșit. În cazul răspunsului greșit se vor acorda 0 puncte.
- Itemul 9:** se acordă 10 puncte pentru răspunsul corect și 2 puncte penalizare pentru încă un alt răspuns greșit. În cazul răspunsului greșit se vor acorda 0 puncte.
- Itemul 10:** se acordă 10 puncte pentru răspunsul corect și 2 puncte penalizare pentru încă un alt răspuns greșit. În cazul răspunsului greșit se vor acorda 0 puncte.
- Itemul 11:** se acordă 10 puncte pentru răspunsul corect și 2 puncte penalizare pentru încă un alt răspuns greșit. În cazul răspunsului greșit se vor acorda 0 puncte.
- Itemul 12:** se acordă 10 puncte pentru răspunsul corect și 2 puncte penalizare pentru încă un alt răspuns greșit. În cazul răspunsului greșit se vor acorda 0 puncte.
- Itemul 13:** se acordă 10 puncte pentru răspunsul corect și 2 puncte penalizare pentru încă un alt răspuns greșit. În cazul răspunsului greșit se vor acorda 0 puncte.
- Itemul 14:** se acordă 10 puncte pentru răspunsul corect și 2 puncte penalizare pentru încă un alt răspuns greșit. În cazul răspunsului greșit se vor acorda 0 puncte.

Notă	5	6	7	8	9	10
Punctaj	50-60	61-70	71-80	81-100	100-115	115-120

Programarea Orientată pe Obiecte

Testul 2

1. Care dintre următoarele declarații arată că *d* este o subclasă a clasei *b*?
 - a) `class d : public b {};`
 - b) `class b : public d {};`
 - c) `class b public b {};`
 - d) `class b, public d {};`
 - e) `class d, public b {};`
2. Selectați propoziția adevărată.
 - a) Moștenirea permite reutilizarea codului.
 - b) Moștenirea fără polimorfism este singura modalitate de a scrie un cod bun.
 - c) Polimorfismul nu poate fi pus în aplicare în C++.
 - d) Moștenirea fără polimorfism este imposibilă.
 - e) Moștenirea are prioritate față de polimorfism.
3. Selectați propozițiile adevărate.
 - a) Funcția redeclarată cu o altă serie de argumente nu va mai putea fi moștenită mai departe, ca fiind o metodă virtuală.
 - b) Constructorii pot fi declarați virtuali.
 - c) Metodele statice ale unei clase pot fi declarate ca fiind virtuale.

- d) Funcțiile nemembru nu pot fi declarate ca fiind virtuale.
- e) Destructorii pot fi declarați virtuali.

4. Selectați afirmațiile adevărate.

- a) Cuvântul-cheie “protected” nu poate fi folosit ca modificador de acces în cadrul unei relații de moștenire.
- b) Funcțiile membru ale clasei derivate au acces la membrii privați ai clasei de bază.
- c) Orice funcție sau clasă *friend* a unei clase derivate nu are aceleași drepturi și posibilități de a accesa membrii clasei de bază ca oricare alt membru al clasei derivate.
- d) Când se declară un obiect dintr-o clasă derivată, compilatorul execută întâi constructorul clasei de bază, apoi constructorul clasei derivate.

5. Examinați următorul cod și alegeți propoziția adevărată.

```
class A {
public:
    A();
    ~A();
private:
    int    n1;
    float n2;
};
class B : public A { };
```

- a) Codul este corect.
- b) În clasa B trebuie să se definească în mod explicit un constructor.
- c) În clasa A este necesar de a inițializa n1 și n2.
- d) Clasa A trebuie să furnizeze un constructor de copiere, pentru ca aceasta să fie utilizată ca clasă-bază.

6. Selectați afirmația adevărată.

- a) Obiectele unei clase derivate au acces la membrii privați ai clasei lor de bază.
- b) Relația de moștenire este tranzitivă.
- c) Funcțiile „friend” ale clasei de bază se moștenesc de către clasa derivată.
- d) Constructorul și destructorul clasei de bază se moștenesc în clasa derivată.

7. Examinați următorul cod și stabiliți cum se poate accesa membrul **int a** din clasa de bază.

```
class baza {
protected:
    int a,b;
public:
    void setab(int n, int m) (a=n;b=m;)
};
class derivata: protected baza {
int c;
public :
    void setc(int n) {c=n;}
};
```

- a) Prin utilizarea doar a funcțiilor membru ale clasei derivate.
 - b) Prin folosirea funcțiilor membru ale clasei derivate și clasei de bază.
 - c) Prin utilizarea doar a funcțiilor membru ale clasei de bază.
 - d) Numai cu ajutorul funcțiilor prietene.
 - e) Prin utilizarea oricărei funcții.
8. Știind că o clasă de bază este derivată și are specificatorul de acces public, selectați afirmația adevărată.
- a) Toți membrii din clasa derivată devin membri privați ai clasei derivate.
 - b) Toți membrii clasei derivate devin membri publici ai clasei de bază.
 - c) Toți membrii clasei de baza devin membri protejați ai clasei derivate.
 - d) Toți membrii protected ai clasei de baza sunt protected și în clasa derivată.
 - e) Toți membrii clasei de baza devin membri publici ai clasei derivate.
9. Examinați următorul cod și alegeți numărul ce indică corect de câte ori este utilizat constructorul bazei.

```
class Baza { };
    class Sub1 : public virtual Baza { };
    class Sub2 : public Baza { };
    class Multi : public Sub1, public Sub2 { };
int main() {
    Multi m;
}
```

- a) 0
 - b) 1
 - c) 2
 - d) 3
 - e) 4
10. Examinați următorul cod și stabiliți care este nivelul de acces pentru funcția membru `setX()` în clasa `D2`.

```
class Baza {
public:
    void setX(int a) { x = a; }
private:
    int x;
};
class D1 : protected Baza { };
class D2 : public D1 { };
```

- a) global
- b) virtual
- c) private
- d) protected
- e) public

11. Examinați următorul cod și alegeți propoziția adevărată:

```
class Person {
public:
    Person();
    ~Person();
protected:
    int x;
};
class AgentEconomic : protected Person {
public:
    AgentEconomic ();
    ~ AgentEconomic ();
    void Person(int a) { this->n = a; }
private:
    int n;
};
main() {
    AgentEconomic agent;
}
```

- a) Codul va eșua atunci când se va încerca de a crea o instanță a clasei *AgentEconomic* și se va încerca accesarea constructorului din clasa *Persoana*.
- b) Nimic.
- c) Nu poate fi folosit specificatorul de acces *protected* când constructorul este public (sau privat), în clasa de bază.
- d) Nu este corect să se definească o funcție numită *Persoana* în clasa *AgentEconomic*.

Selectați condiția în care o clasă necesită un destructor virtual.

- a. Clasa întotdeauna are nevoie de un destructor virtual.
- b. Este nevoie de un destructor virtual, dacă în clasă sunt funcții virtuale pure.
- c. Este nevoie de un destructor virtual, dacă clasele derivate au destructori lor.
- d. Destructorii virtuali nu sunt permisi în C / C ++.
- e. Nu aveți nevoie de destructor virtual în cazul în care există un constructor virtual.

12. Examinați următorul cod. Stabiliți de ce cuvântul-cheie "virtual" a fost adăugat înainte de destructorul *Person*.

```
class Persoana {
public:
    Persoana();
    virtual ~Persoana();
};
class Student : public Persoana {
public:
    Student();
    ~Student();
};
```

- a) Pentru a împiedica instanțierea directă a clasei *Persoana*.
- b) Pentru a face imposibilă supraîncărcarea acestui destructor.

- c) Pentru a face posibil ca alte clase să poată moșteni clasa *Persoana*.
- d) Pentru a îmbunătăți viteza de distrugere a clasei *Persoana*.
- e) Pentru a se asigura că va fi apelat destructorul corespunzător al clasei derivate.

13. Examinați următorul cod și scrieți care declarații din funcția `main()` sunt corecte. Justificați alegerea.

```
#include <iostream.h>
class B1 {
    public:
        int x;
};
class B2 {
    int y;
};
class B3 {
    public:
        int z;
};
class B4 {
    public:
        int t;
};
class D: private B1, protected B2, public B3, B4{ int u; };
main(){
    D d;
    cout<<d.u;
    cout<<d.x;
    cout<<d.y;
    cout<<d.z;
    cout<<d.t;
}
```

14. Examinați următorul cod. Scrieți ce va fi afișat în cazul în care programul este corect. Dacă programul este incorect, găsiți greșeala. Justificați.

```
#include <iostream.h>
class cls1{
    protected:
        int x;
    public:
        cls1(int i=10) { x=i; }
        int get_x() { return x; }
};
class cls2: private cls1{
    public:
        cls2(int i):cls1(i) {}
};
main() {
    cls2 d(37);
    cout<<d.get_x();
}
```

Barem de notare (Testul 2)

Item	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
Punctaj maxim	5	5	5	5	10	10	10	10	10	10	10	10	10	10	10

Itemul 1 – Itemul 3: se acordă câte 5 puncte pentru fiecare item în cazul în care răspunsul este corect și 1 punct penalizare pentru încă un alt răspuns greșit. În cazul răspunsului greșit se vor acorda 0 puncte.

Itemul 4: se acordă 5 puncte pentru răspunsul corect și 2,5 puncte penalizare pentru încă un alt răspuns greșit. În cazul răspunsului greșit se vor acorda 0 puncte.

Itemul 5: se acordă 10 puncte pentru răspunsul corect și 2,5 puncte penalizare pentru încă un alt răspuns greșit. În cazul răspunsului greșit se vor acorda 0 puncte.

Itemul 6: se acordă 10 puncte pentru răspunsul corect și 2,5 puncte penalizare pentru încă un alt răspuns greșit. În cazul răspunsului greșit se vor acorda 0 puncte.

Itemul 7 – Itemul 10 se acordă 10 puncte pentru fiecare item în cazul în care răspunsul este corect și 2 puncte penalizare pentru încă un alt răspuns greșit. În cazul răspunsului greșit se vor acorda 0 puncte.

Itemul 11: se acordă 10 puncte pentru răspunsul corect și 2,5 puncte penalizare pentru încă un alt răspuns greșit. În cazul răspunsului greșit se vor acorda 0 puncte.

Itemul 12 – Itemul 14 se acordă 10 puncte pentru răspunsul corect și 2 puncte penalizare pentru încă un alt răspuns greșit. În cazul răspunsului greșit se vor acorda 0 puncte.

Itemul 15: se acordă 10 puncte pentru răspunsul corect.

Notă	5	6	7	8	9	10
Punctaj	50-60	61-70	71-80	81-100	101-115	116-130

Anexa 4. Teste de evaluare finală Teste de evaluare finală

UNIVERSITATEA DE STAT DIN TIRASPOL
FACULTATEA FIZICĂ, MATEMATICĂ ȘI TEHNOLOGII INFORMAȚIONALE
CATEDRA INFORMATICĂ ȘI TEHNOLOGII INFORMAȚIONALE

Test final

Disciplina: POO (C++)

Aprobat, șef catedră ITI, dr. hab., prof. univ. L. Chiriac _____

Nume _____ Prenume _____ Grupa _____ Data _____

Testul nr. 1

1. Alegeți secvența care va fi afișată în urma executării următoarelor instrucțiuni:

```
#include<iostream.h>
int main()
{ int i,j;
  for(i=1; i<=4; i++){
    for(j=1; j<=i; j++)
      cout<<j;}
```

- a. 1 2 2 3 3 3 4 4 4 4
- b. 1 1 2 1 2 3 1 2 3 4
- c. 1 1 1 1 2 2 2 2 3 3 3 3 4 4 4 4
- d. 1 2 3 4 1 2 3 4 1 2 3 4 1 2 3 4

2. Explicați ce înseamnă „abstractizarea datelor” și „încapsularea (ascunderea) informației”.

Abstractizarea datelor este

Încapsularea informației este

3. Într-o clasă, la definirea metodelor sale, întotdeauna există acces la toate atributele clasei.

☐ Da

☐ Nu

4. Clasa care conține una sau mai multe funcții virtuale pure se numește _____.

5. Completați spațiile libere cu informația corectă.

a. Principiul prin care putem reutiliza și extinde clasele existente este _____.

b. Metoda apelată la instanțierea unei clase este _____.

c. Conceptul de bază în POO ce reunește o colecție de obiecte care partajează aceeași listă de atribute informaționale și comportamentale. _____

6. Alegeți afirmațiile adevărate dintre următoarele propoziții:
- Metodele inline sunt funcții scurte care nu sunt apelate.
 - La compilare, apelul funcției inline este înlocuit de codul ei, similar funcțiilor macro.
 - Metodele inline permit execuția rapidă a codului prin evitarea efortului necesar unui apel de funcție.
 - Metodele inline nu contribuie la creșterea dimensiunii codului executabil.
7. Care dintre funcțiile de mai jos pot fi declarate inline?
- ```
int f(int i);
 if (i>0)
 while (i){i--;cout<<d<<endl;}
```
  - ```
int f(int x){return x++;}
```
 - ```
void f(int x){do puts(" x");
```
  - ```
    a    b    c
        while(--x);}
```
8. Găsiți eroarea și explicați cum poate fi ea corectată.
- Fie definită clasa *Angajat*. Ea conține următorul prototip:
- ```
int angajat (const char *, const char *) ;
```
9. Determinați dacă programul de mai jos este corect. În caz de răspuns afirmativ, spuneți ce afișează el; în caz de răspuns negativ, explicați de ce nu este corect.

```
#include<iostream.h>
class cls1{
public:
 int a;
 cls1() { a=7; }
};
class cls2{
public:
 int b;
 cls2(int i) { b=i; }
 cls2(cls1& x) { b=x.a; }
};
main(){
 cls1 x;
 cout<<x.a;
 cls2 y(x);
 cout<<y.b;
}
```

10. Scrieți un program care:
- implementează o ierarhie de clase cu clasa de bază *Persoana* și subclasele *Inginer* și *Muncitor*;

- clasa *Persoana* va conține attributele: nume, tarif\_oră și nr\_ore lucrate și ca metodă calcul\_salariu;
- clasele *Inginer* și *Muncitor* vor redefini metoda „calcul salariu”, știind că un inginer primește dublul tarifului unei persoane, iar muncitorul primește jumătate din acest tarif;
- în funcția main() se vor crea câte un obiect de tip *Inginer*, respectiv *Muncitor*; se vor afișa numele și salariul corespunzător fiecăruia.

#### Barem de notare

| Item                 | 1 | 2 | 3 | 4 | 5  | 6  | 7  | 8  | 9  | 10 |
|----------------------|---|---|---|---|----|----|----|----|----|----|
| <b>Punctaj maxim</b> | 5 | 5 | 5 | 5 | 10 | 10 | 10 | 14 | 16 | 20 |

| Notă           | 5     | 6     | 7     | 8     | 9     | 10     |
|----------------|-------|-------|-------|-------|-------|--------|
| <b>Punctaj</b> | 30-45 | 46-55 | 56-70 | 71-80 | 81-95 | 96-100 |

## Anexa 5. Teme pentru cercetare suplimentară

### Temele propuse pentru cercetare suplimentară la cursul POO

- ☐ 1. Principiile programării orientate pe obiecte **(plin)**
- ☐ 2. Limbajul de programare C++ **(plin)**
- ☐ 3. Elemente de POO: obiecte, clasă, atribute, metode **(plin)**
- ☐ 4. Declararea claselor și obiectelor în C++ **(plin)**
- ☐ 5. Utilizarea obiectelor în C++ **(plin)**
- ☐ 6. Constructori, destructori **(plin)**
- ☐ 7. Constructorul de copiere
- ☐ 8. Funcții inline **(plin)**
- ☐ 9. Funcții-prietene și clase-prietene
- ☐ 10. Redefinirea operatorilor
- ☐ 11. Drepturile de acces la membri
- ☐ 12. Moștenirea – ierarhii de clase în C++ **(plin)**
- ☐ 13. Clase virtuale **(plin)**
- ☐ 14. Tipuri abstracte de date și obiecte **(plin)**
- ☐ 15. Moștenirea multiplă **(plin)**
- ☐ 16. Membrii statici ai unei clase
- ☐ 17. Polimorfismul **(plin)**

**Anexa 6. Indicatorii statistici calculați cu aplicația SPSS pentru eșantionul de control, anul de studii 2015-2016, secția cu frecvență la zi**

[DataSet1] C:\Users \Desktop\ Capitolul 3 \esantion\_Control\_2015\_2016.sav

DATASET NAME DataSet1 WINDOW=FRONT.  
 FREQUENCIES VARIABLES=eval\_1 eval\_2 Nota\_fin  
 /STATISTICS=STDDEV VARIANCE MEAN MEDIAN MODE SKEWNESS SESKEW  
 KURTOSIS SEKURT  
 /ORDER=ANALYSIS.

|                        |         | Statistics        |         |                   |
|------------------------|---------|-------------------|---------|-------------------|
|                        |         | Test_1            | Test_2  | Nota_finala       |
| N                      | Valid   | 17                | 17      | 17                |
|                        | Missing | 0                 | 0       | 0                 |
| Mean                   |         | 7,2353            | 7,2941  | 6,7059            |
| Median                 |         | 7,0000            | 8,0000  | 7,0000            |
| Mode                   |         | 6,00 <sup>a</sup> | 8,00    | 7,00 <sup>a</sup> |
| Std. Deviation         |         | 1,03256           | 1,15999 | 1,10480           |
| Variance               |         | 1,066             | 1,346   | 1,221             |
| Skewness               |         | ,240              | -,383   | -,280             |
| Std. Error of Skewness |         | ,550              | ,550    | ,550              |
| Kurtosis               |         | -1,032            | -,759   | -1,184            |
| Std. Error of Kurtosis |         | 1,063             | 1,063   | 1,063             |
| Maximum                |         | 9,00              | 9,00    | 8,00              |

a. Multiple modes exist. The smallest value is shown



**Anexa 7. Indicatorii statistici calculați cu aplicația SPSS pentru eșantionul de control, anul de studii 2016-2017, secția cu frecvență la zi**

[DataSet3] C:\Users\Ala\Desktop\Capitolul 3\Statistica\_Ala Gasnas\esantion\_Control\_2016\_2017.sav

DATASET NAME DataSet2 WINDOW=FRONT.  
 FREQUENCIES VARIABLES=eval\_1 eval\_2 t\_final  
 /STATISTICS=STDDEV VARIANCE MEAN MEDIAN MODE SKEWNESS SESKEW  
 KURTOSIS SEKURT  
 /ORDER=ANALYSIS.

|                        |         | Statistics |         |            |
|------------------------|---------|------------|---------|------------|
|                        |         | Test 1     | Test 2  | Test final |
| N                      | Valid   | 17         | 17      | 17         |
|                        | Missing | 0          | 0       | 0          |
| Mean                   |         | 6,5294     | 6,1176  | 6,2235     |
| Median                 |         | 7,0000     | 6,0000  | 6,2000     |
| Mode                   |         | 7,00       | 5,00    | 5,60       |
| Std. Deviation         |         | 1,17886    | 1,05370 | ,73103     |
| Variance               |         | 1,390      | 1,110   | ,534       |
| Skewness               |         | -,209      | ,466    | ,268       |
| Std. Error of Skewness |         | ,550       | ,550    | ,550       |
| Kurtosis               |         | -1,464     | -,931   | -,865      |
| Std. Error of Kurtosis |         | 1,063      | 1,063   | 1,063      |
| Maximum                |         | 8,00       | 8,00    | 7,60       |

a. Multiple modes exist. The smallest value is shown

**Anexa 8. Indicatorii statistici calculați cu aplicația SPSS pentru eșantionul de control, anul de studii 2016-2017, secția cu frecvență redusă**

[DataSet4] C:\Users\Ala\Desktop\ Capitolul 3\Statistica\_Ala Gasnas\esantion\_Control\_2016\_2017\_ff.sav

DATASET NAME DataSet2 WINDOW=FRONT.  
 FREQUENCIES VARIABLES=eval\_1 eval\_2 t\_final  
 /STATISTICS=STDDEV VARIANCE MEAN MEDIAN MODE SKEWNESS SESKEW  
 KURTOSIS SEKURT  
 /ORDER=ANALYSIS.

|                        |         | Statistics |         |            |
|------------------------|---------|------------|---------|------------|
|                        |         | Test 1     | Test 2  | Test final |
| N                      | Valid   | 11         | 11      | 11         |
|                        | Missing | 0          | 0       | 0          |
| Mean                   |         | 6,0909     | 6,0000  | 6,1818     |
| Median                 |         | 6,0000     | 5,0000  | 6,0000     |
| Mode                   |         | 6,00       | 5,00    | 6,00       |
| Std. Deviation         |         | ,94388     | 1,26491 | ,75076     |
| Variance               |         | ,891       | 1,600   | ,564       |
| Skewness               |         | ,663       | ,725    | ,320       |
| Std. Error of Skewness |         | ,661       | ,661    | ,661       |
| Kurtosis               |         | ,199       | -1,302  | -,523      |
| Std. Error of Kurtosis |         | 1,279      | 1,279   | 1,279      |
| Maximum                |         | 8,00       | 8,00    | 7,50       |

**Anexa 9. Indicatorii statistici calculați cu aplicația SPSS pentru eșantionul experimental, anul de studii 2015-2016, secția cu frecvență la zi**

[DataSet5] C:\Users\Ala\Desktop\ Capitolul 3\Statistica\_Ala  
Gasnas\esantion\_exp\_2015\_2016.sav

DATASET NAME DataSet1 WINDOW=FRONT.  
FREQUENCIES VARIABLES=eval\_1 eval\_2 Nota\_fin  
/STATISTICS=STDDEV VARIANCE MEAN MEDIAN MODE SKEWNESS SESKEW  
KURTOSIS SEKURT  
/ORDER=ANALYSIS.

|                        |         | Statistics |        |             |
|------------------------|---------|------------|--------|-------------|
|                        |         | Test 1     | Test 2 | Nota finala |
| N                      | Valid   | 14         | 14     | 14          |
|                        | Missing | 0          | 0      | 0           |
| Mean                   |         | 7,0714     | 7,4286 | 7,6429      |
| Median                 |         | 7,0000     | 7,5000 | 8,0000      |
| Mode                   |         | 7,00       | 8,00   | 8,00        |
| Std. Deviation         |         | ,99725     | ,85163 | ,92878      |
| Variance               |         | ,995       | ,725   | ,863        |
| Skewness               |         | -,161      | -,178  | -,487       |
| Std. Error of Skewness |         | ,597       | ,597   | ,597        |
| Kurtosis               |         | ,751       | -,296  | -,226       |
| Std. Error of Kurtosis |         | 1,154      | 1,154  | 1,154       |
| Maximum                |         | 9,00       | 9,00   | 9,00        |

**Anexa 10. Indicatorii statistici calculați cu aplicația SPSS pentru eșantionul experimental, anul de studii 2016-2017, secția cu frecvență la zi**

GET

FILE='G:\Statistica\_Ala Gasnas\esantion\_exp\_2016\_2017.sav'.

DATASET NAME DataSet2 WINDOW=FRONT.

FREQUENCIES VARIABLES=eval\_1 eval\_2 t\_final

/STATISTICS=STDDEV VARIANCE MEAN MEDIAN MODE SKEWNESS SESKEW  
KURTOSIS SEKURT

/ORDER=ANALYSIS.

| Statistics             |         |        |        |                   |
|------------------------|---------|--------|--------|-------------------|
|                        |         | Test 1 | Test 2 | Test final        |
| N                      | Valid   | 13     | 13     | 13                |
|                        | Missing | 0      | 0      | 0                 |
| Mean                   |         | 8,1538 | 8,6923 | 8,8769            |
| Median                 |         | 8,0000 | 9,0000 | 9,0000            |
| Mode                   |         | 8,00   | 8,00   | 8,00 <sup>a</sup> |
| Std. Deviation         |         | ,68874 | ,75107 | ,72819            |
| Variance               |         | ,474   | ,564   | ,530              |
| Skewness               |         | -,203  | ,611   | ,234              |
| Std. Error of Skewness |         | ,616   | ,616   | ,616              |
| Kurtosis               |         | -,496  | -,776  | -1,311            |
| Std. Error of Kurtosis |         | 1,191  | 1,191  | 1,191             |
| Maximum                |         | 9,00   | 10,00  | 10,00             |

a. Multiple modes exist. The smallest value is shown

**Anexa 11. Indicatorii statistici calculați cu aplicația SPSS pentru eșantionul experimental, anul de studii 2016-2017, secția cu frecvență redusă**

```
GET
FILE='G:\Statistica_Ala Gasnas\esantion_exp_2016_2017_ff.sav'
DATASET NAME DataSet2 WINDOW=FRONT.
FREQUENCIES VARIABLES=eval_1 eval_2 t_final
 /STATISTICS=STDDEV VARIANCE MEAN MEDIAN MODE SKEWNESS SESKEW
 KURTOSIS SEKURT
 /ORDER=ANALYSIS.
```

|                        |         | Statistics |         |            |
|------------------------|---------|------------|---------|------------|
|                        |         | Test_1     | Test_2  | Test_final |
| N                      | Valid   | 12         | 12      | 12         |
|                        | Missing | 0          | 0       | 0          |
| Mean                   |         | 7,1667     | 7,7500  | 7,9167     |
| Median                 |         | 7,0000     | 8,0000  | 8,0000     |
| Mode                   |         | 6,00       | 8,00    | 8,00       |
| Std. Deviation         |         | 1,26730    | 1,48477 | 1,25831    |
| Variance               |         | 1,606      | 2,205   | 1,583      |
| Skewness               |         | ,596       | -,087   | ,430       |
| Std. Error of Skewness |         | ,637       | ,637    | ,637       |
| Kurtosis               |         | -1,367     | -,103   | -,387      |
| Std. Error of Kurtosis |         | 1,232      | 1,232   | 1,232      |
| Maximum                |         | 9,00       | 10,00   | 10,00      |

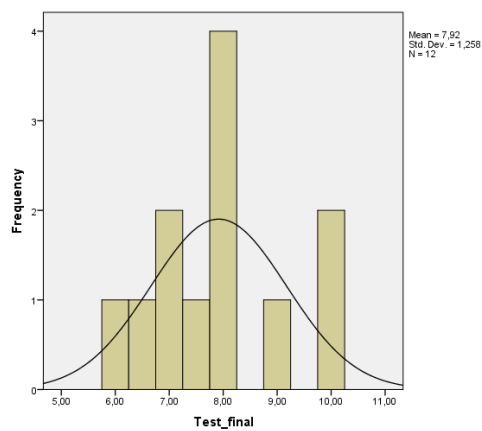
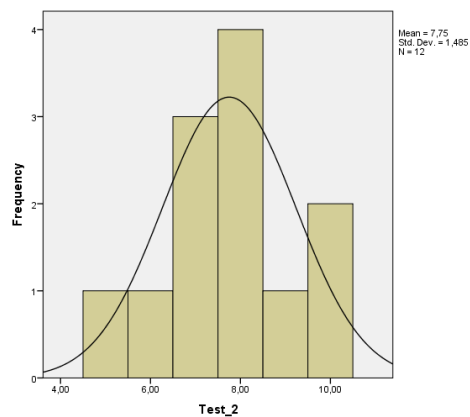
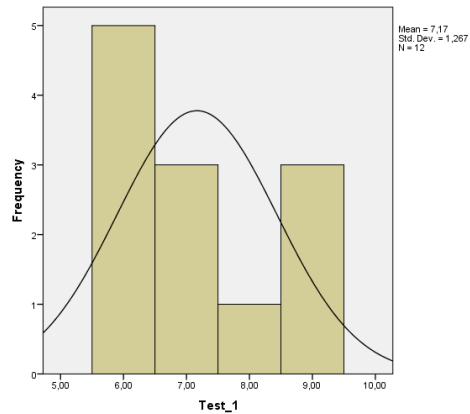
## Anexa 12. Histogramele eșantionului experimental 2016-2017, grupele cu frecvență redusă

GET

FILE='C:\Users\Ala\Desktop\ Capitolul 3\Statistica\_Ala  
Gasnas\esantion\_exp\_2016\_2017\_ff.sav'.  
DATASET NAME DataSet7 WINDOW=FRONT.

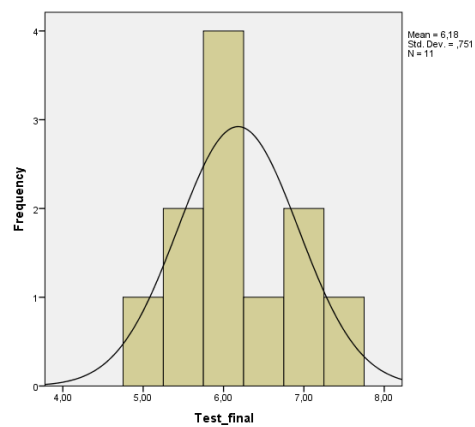
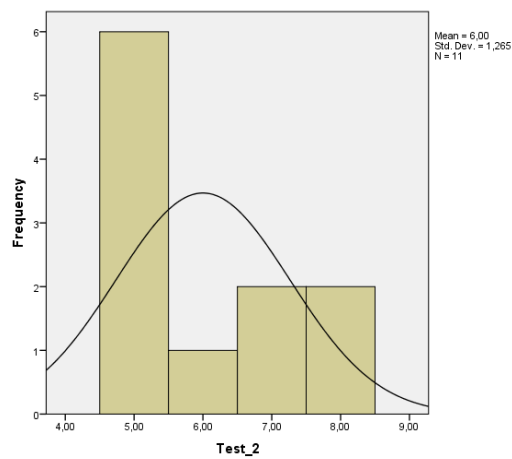
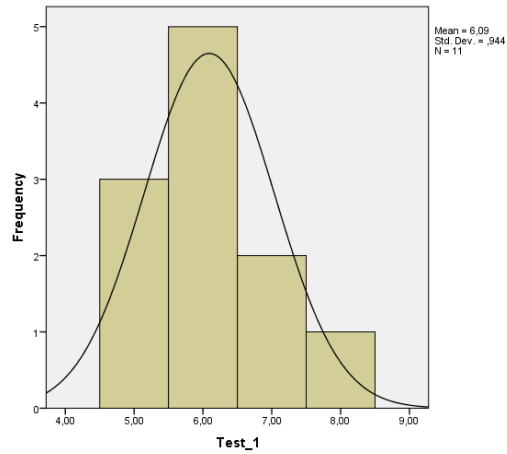
GRAPH

/HISTOGRAM(NORMAL)=eval\_1.



### Anexa 13. Histogramele eșantionului de control 2016-2017, grupele cu frecvență redusă

[DataSet4] C:\Users\Ala\Desktop\ Capitolul 3\Statistica\_Ala Gasnas\esantion\_Control\_2016\_2017\_ff.sav



**Anexa 14. Rezultatele experimentului pedagogic, anul de studii 2015-2016,  
secția cu frecvență la zi**

| <b>Id_pers<br/>(EE)</b> | <b>Initial</b> | <b>eval_1</b> | <b>eval_2</b> | <b>t_final</b> |
|-------------------------|----------------|---------------|---------------|----------------|
| B-1                     | 5,40           | 7,00          | 8,00          | 7,00           |
| B-2                     | 5,90           | 6,00          | 6,00          | 5,00           |
| B-3                     | 7,00           | 8,00          | 8,00          | 8,00           |
| B-4                     | 6,80           | 8,00          | 8,00          | 7,00           |
| B-5                     | 6,60           | 7,00          | 7,00          | 6,00           |
| B-6                     | 5,60           | 7,00          | 6,00          | 6,00           |
| B-7                     | 5,30           | 8,00          | 7,00          | 6,00           |
| B-8                     | 5,40           | 6,00          | 6,00          | 6,00           |
| B-9                     | 8,50           | 8,00          | 8,00          | 8,00           |
| B-10                    | 5,20           | 8,00          | 8,00          | 8,00           |
| B-11                    | 5,40           | 7,00          | 8,00          | 7,00           |
| B-12                    | 8,20           | 9,00          | 9,00          | 8,00           |
| B-13                    | 5,40           | 6,00          | 7,00          | 7,00           |
| B-14                    | 5,40           | 6,00          | 5,00          | 5,00           |
| B-15                    | 8,30           | 9,00          | 9,00          | 8,00           |
| B-16                    | 5,50           | 7,00          | 8,00          | 7,00           |
| B-17                    | 6,00           | 6,00          | 6,00          | 5,00           |
| A-1                     | 5,90           | 8,00          | 8,00          | 8,00           |
| A-2                     | 5,90           | 8,00          | 8,00          | 8,00           |
| A-3                     | 8,00           | 7,00          | 8,00          | 8,00           |
| A-4                     | 4,90           | 6,00          | 7,00          | 7,00           |
| A-5                     | 5,20           | 7,00          | 7,00          | 8,00           |
| A-6                     | 5,40           | 7,00          | 6,00          | 6,00           |
| A-7                     | 6,80           | 7,00          | 6,00          | 7,00           |
| A-8                     | 6,50           | 9,00          | 9,00          | 9,00           |
| A-9                     | 5,30           | 7,00          | 7,00          | 6,00           |
| A-10                    | 9,10           | 7,00          | 8,00          | 8,00           |
| A-11                    | 5,40           | 6,00          | 7,00          | 8,00           |
| A-12                    | 5,90           | 8,00          | 8,00          | 9,00           |
| A-13                    | 8,90           | 7,00          | 8,00          | 8,00           |
| A-14                    | 5,00           | 5,00          | 7,00          | 7,00           |



**Anexa 15. Rezultatele experimentului pedagogic, anul de studii 2016-2017,  
secția cu frecvență la zi**

| <b>Id_pers<br/>(EE)</b> | <b>Initial</b> | <b>eval_1</b> | <b>eval_2</b> | <b>t_final</b> |
|-------------------------|----------------|---------------|---------------|----------------|
| D-1                     | 5,40           | 8,00          | 7,00          | 6,80           |
| D-2                     | 5,70           | 5,00          | 5,00          | 5,00           |
| D-3                     | 5,50           | 7,00          | 5,00          | 6,00           |
| D-4                     | 6,10           | 7,00          | 6,00          | 6,20           |
| D-5                     | 6,20           | 5,00          | 8,00          | 6,20           |
| D-6                     | 6,10           | 7,00          | 8,00          | 7,20           |
| D-7                     | 6,30           | 7,00          | 7,00          | 7,00           |
| D-8                     | 5,30           | 5,00          | 6,00          | 5,60           |
| D-9                     | 5,60           | 6,00          | 5,00          | 5,60           |
| D-10                    | 7,20           | 8,00          | 7,00          | 7,60           |
| D-11                    | 7,40           | 7,00          | 6,00          | 7,00           |
| D-12                    | 7,20           | 8,00          | 6,00          | 6,20           |
| D-13                    | 5,80           | 7,00          | 5,00          | 5,60           |
| D-14                    | 5,70           | 5,00          | 6,00          | 5,60           |
| D-15                    | 6,80           | 8,00          | 7,00          | 6,80           |
| D-16                    | 5,30           | 5,00          | 5,00          | 5,40           |
| D-17                    | 5,10           | 6,00          | 5,00          | 6,00           |
| C-1                     | 5,60           | 9,00          | 10,00         | 10,00          |
| C-2                     | 6,80           | 8,00          | 9,00          | 9,00           |
| C-3                     | 6,60           | 9,00          | 10,00         | 10,00          |
| C-4                     | 5,40           | 8,00          | 8,00          | 8,40           |
| C-5                     | 5,70           | 9,00          | 9,00          | 9,40           |
| C-6                     | 5,90           | 9,00          | 9,00          | 9,00           |
| C-7                     | 5,50           | 7,00          | 8,00          | 8,00           |
| C-8                     | 6,90           | 8,00          | 9,00          | 9,40           |
| C-9                     | 5,80           | 8,00          | 8,00          | 8,40           |
| C-10                    | 6,90           | 8,00          | 9,00          | 9,40           |
| C-11                    | 6,40           | 8,00          | 8,00          | 8,40           |
| C-12                    | 6,70           | 8,00          | 8,00          | 8,00           |
| C-13                    | 5,40           | 7,00          | 8,00          | 8,00           |
|                         |                |               |               |                |

**Anexa 16. Rezultatele experimentului pedagogic, anul de studii 2016-2017,  
secția cu frecvență redusă**

| Id_pers<br>(EE) | Initial | eval_1 | eval_2 | t_final |
|-----------------|---------|--------|--------|---------|
| F-1             | 6,30    | 8,00   | 5,00   | 6,00    |
| F-2             | 6,10    | 6,00   | 5,00   | 5,50    |
| F-3             | 6,80    | 7,00   | 5,00   | 6,50    |
| F-4             | 5,20    | 5,00   | 5,00   | 5,00    |
| F-5             | 6,50    | 6,00   | 7,00   | 6,00    |
| F-6             | 7,60    | 7,00   | 8,00   | 7,50    |
| F-7             | 6,30    | 5,00   | 5,00   | 5,50    |
| F-8             | 7,10    | 5,00   | 8,00   | 7,00    |
| F-9             | 7,50    | 6,00   | 7,00   | 7,00    |
| F-10            | 6,80    | 6,00   | 5,00   | 6,00    |
| F-11            | 6,40    | 6,00   | 6,00   | 6,00    |
| E-1             | 9,60    | 9,00   | 10,00  | 10,00   |
| E-2             | 7,10    | 8,00   | 8,00   | 8,00    |
| E-3             | 7,70    | 9,00   | 9,00   | 9,00    |
| E-4             | 4,80    | 6,00   | 7,00   | 7,50    |
| E-5             | 6,80    | 7,00   | 8,00   | 8,00    |
| E-6             | 6,80    | 7,00   | 8,00   | 8,00    |
| E-7             | 8,60    | 9,00   | 10,00  | 10,00   |
| E-8             | 7,10    | 6,00   | 5,00   | 6,00    |
| E-9             | 5,20    | 6,00   | 7,00   | 7,00    |
| E-10            | 7,50    | 6,00   | 6,00   | 6,50    |
| E-11            | 5,50    | 7,00   | 8,00   | 8,00    |
| E-12            | 5,60    | 6,00   | 7,00   | 7,00    |
|                 |         |        |        |         |
|                 |         |        |        |         |
|                 |         |        |        |         |

## **DECLARAȚIA PRIVIND ASUMAREA RĂSPUNDERII**

Subsemnata, declar pe răspundere personală că materialele prezentate în teza de doctor sunt rezultatul propriilor cercetări și realizări științifice. Conștientizez că, în caz contrar, urmează să suport consecințele în conformitate cu legislația în vigoare.

Ala Gasnaș

21.05.2018

## CURRICULUM VITAE

Numele: Gasnaș

Prenumele: Ala

Cetățenia: Republica Moldova



### **Educație și formare:**

2014–2017 – studii prin doctorat la specialitatea 532.02 –

*Didactica informaticii*, UST

1978–1983 – studii superioare, USM, Facultatea *Matematică și Cibernetică*

1967–1977 – studii medii, școala nr. 1, or. Chișinău.

### **Stagii:**

- Cursuri de perfecționare. *Metodologia utilizării TIC în învățământul superior*. 27.10.2010 – 17.01.2011. UST, Chișinău, Republica Moldova
- Cursuri de perfecționare. *Description of Qualifications in Higher Education in the Republic of Moldova* (În cadrul Proiectului *Development of Quality Assurance in Higher Education in Moldova - QUAEM*). 4–18 septembrie, 2015, Universitatea de Stat din Tiraspol

**Domenii de interes științific:** didactica informaticii; implementarea TIC în procesul didactic; programarea orientată pe obiecte; programare web; limbaje de asamblare

### **Participări la foruri științifice naționale și internaționale:**

- Conferința științifică internațională *The 20<sup>TH</sup> Conference on Applied and Industrial Mathematics: Dedicated to Academician Mitrofan M. Cioban*. Chișinău, 22–25 august, 2012;
- Conferința științifico-didactică națională cu participare internațională *Probleme actuale ale didacticii științelor reale*, consacrată aniversării a 80 de ani de la nașterea profesorului universitar Andrei Hariton. Chișinău, 4–6 octombrie, 2013;
- Conferința științifico-didactică națională cu participare internațională *Învățământul superior din Republica Moldova la 85 de ani*. Chișinău, 24–25 septembrie, 2015;
- Conferința științifică internațională *Mathematics & Information Technologies: Research and Education*, dedicată celor 70 de ani ai Universității de Stat din Moldova. Chișinău, 23–26 iunie, 2016;

- Conferința științifică internațională *The Fourth Conference of Mathematical Society of the Republic of Moldova*, dedicată centenarului lui Vladimir Andrunachevici. Chișinău, 28 iunie –2 iulie, 2017;
- Conferința științifică internațională *The Conference on Applied and Industrial Mathematics*: Iași, România, 15-16 septembrie 2017;
- Conferința științifico-didactică națională cu participare internațională ediția a II-a, *Probleme actuale ale didacticii științelor reale*, consacrată aniversării a 80-a a profesorului universitar Ilie Lupu. Chișinău, 11–12 mai, 2018.

**Lucrări științifice și științifico-metodice publicate:** 4 articole științifice în reviste naționale de categoriile B și C; un articol în curs de apariție. 7 comunicări la conferințe științifice.

***Cunoașterea limbilor:***

|                       | <b>Limba rusă</b> | <b>Limba engleză</b> | <b>Limba franceză</b> |
|-----------------------|-------------------|----------------------|-----------------------|
| Abilitatea de a citi  | Excelent          | Excelent             | Excelent              |
| Abilitatea de a scrie | Excelent          | Bine                 | Bine                  |
| Abilitatea de a vorbi | Excelent          | Bine                 | Bine                  |

***Date de contact:***

Adresa: or. Chișinău, str. Gh. Iablocikin 5

Telefon: 069107737; e-mail: [alagasnas@gmail.com](mailto:alagasnas@gmail.com)