

**MINISTERUL EDUCAȚIEI AL REPUBLICII MOLDOVA  
UNIVERSITATEA DE STAT „ALECU RUSSO” DIN BĂLȚI  
FACULTATEA DE ȘTIINȚE REALE, ECONOMICE ȘI ALE MEDIULUI  
CATEDRA DE MATEMATICĂ ȘI INFORMATICĂ**

# **Curriculum**

## **Programarea orientată pe obiect I**

**pentru specialitățile 141.02 „Informatica”, 141.01 „Matematica” și 141.02 „Informatica”,  
444.1 „Informatica”**

**Ciclul I, studii superioare de licență, învățământ cu frecvență redusă**

Titularul cursului: Eugeniu CABAC,  
dr., conf. univ.

**BĂLȚI, 2016**

Curriculum-ul a fost discutat la ședința Catedrei de Matematică și Informatică, proces-verbal nr. 1 din 29.08.2016.

Șeful Catedrei de Matematică și Informatică, dr., conf. univ. Eugeniu Plohotniuc

\_\_\_\_\_.

Curriculum-ul a fost aprobat la ședința Consiliului Facultății de Științe Reale, Economice și ale Mediului, proces-verbal nr. 5 din 20.10.2016.

Decanul Facultății de Științe Reale, Economice și ale Mediului, dr. hab., prof. univ. Pavel Topală \_\_\_\_\_.

### Informații de identificare a unității de curs

**Facultatea:** Științe Reale, Economice și ale Mediului

**Catedra:** Matematică și Informatică

**Domeniul general de studiu:** 14 Științe ale Educației, 44 Științe exacte

**Domeniul de formare profesională la ciclul I/II:** 141 Educație și formarea profesorilor, 444 Informatica

**Denumirea specializării:** Informatica (IP), Matematica și informatica (MI), Informatica (științe exacte) (IȘ)

**Denumirea unității de curs:** Programarea orientată pe obiect I

### Administrarea unității de curs

Plan licență	Codul unității de curs	Credite ECTS	Total ore	Repartizarea orelor				Forma de evaluare	Limba de predare
				Prel.	Sem.	Lab.	L. ind.		
IP	F.04.O.016	6	180			24	144	Examen	Rom/rus
MI	F.04.O.020	5	150	12	-	20	118		
IȘ	F.04.O.019	6	180			24	144		

**Statutul:** fundamental, obligatorie

### Informații referitoare la cadrul didactic

**Titularul cursului** - *Eugeniu Cabac*, doctor în științe pedagogice, magistrul în tehnologii informaționale. Absolvent al Universității de Stat „Alec Russo” din mun. Bălți, specialitatea „Matematica și informatica”. A efectuat stagii în diverse universități peste hotare, unde s-a specializat în domeniul informaticii, programării, utilizării tehnologiei informației și a comunicațiilor în învățământ, e-learning-ului.

**E-mail:** [eugeniu.cabac@gmail.com](mailto:eugeniu.cabac@gmail.com)

**Orele de consultații:** luni, 16:00 – 17:30. Consultațiile se oferă atât în regim „față-în-față”, cât și prin utilizarea poștei electronice, aplicației Skype sau aplicației „oovoo”. Numele în Skype – eugeniu.cabac.

### Integrarea unității de curs în programul de studii

Formarea programatorului, inclusiv, a profesorului de informatică, începe cu studierea cursurilor de bazele programării și limbaje structurate. Unitatea de curs, care integrează cunoștințele și abilitățile obținute la cursurile anterioare și pregătește studentul pentru utilizarea obiectelor în programare este „Programarea orientată pe obiect I”.

Cursul este orientat spre formarea principalelor competențe de proiectare și realizare a aplicațiilor în limbaje de nivel înalt cu utilizarea obiectelor.

### **Cunoștințe și competențe învățate/dobândite anterior, necesare pentru însușirea unității de curs**

- a. *Bazele programării*: tipuri de date, variabile, algoritmi, structuri ciclice, structuri alternative, funcții.
- b. *Programarea structurată*: tipuri de date complexe, structuri, particularitățile limbajului C++.
- c. *Aplicații generice*: utilizarea aplicațiilor standarde din sistemul de operare, interacțiunea cu dispozitivele calculatorului.

### **Competențele formate/dezvoltate în cadrul unității de curs**

În cadrul studierii unității de curs studenții își vor dezvolta următoarele competențe (CP – competențe profesionale; CT – competențe transversale):

**CP1.** Operarea cu fundamentele științifice ale matematicii, informaticii și ale științelor educației și utilizarea acestor noțiuni în comunicarea profesională.

**CP2.** Elaborarea modelelor pentru descrierea fenomenelor și proceselor reale.

**CP3.** Proiectarea, elaborarea și analiza algoritmilor pentru rezolvarea problemelor.

**CP4.** Programarea, dezvoltarea și mentenanța aplicațiilor informatice în limbaje de nivel înalt .

**CT1.** Aplicarea regulilor de muncă riguroasă și eficientă, manifestarea unei atitudini responsabile față de domeniul științific și didactic, pentru valorificarea optimă și creativă a propriului potențial în situații specifice, cu respectarea principiilor și a normelor de etică profesională.

**CT3.** Identificarea oportunităților de formare continuă și valorificarea eficientă a resurselor și tehnicilor de învățare pentru propria dezvoltare.

### **Finalitățile unității de curs**

La finalizarea studierii unității de curs și realizarea sarcinilor de învățare studentul va fi capabil:

- să identifice clasele și obiectele necesare pentru realizarea aplicației;
- să proiecteze structura aplicației;
- să interacționeze cu diverse dispozitive și aplicații în mod program;
- să elaboreze aplicații în limbaje de nivel înalt, utilizând bibliotecile de obiecte existente;
- să elaboreze obiecte și clase de obiecte proprii, în baza obiectelor existente sau de la zero.

## Conținutul unității de curs

### Prelegeri

Nr. d/o	Subiectele de studiu	Ore
		P
1.	<b>Noțiunea de clasă și obiect.</b> Noțiunea de clasă. Structura clasei. Variabilele și funcțiile clasei. Constructor și destructor. Obiect. Crearea obiectului. Crearea clasei de la zero. Moștenire. Clase derivate. Supraîncărcarea funcțiilor. Polimorfism. Implementarea metodelor în clasă. Pointerul this. Incapsulare.	2
2.	<b>Funcții și clase template.</b> Modalități de implementare a funcțiilor și claselor șablon. Interfețe. Structura bibliotecilor de obiecte. Ierarhia obiectelor.	2
3.	<b>Proiecte și module.</b> Structura proiectului. Parametrii proiectului. Module. Interacțiunea dintre module în cadrul proiectelor.	2
4.	<b>Obiecte standard.</b> Form. Noțiune de fereastră. Proprietățile și funcțiile de bază a ferestrei. Evenimente. Tipuri de ferestre. Noțiune de container de obiecte. Poziția absolută și poziția relativă. Alinierea obiectelor.	2
5.	<b>Interacțiunea cu sistemul de operare.</b> Funcții interne a sistemului de operare. Ferestre de dialog.	1
6.	<b>Lucrul cu liste. Lucrul cu valori dinamice. Stream.</b> ComboBox. ListBox. Poziționarea în liste. Flux binar. Convertarea fluxurilor. Operații de introducere și afișare utilizând fluxuri de date.	1
7.	<b>Lucrul cu biblioteci dinamice.</b> Structura bibliotecii dinamice. Interface. Implementation. Conectarea statică și dinamică a bibliotecii dinamice. Biblioteci dinamice de funcții și resurse.	1
8.	<b>Clasa Canvas.</b> Principiile de lucru cu grafica. Funcții standard pentru desenarea unor figuri geometrice. Metode de selectare a culorii. Modificarea reprezentării grafice a obiectului.	1
<b>Total</b>		<b>12</b>

### Laboratoare

Nr. d/o	Subiectele de studiu	Ore
		L
1.	Lucrarea de laborator nr. 1. Crearea claselor simple.	2
2.	Lucrarea de laborator nr. 2. Crearea claselor prin moștenire.	2
3.	Lucrarea de laborator nr. 3. Supraîncărcarea funcțiilor. Clase template.	4
4.	Lucrarea de laborator nr. 4. Crearea aplicațiilor.	2

5.	Lucrarea de laborator nr. 5. Aplicația “Calculatorul”	2
6.	Lucrarea de laborator nr. 6. Aplicația “Redactor text”	2
7.	Lucrarea de laborator nr. 7. Utilizarea resurselor sistemului de operare	4
8.	Lucrarea de laborator nr. 8. Crearea bibliotecilor dinamice și interacțiunea cu ele	2
9.	Lucrarea de laborator nr. 9. Aplicația “Redactor grafic”	4*
<b>Total</b>		<b>24</b>

\* Pentru specialitate “Matematica și informatica” lucrarea de laborator nr. 9 este exclusă (astfel numărul total de ore pentru specialitatea nominalizată va fi de 20 de ore)

### **Principiile de lucru în cadrul unității de curs**

1. Calendarul cursului (termenii-limită de prezentare a sarcinilor propuse spre rezolvare, momentele de evaluare etc.) este corelat cu calendarele la alte unități de curs din semestru. De aceea prezentarea sarcinilor după termenul-limită indicat în calendar nu este salutăată, iar studenții care amână frecvent prezentarea sarcinilor își formează o imagine nefavorabilă.

2. Nu este salutăată întârzierea la ore.

3. Este salutăată poziția activă a studentului, care studiază din propria inițiativă noi conținuturi, propune soluții, formulează întrebări în cadrul prelegerilor și a orelor practice.

4. În cadrul unității de curs o atenție sporită va fi oferită respectării principiilor etice. Prezentarea unor soluții a sarcinilor, preluate de la colegi sau din alte surse, preluarea informațiilor din diverse surse, fără a face trimitere la sursă, va fi considerată plagiat și va fi sancționată prin note de „1” .

### **Resursele informaționale la unitatea de curs**

#### **Obligatorie**

1. Bertrand Mayer. Object-Oriented Software Construction. Prentice Hall, 1997.
2. Павловская Т.А.; Щупак Ю.А. С++. Объектно-ориентированное программирование. Практикум. Питер, 2005.

#### **Opțională**

3. Лафоре Роберт. Объектно-ориентированное программирование в С++. Питер, 2004.
4. Java Programming Tutorial. Object-oriented Programming Basics. Disponibil on-line pe adresa: [www.ntu.edu.sg](http://www.ntu.edu.sg).

### **Evaluarea**

Cunoștințele, capacitățile și competențele studenților vor fi evaluate:

- în cadrul lecțiilor practice (conform calendarului unității de curs);
- prin susținerea a două teste la partea teoretică prin intermediul platformei de învățare Moodle.
- la examenul final (conform orarului întocmit de decanat).

Nota finală la unitatea de curs „Programarea orientată pe obiect I” se calculează conform formulei:

$$N_f = 0,6 * n_s + 0,4 * n_e,$$

unde  $N_f$  – nota finală;  $n_s$  – media notelor obținute pe parcursul semestrului,  $n_e$  – nota de la examen.

Media notelor obținute pe parcursul semestrului se calculează conform formulei:

$$n_s = 0,2 * n_t + 0,8 * n_l,$$

unde  $n_t$  este media notelor de la teste,  $n_l$  – media notelor de la lucrările de laborator.

Examenul final se susține la calculator prin realizarea unei sarcini complexe.