

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

## **Curriculum pentru unitatea de curs**

# **„Limbaje de programare structurată”**

Ciclul I – studii superioare de licență

Codul și denumirea domeniului general de studii:

061 Tehnologii ale informației și comunicațiilor

Specialitatea: 0613.1 Tehnologia informației

Forma de învățământ: cu frecvență

### **Autori:**

conf. univ., dr. Diana MOGLAN

lector univ. Vitalie ȚÎCĂU

**BĂLȚI, 2021**

Curriculum-ul a fost discutat la ședința Catedrei de Matematică și Informatică,

proces-verbal nr. 8 din 12.02. 2021.

Șeful Catedrei de Matematică și Informatică,

dr., conf. univ. Corina NEGARA *Corina*.

Curriculum-ul a fost aprobat la ședința

Consiliului Facultății de Științe Reale, Economice și ale Mediului,

proces-verbal nr. 11 din 16.04. 2021.

Decanul Facultății de Științe Reale, Economice și ale Mediului,

dr. conf. univ. Ana CIOBANU *Ciobanu*



## Informații de identificare a unității de curs

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

**Catedra:** matematică și informatică

**Domeniile generale de studiu:** 011 Științe ale educației

**Domeniul de formare profesională la ciclul I:** 0114 Formarea profesorilor

**Specialitatea:** 0114.1 Matematică și 0114.2 Informatică (MI), ciclul I, studii superioare de licență

**Denumirea unității de curs:** Limbaje de programare structurată.

## Administrarea unității de curs

Plan licență	Codul unității de curs	Credite ECTS	Total ore	Repartizarea orelor				Forma de evaluare	Limba de predare
				Curs	Sem.	Lab.	Lucrul ind.		
TI	S.02.O.011	6	150	30	–	60	90	Examen	Rom/Rus

**Anul de studiu și semestrul în care se studiază disciplina:** Anul I, Semestrul 1

**Forma de organizare a învățământului:** Cu frecvență

**Regimul disciplinei (obligatorie/opțională/la liberă alegere):** Obligatorie

**Categoria formativă:** Unitate de curs de specializare.

## Informații referitoare la cadrele didactice

**Diana MOGLAN**, dr. în științe pedagogice, conferențiar universitar, absolventa Universității de Stat „Alec Russo” din Bălți, specialitatea „Matematică și informatică”. A efectuat studiile de master la specialitatea „Informatică”. A susținut teza de doctor în pedagogie la specialitatea „Teoria și metodologia instruirii (științe exacte și ale mediului)”.

*E-mail:* mogdiana@gmail.com

*Telefon:* 0-231-52-4-88

*Skype:* mogdiana

*Orele de consultații:* miercuri, 15:00 – 16:30. Consultațiile se oferă atât în regim „față-în-față”, cât și prin utilizarea poștei electronice sau prin Skype.

**Vitalie ȚÎCĂU**, lector universitar la Catedra de matematică și informatică. A absolvit Universitatea de Stat din Moldova, Facultatea de Matematică și Cibernetică, specialitatea „Matematica aplicată”. A absolvit studii de doctorat la specialitatea „Matematica de calcul”, la Universitatea de Stat din Moldova, municipiul Chișinău, Republica Moldova. A realizat publicații aplicative și metodice cu tematica: „Structuri discrete”, „Limbaje de programe

orientate obiect”, „Analiza numerică”, „Prelucrarea informației grafice”, “Rezolvări de probleme din domeniul informaticii”.

*Domeniul de interes științific:* Informatica, învățare electronică, Dezvoltarea programelor, calcul numeric.

*Biroul:* Sala de calculatoare 140, decanatul ȘREM, DTI USARB

*Telefon:* 0-231-52-4-88; 0-231-52-3-98; 0-231-52-3-84

*E-mail:* vitalie.ticau@usarb.md

### **Integrarea unității de curs în programul de studii**

Cursul „Limbaje de programare structurată” se predă în semestrul II la toate specialitățile de informatică. Este o disciplină de specializare. Rezolvarea unei probleme la calculator parcurge 4 etape: specificarea problemei, proiectarea rezolvării (algoritmului), codificarea rezolvării (programarea propriu-zisă), exploatarea și întreținerea problemei. Cursul are ca obiectiv familiarizarea studenților cu etapa a treia – codificarea rezolvării (programarea propriu-zisă).

Cursul „Limbaje de programare structurată” este o continuare a cursului fundamental „Bazele programării” și contribuie la dezvoltarea competenței de programare a studenților aplicând diverse limbaje de programare, în special limbajul C/C++. Acest curs este intermediar și are o continuitate în cursurile „Programare orientată pe obiect”, „Analiza numerică”, etc.

Viitorii programatori vor învăța să aplice diverși algoritmi, să folosească cele mai optimale metode.

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

#### **1. Cunoștințe:**

- cunoașterea arhitecturii și principiului de funcționare al calculatorului;
- cunoașterea reprezentării în memoria calculatorului a numerelor întregi și/sau reale cu și fără semn, a informației caracteriale;
- posedarea la nivel teoretic și aplicativ a informaticii preuniversitare în limitele standardelor de studiu eficient al învățării.

#### **2. Deprinderi:**

- utilizarea eficientă a resurselor sistemelor de calcul, sistemelor de operare și ale Internetului;
- posedarea deprinderilor de elaborare și analiză a algoritmilor de rezolvare a problemelor practice în baza precizării problemei;

- posedarea deprinderilor de lucru cu MS OFFICE: elaborare de documente WORD, prezentări Power Point etc.
3. Posedarea la nivel teoretic și aplicativ a cunoștințelor și abilităților dobândite în cadrul disciplinei preuniversitare „Informatica” în limitele standardelor de studiu eficient al învățării.

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

În cadrul studierii unității de curs „Limbaje de programare structurată” la studenți vor fi dezvoltate următoarele competențe (CP – competențe profesionale; CT – competențe transversale):

- CP2.** Elaborarea modelelor pentru descrierea fenomenelor și proceselor reale.
- CP3.** Proiectarea, elaborarea și analiza algoritmilor pentru rezolvarea problemelor;
- CP6.** Prelucrarea datelor, analiza și interpretarea lor.
- CT2.** Desfășurarea eficientă și eficace a activității organizate în echipă.

### **Finalitățile unității de curs**

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

- să analizeze paradigmele de programare structurată și programare modulară;
- să proiecteze algoritmul și etapele de elaborare și rezolvare a problemei;
- să rezolve probleme concrete, bazate pe aplicarea tipurilor de date omogene și mixte;
- să rezolve probleme concrete evidențiind aplicarea structurilor dinamice și statice de date;
- să rezolve probleme concrete prin aplicarea algoritmilor clasici de programare;
- să lămurească pas cu pas îndeplinirea programului în baza exemplurilor concrete.

### **Conținutul unității de curs și metodele de instruire**

Unitatea de curs „Limbaje de programare structurată” este divizată în două unități de învățare. În următorul tabel este prezentată denumirea și conținutul scurt al temelor la desfășurarea prelegerilor (**P** – ore prelegeri; **LI** – ore lucru independent)

Nr. d/o	Denumirea și conținutul scurt al temei	Curs, ore P / LI
<b>Unitatea de conținut 1: Structura unui program. Instrucțiuni. Tipuri de date simple, omogene</b>		
1.	Generalități privind limbajele de programare structurată. Medii de programare. Sintaxa și alfabetul limbajului. Cuvintele cheie. Blocuri de instrucțiuni.	1/1
2.	Tipuri de date. Constante și variabile. Declararea variabilelor. Adresele și inițializarea variabilelor.	1/1
3.	Funcțiile de citire și de scriere a datelor în mod standard.	1/1

Nr. d/o	Denumirea și conținutul scurt al temei	Curs, ore P / LI
4.	Operatori: aritmetici, de atribuire, de incrementare și decrementare, relaționali, logici, la nivel de bit, condițional, de rezoluție.	1/1
5.	Instrucțiuni de decizie: <i>if, if-else, switch</i> .	1/1
6.	Instrucțiuni de ciclare: <i>for, while, do-while</i> . Clasificarea ciclurilor. Instrucțiunile <i>break</i> și <i>continue, goto, exit</i> .	2/2
7.	Definirea și declararea funcțiilor. Apelul funcțiilor. Parametri formali și actuali. Transmiterea parametrilor prin: valoare, adresă, referință. Variabile globale și locale. Variabile automate, statice, externe. Durata de viață a variabilelor.	2/2
8.	Directive de preprocesare: <i>#define, #include, #if, #ifdef, #ifndef, #endif, #else</i> .	1/1
9.	Pointeri. Declararea și utilizarea pointerilor. Utilizarea pointerilor ca parametri ai funcțiilor. Mecanismul de indirectare.	1/1
10.	Tabele. Tabele unidimensionale. Inițializarea tabelului. Accesarea elementelor unui vector. Utilizarea tabelelor ca pointeri. Algoritm de căutare în vector. Determinarea valorii minime și maxime. Algoritm de formare a unui vector din alți vectori. Algoritm de deplasare a elementelor unui vector. Algoritmi de rotire a elementelor unui vector. Algoritmi de sortare a vectorilor.	2/2
11.	Tabele bidimensionale. Accesul la elementele tabelului. Prelucrarea tabelelor pe rânduri sau coloane. Algoritmi de formare a tabelelor. Determinarea elementelor minime și maxime. Includerea unui rând / unei coloane în tabel. Excluderea unui rând sau a unei coloane din tabel. Generarea matricelor.	2/2
<b>Unitatea de conținut 2: Șiruri de caractere. Tipuri de date derivate. Apeluri recursive</b>		
12.	Șiruri de caractere. Utilizarea șirurilor de caractere ca pointeri. Funcții standard de prelucrare a șirurilor de caractere.	2/2
13.	Structuri și uniuni. Accesarea elementelor structurii. Structuri înlănțuite. Inițializarea structurilor. Redenumirea tipurilor de date. Pointeri la structuri. Câmpuri. Tipul enumerare.	2/2
14.	Fișiere. Modul standard de intrare-ieșire. Clasificarea fișierelor. Deschiderea și închiderea unui fișier. Modul de acces la deschiderea fișierelor. Scrierea și citirea fișierelor text de tipul: caracter, șir de caracter, formatat.	2/2
15.	Fișiere binare. Scrierea și citirea fișierelor binare de tip înregistrare. Accesarea directă și secvențială a articolelor unui fișier. Fișiere standard. Operații de actualizare, service și documentare.	2/2
16.	Recursia. Recursie directă și indirectă.	1/1
17.	Supraîncărcarea funcțiilor. Funcții inline. Funcții cu parametri implicați, cu număr variabil de parametri.	1/1
18.	Structuri dinamice de date. Stiva. Coada. Lista liniară. Liste dublu înlănțuite și liste dublu înlănțuite cu santinele. Implementarea stivelor, cozilor, listelor. Operațiile posibile cu structurile dinamice de date.	2/2
19.	Elaborarea programelor complexe. Fișiere proiect. Etapele de realizare a unui program complex. Modele de memorie. Tipurile de pointeri.	1/1
20.	Elemente fundamentale ale POO. Clase și obiecte. Modificatorii de protecție. Mesaje și metode. Constructori. Destructorii. Moștenirea. Polimorfismul. Ierarhii de clase. Funcții virtuale.	2/2
<b>Total</b>		<b>30/30</b>

Orele de laborator sunt organizate în formă de realizare a 7 proiecte de program și o lecție de inițiere. În următorul tabel este prezentată tematica proiectelor de program și repartizarea orelor (**L** – ore de laborator, **LI** – ore lucru independent).

<b>Nr. d/o</b>	<b>Tema</b>	<b>Nr. de ore L / LI</b>
1.	Mediul de programare. Înscrierea programelor. Compilarea. Link - editarea. Lansarea în execuție. Instrucțiuni de intrare – ieșire. Trasarea pe pași a programului. Utilizare Help.	2/2
2.	Proiect de programe Nr 1. Organizarea operațiilor la nivel de bit. Algoritmi clasici.	8/8
3.	Proiect de programe Nr 2. Organizarea ciclurilor în program.	8/8
4.	Proiect de programe nr. 3. Organizarea ciclurilor în program. Gestiunea tabelor de numere. Pointeri. Funcții.	8/8
5.	Proiect de programe nr. 4. Gestiunea tabelor bidimensionale de numere.	8/8
6.	Proiect de programe nr. 5. Șiruri de caractere. Fișiere text.	8/8
7.	Proiect de programe nr. 6. Structuri. Fișiere binare.	8/8
8.	Proiect de programe nr. 7 (opțional). Structuri dinamice.	8/8
9.	Evaluare	2/2
<b>Total</b>		<b>60/ 60</b>

### **Activități de lucru individual**

Activitatea de lucru individual este o componentă obligatorie a activității de învățare la disciplină. În cadrul studierii acestui curs studenților li se vor propune o serie de teme și probleme care urmează a fi studiate și soluționate independent. Însărcinările practice pentru lucrul individual sunt propuse în cadrul orelor de laborator.

Pentru unitatea de curs este creat un suport de susținere a activităților la curs pe platforma MOODLE. Odată cu începerea cursului studenții sunt înregistrați la curs pe platforma MOODLE și trebuie să realizeze diferite sarcini propuse pe platformă (studierea materialului, realizarea testelor grilă, rezolvarea problemelor).

La predarea conținutului unității de curs respective vor fi aplicate următoarele metode de predare și învățare: expunerea, conversația, algoritmizarea, exercițiul și problematizarea.

Pentru asigurarea realizării metodelor didactice menționate se vor utiliza suportul de curs, cursul de instruire pe platforma universitară, culegere de prezentări Power Point, consultații individuale.

Studentul trebuie să realizeze un set de probleme cu însărcinări practice, unde sunt indicate: scopul lucrării, formularea problemei, date inițiale, indicații metodice și recomandări,

cerințele față de program și variantele însărcinărilor. În special sunt prezente sarcini de implementare a materialului teoretic cu aplicații în practică și a cercetării individuale. În special sunt prezente sarcini de implementare a materialului teoretic cu aplicații în practică.

Activitatea individuală pe parcursul semestrului va fi apreciată cu 3 note la finele semestrului pe următoarele teme de cercetare:

1. *Instrucțiuni alternative și iterative*. Aplicarea algoritmilor clasici din teoria numerelor cu determinarea de numere perfecte, Fibonacci, Lucas, Pell, Cullen, prime, divizibile, palindroame, Demlo, pare / impare, fericite, pandigitale, abundente, puternice, Armstrong, ondulatorii netriviiale, aspirante, briliante, Devlali, norocoase, etc. ( $L_1$ )

2. *Șiruri de caractere de tip string*. Diferențierea tipului *string* de tipul *char* \*. Aplicarea algoritmilor determinării / ștergerii / înlocuirii cuvintelor dintr-un text cu o proprietate anumită cuvântul din text cu lungime maximă, cuvintele numai din litere minuscule / majuscule, ce conțin cel puțin 2 litere indicate, cuvintele, care încep și se termină cu una și aceeași literă, cu lungimea mai mare ca 6 caractere, care încep cu litera dată, palindroame, ce conțin orice vocală doar o singură dată, ce conțin litera indicată de cel puțin 2 ori, cu litere duble (aceleași 2 litere consecutive), ce conțin toate vocalele (a, e, i, o, u), etc. ( $L_2$ )

3. *Aplicarea structurilor dinamice la implementarea operațiilor de actualizare a datelor din fișierele binare*: aranjarea (ștergerea, adăugare, modificarea) datelor, referitor la diferite contexte practice: grupa de studenți, clasa de elevi, datele zodiacului, salariile angajaților, datele sportivilor, cărțile din bibliotecă, gara auto, țările, medicamentele din farmacie, rețele sociale, agenda de telefoane, indicator de cuvinte, tarife comunale, gara de tren, etc. ( $L_3$ )

Nota pentru activitatea individuală se va calcula ca media a 3 note:

$$N_i = \frac{L_1 + L_2 + L_3}{3}.$$

### **Evaluarea**

Cunoștințele, abilitățile și competențele studenților vor fi evaluate pe parcursul semestrului cu trei note.

1. Nota evaluării curente  $N_{ec}$ , va fi determinată prin calculul mediei notelor obținute în cadrul orelor de laborator prin rezolvarea proiectelor de program (minimum 6 note).
2. Nota evaluării periodice  $N_p$  va fi determinată după studiul a jumătate din partea teoretică a cursului prin susținerea unui test grilă cu itemi teoretici și practici pe platforma MOODLE. Nota obținută se calculează conform baremului prestabilit.
3. Nota evaluării lucrului individual  $N_i$ .

La evaluarea finală sunt admiși studenții care întrunesc următoarea condiție – toate notele  $N_{ec}$ ,  $N_p$  și  $N_i$  au valoarea de cel puțin 5.



Nota medie semestrială  $N_S$  se calculează ca media aritmetică a componentelor de mai sus conform formulei:

$$N_S = (N_{ec} + N_p + N_i)/3.$$

Nota medie semestrială constituie 60% din nota generală la unitatea de curs. Celelalte 40% le constituie nota de la examen.

*Evaluarea finală* se realizează în formă de examen scris – test cu itemi similari cu itemi de la testele de autoevaluare de pe platforma de învățare MOODLE. În total sunt 3 itemi de evaluare a cunoștințelor teoretice, 6 itemi de analiză a algoritmilor și 3 sarcini practice de definire a unei funcții pe unitățile de conținut nr. 1 și nr. 2.

Chestionarul pentru examen conține doar temele indicate în conținuturi.

Durata desfășurării examenului – 90 min. Studentul la examen este apreciat cu o notă ( $N_E$ ) conform baremului.

Nota finală ( $N_F$ ) se calculează după formula:

$$N_F = 0.6 \times N_S + 0.4 \times N_E.$$

Modele de teste grilă de evaluare curentă, periodică și finală sunt plasate pe platforma

### Model de test de evaluare finală

#### Catedra de matematică și informatică

Aprob:

Șeful Catedrei \_\_\_\_\_

Probă de evaluare finală la unitatea de curs „Limbaje de programare structurată”

#### Testul Nr. 1

<b>În itemii 1-3 scrie răspunsurile corecte (1p) // Для итемов 1-3 напиши правильные ответы (1p)</b>				
<b>1. Biblioteca ce conține definiții ale constantelor și funcțiilor standard pentru prelucrarea șirurilor de caractere: //</b> Библиотека, содержащие стандартные определения функций обработки строк символов:				
<i>math.h</i>	<i>conio.h</i>	<i>iostream.h</i>	<i>string.h</i>	
	<i>stdlib.h</i>			
<b>2. Scrie tipurile de date structurale: // Напиши структурные типы данных:</b>				
<i>int</i>	<i>struct</i>	<i>union</i>	<i>FILE</i>	<i>double</i>
<b>3. Scrie operatorii logici // Напиши логические операторы</b>				
+	+=	++	--	=
*	*=	/	/=	%
%=	==	=	<	!
<=	>=	&&	//	~
!=	?:	<<	>>	
<b>În itemii 4-9 scrie ce va fi afișat pe ecran, ținând cont de sugestii (1p). Argumentează răspunsul prin-o frază. (1p) // Для итемов 4-9 напиши что будет выведено на экран (1p). Аргументируй ответ одной фразой.(1p)</b>				

<p><b>4. Tema: Operatori (2p)</b></p> <pre>int a = 1234; int b = a%100; a = a/100; cout &lt;&lt; b*100+a;</pre> <p><i>1234      1243      2134      3421      34121</i></p>	<p><b>5. Tema: Instrucțiuni iterative (2p)</b></p> <pre>int n = 10; for(int i=0; i&lt;n/2; i++)     cout &lt;&lt; 9-2*i &lt;&lt; ' ' &lt;&lt; 10-2*i;</pre> <p><i>2 1 4 3 6 5 8 7 10 9                      9 10 7 8 5 6 3 4 1 2</i>  <i>1 2 3 4 5 6 7 8 9 10                      10 9 8 7 6 5 4 3 2 1</i></p>
<p><b>6. Tema: tabele unidimensionale (2p)</b></p> <pre>int i, n = 10; int a[] = {2,10,7,3, 5, 8, 9, 1, 6, 4}; int aux = 0; for(i=0; i&lt;n; i++)     if(a[i]%2 != 0) aux += a[i]; cout &lt;&lt; aux;</pre> <p><i>30              55              25              15</i></p>	<p><b>7. Tema: tabele bidimensionale (2p)</b></p> <pre>int a[5][5] = {{2, 10, -4, 3}, {1, 5, 7, 8}, {9, 6, 0, 4}}; int aux = a[0][0]; for(int i=1; i&lt;3; i++)     for(int j=0; j&lt;4; j++)         if(a[i][j] &gt; aux) aux = a[i][j]; cout &lt;&lt; aux;</pre> <p><i>0              9              -4              10</i></p>
<p><b>8. Tema: Șiruri de caractere (2p)</b></p> <pre>char fraza[81] = "In anii 2000-2019 C++ ramine inca un limbaj popular!"; for(int i=0; i&lt;strlen(fraza); i++)     if((fraza[i] &gt;= 'a') &amp;&amp; (fraza[i] &lt;= 'z'))         fraza[i] -= 32; cout &lt;&lt; fraza;</pre> <p><i>IN ANII 2000-2019 C++ RAMINE INCA UN LIMBAJ POPULAR!</i>  <i>In Anii 2000-2019 C++ Ramine Inca Un Limbaj Popular!</i>  <i>In anii 2000-2019 C++ ramine inca un limbaj popular!</i>  <i>in anii 2000-2019 c++ ramine inca un limbaj popular!</i></p>	<p><b>9. Tema: Structuri (2p)</b></p> <pre>struct Student { int id; char nume[10]; float media; }; char name[10]; float med = 0; Student grupa[10] = {{1,"Ana",8.00}, .... }; for(int i = 0; i&lt;10; i++)     if(grupa[i].media &gt; med) {         strcpy(name, grupa[i].nume);         med = grupa[i].media;     } cout &lt;&lt; name;</pre> <p><i>Ionela              Ana              Ion              Igor</i></p>
<p><b>10. Elaborarea programului (6p)</b> Se dă un tabel, format din <math>N \leq 30</math> numere – temperaturile zilnice. Definește o funcție, care va determina (va returna) temperatura cea mai mare negativă. // Задан массив из <math>N</math> чисел – ежедневные температуры. Определи функцию для выявления (возврата) самой высокой отрицательной температуры.</p>	
<p><b>11. Elaborarea programului (7p)</b> Definește o funcție, în care dintr-un fișier text <i>Date1.txt</i> de înscris în alt fișier text <i>Date2.txt</i> cuvintele-palindroame. // Определи функцию, в которой из текстового файла <i>Date1.txt</i> переносятся в другой текстовый файл <i>Date2.txt</i> слова-палиндромы.</p>	
<p><b>12. Elaborarea programului (8p)</b> În fiecare rând a fișierului text <i>Date.in</i> se dau numele, clasa și 4 note ale fiecărui elev. De scris o secvență de program, care înscrie în fișierul binar <i>Date.out</i> datele, aranjate în ordinea descrescătoare a notei medii, utilizând liste dinamice. // В каждой строке текстового файла <i>Date.in</i> записаны имена, класс и 4 оценки учеников. Напиши фрагмент записи данных в двоичный файл <i>Date.out</i>, отсортированных в порядке убывания среднего балла, применяя динамические списки.</p>	

Punctaj	0-1	2-5	6-9	10-13	14-17	18-21	22-25	26-30	31-33	34-36
Nota	1	2	3	4	5	6	7	8	9	10

Titularul:  
Data 2021 cursului

## Resurse informaționale la unitatea de curs

### Obligatorii:

1. MOCRIENCO, FI.; VLĂDOIU, D. *Tehnologia informației. Programarea calculatoarelor*. București, 2006, 214 p. ISBN 10 973-0-04555-0.
2. CHERCEZ, E; ȘERBAN, M. *Programarea în limbajul C/C++ pentru liceu*. Vol. I, II și III. Iași, Polirom, 2005-2010, 586 p. ISBN 937-681-868-3.
3. NEGRESCU, L. *Limbajele C și C++ pentru începători*. Volumul 1. Limbajul C. Volumul 2. Limbajul C++. Cluj, Cartea albastră, 1996, 783 p. ISBN 973-86980-9-3.
4. ÎNTUNERIC, A.; SICHIM, Cr. *Informatică. Teste grilă C/C++*. Iași, Polirom, 2003, 224 p. ISBN 973-681-185-9.
5. СТРАУСТРУП, Б. *Язык программирования C++*. Краткий курс 2-е изд. Издательская группа "Диалектика-Вильямс", 2019, 320 с. ISBN 978-5-907144-12-5, 978-0-13-499783-4.
6. СЕДЖВИК, С. *C/C++*. *Алгоритмы на C++*. М.: Мир, 2001. Издательская группа "Диалектика-Вильямс", 2019, 1056 с. ISBN 978-5-907144-21-7, 978-0-321-60633-4.
7. ШИЛДТ, Г. *C++ from the Ground Up / C++: базовый курс*, Издательская группа "Диалектика-Вильямс", 2019, 624 с. ISBN 978-5-907144-15-9.

### Suplimentare:

1. BARKAKATI, N. *Borland C++*. *Ghidul programatorului*. București: Teora, 1997.
2. BALTAȘ, I. *Programarea în limbajele C și C++*. București, Teora, 1996.
3. BOTEZ, C. GOSPODARU, M. *Limbaje de programare*. Introducere în limbajul C. Iași, 1998.
4. NAMIR, C. Shamas. *Curs rapid de Borland C++ 4*. București: Teora, 1996 (trad.engl.).
5. СТРАУСТРУП, Б. *Язык программирования C++*, 3-е изд. СПб.: Невский Диалект, 1999.
6. ПОЛ, А. *Объектно-ориентированное программирование на C++*. СПб.: Питер, 2001.
7. RÎNGACI, V. D.; LATUL, G. V.; SOCOL V. V. *Programare în limbajul Turbo C*. (în 3 părți). Chișinău, 1994.
8. BOTEZ, C.; NEAGA, C. *Limbajul de programare C*. Exemple. Iași, 1997.

## **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 discipline 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 disciplinei 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” .
5. Înregistrarea pe platforma de învățare MOODLE și înscrierea la cursul electronic „Limbaje de programare structurată I” este obligatorie.