



# UNIVERSITATEA TEHNICĂ A MOLDOVEI

**Valentina TÎRȘU**

## PROGRAMARE

### **Ghid metodic pentru lucrările de laborator**

```
include <stdio.h>  
include <string.h>  
int main ()  
{  
    char checkword[80] = "password"  
    char password[80] = "";  
    do  
    {  
        printf ("Enter password: ");  
        scanf ("%s", password);  
    }  
    while (strcmp(password, ch  
    return 0;  
}
```

**Chișinău  
2022**

**UNIVERSITATEA TEHNICĂ A MOLDOVEI**  
**FACULTATEA ELECTRONICĂ ȘI TELECOMUNICAȚII**  
**DEPARTAMENTUL TELECOMUNICAȚII**  
**ȘI SISTEME ELECTRONICE**

**PROGRAMARE**

**Ghid metodic**  
**pentru lucrările de laborator**

**Chișinău**  
**Editura „Tehnica-UTM”**  
**2022**

**CZU 004.4(076.5)**  
**T 63**

Lucrarea a fost discutată și aprobată pentru editare la consiliul Facultății Electronice și Telecomunicații, proces verbal nr.03 din 10.11.2022.

Disciplina *Programare* prevede studierea limbajului de programare C/C++ și este studiată la Facultatea Electronică și Telecomunicații, specialitățile *Tehnologii și sisteme de telecomunicații, Rețele și software de telecomunicații, Inginerie și management în telecomunicații*. Studiul acestei discipline condiționează acumularea cunoștințelor necesare studenților privind programarea structurată cu scopul acumulării experienței practice în programarea calculatoarelor.

Autor: conf. univ., dr. Valentina TÎRȘU

Recenzent: conf. univ., dr. Sergiu TUTUNARU

**DESCRIEREA CIP A CAMEREI NAȚIONALE A CĂRȚII DIN RM**

**Tîrșu, Valentina.**

Programare: Ghid metodic pentru lucrările de laborator / Valentina Tîrșu; Universitatea Tehnică a Moldovei, Facultatea Electronică și Telecomunicații, Departamentul Telecomunicații și Sisteme Electronice.

– Chișinău: Tehnica-UTM, 2022. – 130 p.: fig., tab.

Aut. indicat pe vs. f. de tit. – Referințe bibliogr. la sfârșitul temelor. – 50 ex.

**ISBN 978-9975-45-861-0**

**© UTM, 2022**

## CUPRINS

PRELIMINARII.....	6
Lucrarea de laborator nr.1. Interfața grafică Dev C++. Scrierea programelor simple.....	12
1.1. Instalarea mediului de programare DEV C++.....	12
1.2. Crearea, redactarea și salvarea programului.....	13
1.3. Compilarea și executarea programului.....	16
1.4. Diagnosticarea erorilor și avertizările emise de compilator și editorul de legături.....	20
1.5. Subiecte de evaluare.....	21
Bibliografie.....	23
Lucrarea de laborator nr.2. Programe secvențiale.....	24
2.1. Sarcina pentru soluționare.....	24
2.2. Exemplu de soluționare a sarcinii.....	25
2.3. Subiecte de evaluare.....	29
Bibliografie.....	30
Lucrarea de laborator nr.3. Scrierea expresiilor matematice.....	31
3.1. Sarcina pentru soluționare.....	31
3.2. Exemplu de soluționare a sarcinii.....	32
3.3. Subiecte de evaluare.....	35
Bibliografie.....	36
Lucrarea de laborator nr.4. Deplasări pe biți (afișarea variabilelor în diferite sisteme de numerotație).....	37
4.1. Sarcina pentru realizare.....	37
4.2. Exemplu de soluționare a sarcinii.....	38
4.2.1. Programul arhivării.....	38
4.2.2. Programul de dezarhivare.....	42
4.3. Subiecte de evaluare.....	45
Bibliografie.....	46
Lucrarea de laborator nr.5. Aplicarea instrucțiunilor decizionale (if).....	47
5.1. Sarcina pentru realizare.....	47
5.2. Exemplu de soluționare a sarcinii.....	48

5.3. Subiecte de evaluare.....	52
Bibliografie.....	53
Lucrarea de laborator nr.6. Aplicarea instrucțiunilor decizionale (switch).....	54
6.1. Sarcina pentru soluționare.....	54
6.2. Exemplu de soluționare a sarcinii.....	54
6.3. Subiecte de evaluare.....	58
Bibliografie.....	59
Lucrarea de laborator nr.7. Aplicarea instrucțiunilor repetitive.....	60
7.1. Sarcina pentru soluționare.....	60
7.2. Exemplu de soluționare a sarcinii.....	60
7.3. Subiecte de evaluare.....	68
Bibliografie.....	69
Lucrarea de laborator nr.8. Aplicarea instrucțiunilor repetitive imbricate.....	70
8.1. Sarcina pentru soluționare.....	70
8.2. Exemplu de soluționare a sarcinii.....	70
8.3. Subiecte de evaluare.....	78
Bibliografie.....	79
Lucrarea de laborator nr.9. Prelucrarea tablourilor unidimensionale în C/C++.....	80
9.1. Sarcina pentru soluționare.....	80
9.2. Exemplu de soluționare a sarcinii.....	81
9.3. Subiecte de evaluare.....	88
Bibliografie.....	88
Lucrarea de laborator nr.10. Prelucrarea tablourilor bidimensionale în C/C++.....	89
10.1. Sarcina pentru soluționare.....	89
10.2. Exemplu de soluționare a sarcinii.....	90
10.3. Subiecte de evaluare.....	94
Bibliografie.....	95
Lucrarea de laborator nr.11. Prelucrarea șirurilor de caractere în C/C++.....	96
11.1. Sarcina pentru soluționare.....	96
11.2. Exemplu de soluționare a sarcinii.....	96

11.3	Subiecte de evaluare.....	102
	Bibliografie.....	103
Lucrarea de laborator nr.12. Prelucrarea tablourilor prin intermediul pointerilor.....		
		104
12.1.	Sarcina pentru soluționare.....	104
12.2.	Exemplu de soluționare a sarcinii.....	104
12.3.	Subiecte de evaluare.....	109
	Bibliografie.....	110
Lucrarea de laborator nr.13. Funcții în limbajul C/C++.....		
		111
13.1.	Sarcina pentru soluționare.....	111
13.2.	Exemplu de soluționare a sarcinii.....	111
13.3.	Soluționarea algoritmului programului (var.nr. 1)...	114
13.4.	Soluționarea algoritmului programului (var.nr. 2)...	118
13.5.	Subiecte de evaluare.....	121
	Bibliografie.....	122
Anexe.....		
		123
Anexa 1.....		
		123
Anexa 2.....		
		124
Anexa 3.....		
		127

## PRELIMINARII

Disciplina *Programare* prevede studierea limbajului de programare C/C++ și este studiată la Facultatea Electronică și Telecomunicații, specialitățile *Tehnologii și sisteme de telecomunicații, Rețele și software de telecomunicații, Inginerie și management în telecomunicații*. Studiul acestei discipline condiționează acumularea cunoștințelor necesare studenților privind programarea structurată cu scopul acumulării experienței practice în programarea calculatoarelor.

Sarcinile disciplinei sunt:

- studierea limbajului de programare C/C++ pentru a căpăta cunoștințe teoretice și practice în programarea calculatoarelor;
- formarea deprinderilor practice de bază necesare pentru aplicarea calculatorului în rezolvarea problemelor din diverse domenii, precum și pentru a fi la curent cu cerințele actuale în acest domeniu.

Legăturile interdisciplinare ale disciplinei *Programare* sunt asigurate, prin continuitate, de cursul liceal *Informatica* și de studierea ulterioară a disciplinelor *Programarea calculatoarelor și limbaje de programare* și *Limbaje de programare specializate*, care au la bază limbajul C/C++.

Competențe profesionale dezvoltate de disciplină:

- CP3. Aplicarea cunoștințelor, conceptelor și metodelor de bază privind arhitectura sistemelor de calcul, microprocesoare, microcontrolere, limbaje și tehnici de programare:

- ✓ descrierea funcționării unui sistem de calcul, a principiilor de bază ale arhitecturii microprocesoarelor și microcontrolerelor de uz general, a principiilor generale ale programării structurate;

- ✓ utilizarea unor limbaje de programare de uz general și specifice aplicațiilor cu microprocesoare și microcontrolere; explicarea funcționării unor sisteme de control automat care folosesc aceste arhitecturi și interpretarea rezultatelor experimentale;
  - ✓ rezolvarea problemelor practice concrete care includ elemente de structuri de date și algoritmi, programare și utilizare de microprocesoare sau microcontrolere;
  - ✓ elaborarea de programe într-un limbaj de programare general și/sau specific, pornind de la specificarea cerințelor până la execuție, depanare și interpretarea rezultatelor în corelație cu procesorul utilizat;
  - ✓ realizarea de proiecte care implică componente hardware (procesoare) și software (programare).
- CP4. Conceperea, implementarea și operarea serviciilor de date, voce, video, multimedia bazate pe înțelegerea și aplicarea noțiunilor fundamentale din domeniul comunicațiilor și transmisiunii informației:
- ✓ definirea conceptelor, principiilor și metodelor folosite în domeniile: programarea calculatoarelor, limbaje de nivel înalt și specifice, tehnici CAD de realizare a modulelor electronice, microcontrolere, sisteme electronice programabile, grafică, arhitecturi hardware reconfigurabile;
  - ✓ explicarea și interpretarea cerințelor specifice structurilor hardware și software din domeniile: limbaje de nivel înalt și specifice, tehnici CAD de realizare a modulelor electronice, microcontrolere, sisteme electronice;
  - ✓ identificarea și optimizarea soluțiilor hardware și software ale problemelor legate de electronica industrială, electronica auto, robotică, producția bunurilor de larg consum;



- ✓ utilizarea criteriilor de performanță adecvate pentru evaluare, inclusiv, prin simulare a hardware-ului și software-ului unor activități de servicii în care se folosesc microcontrolere;
  - ✓ proiectarea de echipamente dedicate din domeniile electronicii aplicate, care folosesc microcontrolere, circuite programabile sau sisteme de calcul cu arhitectură simplă, inclusiv a programelor aferente.
- CP5. Selectarea, instalarea, configurarea și exploatarea echipamentelor de telecomunicații fixe sau mobile și echiparea unui amplasament cu rețele uzuale de telecomunicații:
- ✓ definirea elementelor specifice care individualizează dispozitivele și circuitele electronice din domeniile: sisteme automate, gestionarea energiei electrice, electronică auto, electronică video, bunuri de larg consum;
  - ✓ interpretarea calitativă și cantitativă a funcționării circuitelor din domeniile: sisteme automate, gestionarea energiei electrice, electronica auto, electronica video-audio, bunuri de larg consum;
  - ✓ elaborarea specificațiilor tehnice, instalarea și exploatarea echipamentelor din domeniile electronicii aplicate;
  - ✓ evaluarea, pe baza criteriilor de calitate tehnică și de impact asupra mediului, a echipamentelor din domeniile electronicii aplicate;
  - ✓ proiectarea, folosind principii și metode consacrate ale unor subsisteme de complexitate redusă din domeniile electronicii aplicate.
- CP6. Rezolvarea problemelor specifice rețelelor de comunicații de bandă largă: propagare în diferite medii de

transmisiune, circuite și echipamente pentru frecvențe înalte (microunde și optice):

- ✓ definirea principiilor și metodelor ce stau la baza fabricării, reglajului, testării și diagnosticării aparatelor și echipamentelor din domeniile electronicii aplicate;
- ✓ explicarea și interpretarea proceselor de producție și activităților de mentenanță ale aparaturii electronice, identificând punctele de testare și mărimile electrice de măsurat;
- ✓ aplicarea principiilor de management al activităților de producție, exploatare și service în domeniile electronicii aplicate;
- ✓ utilizarea criteriilor și metodelor de evaluare a calității activităților de producție și service în domeniile electronicii aplicate.

Lucrarea este structurată în 13 capitole care abordează toate temele din curriculumul disciplinei și are scopul de a contribui la studierea instrucțiunilor și a facilităților de programare a limbajului C/C++.

**Sunt propuse următoarele capitole:**

<i>Nr./o</i>	<i>Unități de conținut</i>	<i>Ore</i>	
		<i>curs</i>	<i>Laborator</i>
1	Noțiuni generale. Bazele programării în limbajul C/C++	2	2
2	Limbajul C/C++. Elementele limbajului de programare C/C++.	2	2
3	Limbajul C/C++. Tipuri de date de bază în limbajul C/C++	2	2
4	Funcții de intrare/ieșire în limbajul C/C++	2	2
5	Instrucțiuni decizionale în limbajul C/C++. Instrucțiunea if	2	2

6	Instrucțiuni decizionale în limbajul C/C++. Instrucțiunea switch	2	2
7	Instrucțiuni repetitive în limbajul C/C++. Instrucțiunea for	2	2
8	Instrucțiuni repetitive în limbajul C/C++. Instrucțiunea while, do-while	2	2
9	Tipul de date tablou. Declararea și inițializarea tablourilor	2	2
10	Tipul de date tablou. Accesarea elementelor tabloului	2	2
11	Funcții de prelucrare a șirurilor	2	2
12	Tipul de date tablou. Sortarea tabloului	2	
13	Tipul de date pointer. Declararea, inițializarea și aplicarea variabilelor de tip pointer	2	4
14	Funcții în limbajul C/C++. Noțiunea funcției și prototipul funcției	2	
15	Funcții în limbajul C/C++. Apel către funcții și returnare valori din ele	2	4
<b>Total:</b>		<b>30</b>	<b>30</b>

Fiecare capitol este structurat în următoarele niveluri:

- enunțul sarcinii de soluționare;
- exemplu rezolvat și descris;
- subiecte de autoevaluare;
- bibliografie.

Toate exemplele expuse în lucrare sunt rulate pe mediul de dezvoltare integrat Dev-C++, dar pot fi rulate și pe un calculator cu compilatorul CodeBlocks sau VisualC++.

Materialul este expus astfel, încât să fie înțeles și însușit în mod succesiv. Se consideră că studentul va executa fiecare exemplu la calculator, va răspunde la întrebări și va efectua lucrul asupra exercițiilor de evaluare propuse.

Fiecărui student i se vor propune o serie de probleme pentru lucrul individual cu referire la capitolele lucrării (vezi anexa 3). Conținutul lucrărilor de laborator exemplifică algoritmi de soluționare a problemelor.

Pentru executarea lucrului, individual se vor respecta următoarele cerințe:

1) profesorul formulează problema pentru fiecare capitol împreună cu studentul;

2) studentul analizează inițial problema, elaborează programul, apoi o salvează pe calculator;

3) susținerea fiecărei probleme se face în mod practic prin prezentarea lucrării și testarea asistată de calculator;

4) la finele cursului (semestrului) studentul va prezenta raportul care va include setul de probleme elaborate individual pe parcursul semestrului.

Autoarea exprimă sincere mulțumiri dnei Lilia SAVA, conf. univ., pentru ajutorul acordat, dlui Sergiu TUTUNARU pentru atenția și sfaturile valoroase la elaborarea lucrării.

## **Bibliografie:**

1. Kris Jamsa & Lars Klander. *Totul despre C și C++*. București: Editura Teora, 2006.

2. Herbert Schildt. *C++ manual complet*. București: Editura Teora, 1999.

Disponibil: <https://drive.google.com/file/d/1BrQtITgykcWk03xxtl3Q0-nvcRRFAsy7/view?usp=sharing>

3. Tutorials C++ Language. Operators.

Disponibil: <https://m.cplusplus.com/doc/tutorial-ro/functions/>