



Digitally signed by
Library TUM
Reason: I attest to the
accuracy and integrity
of this document

UNIVERSITATEA TEHNICĂ A MOLDOVEI

PROGRAMAREA CALCULATOARELOR, STRUCTURI DE DATE ȘI ALGORITMI ÎN LIMBAJUL C

CULEGERE DE PROBLEME
*pentru executarea lucrărilor
de laborator și de control*



Chișinău
2019

UNIVERSITATEA TEHNICĂ A MOLDOVEI
FACULTATEA CALCULATOARE, INFORMATICĂ ȘI
MICROELECTRONICĂ
DEPARTAMENTUL INFORMATICĂ ȘI INGINERIA SISTEMELOR

PROGRAMAREA CALCULATOARELOR, STRUCTURI DE DATE ȘI ALGORITMI ÎN LIMBAJUL C

CULEGERE DE PROBLEME
*pentru executarea lucrărilor
de laborator și de control*

Chișinău
Editura „Tehnica-UTM”
2019

CZU 004.432.4(076.5)

K 91

Lucrarea conține sarcini pentru lucrări de laborator și de control, care sunt predestinate pentru studierea diverselor teme referitor la algoritimizare, structuri de date și programare în limbajul C: algoritmi ciclici și cu ramificări, tablouri unidimensionale și bidimensionale, pointeri și alocarea dinamică a memoriei, șiruri de caractere, fișiere, funcții, tablouri de structuri, algoritmi de căutare și sortare, implementarea tipurilor abstracte de date – liste înlănțuite și arbori. Pentru fiecare lucrare de laborator sunt indicate: tema, scopul lucrării, enunțul problemei și sunt prezentate 20 variante de sarcini.

Lucrarea este destinată studenților tuturor specialităților Facultății Calculatoare, Informatică și Microelectronică cu forma de studii la zi și cu frecvență redusă, care studiază disciplinele *Programarea calculatoarelor* și *Structuri de date și algoritmi*, precum și studenților altor facultăți ale UTM.

Autori: conf. univ., dr. Mihail Kulev
conf. univ., dr. Liudmila Luchianova
conf. univ., dr. Lucia Stadler

Redactor responsabil: conf. univ., dr. Mihail Kulev

Recenzent: conf. univ., dr. Vasile Moraru

**DESCRIEREA CIP A CAMEREI NAȚIONALE A CĂRȚII
Kulev, Mihail**

Programarea calculatoarelor, structuri de date și algoritmi în limbajul C: Culegere de probleme pentru executarea lucrărilor de laborator și de control / Mihail Kulev, Liudmila Luchianova, Lucia Stadler; Univ. Tehn. a Moldovei, Fac. Calculatoare, Informatică și Microelectronică, Dep. Informatică și Ingineria Sistemelor. – Chișinău: Tehnica-UTM, 2019. – 40 p. Bibliogr.: p. 38 (13 tit.). – 100 ex. ISBN 978-9975-45-595-4. 004.432.4(076.5) K 91

ISBN 978-9975-45-595-4.

© UTM, 2019

Lucrarea de laborator nr.1.

Tema: Utilizarea instrucțiunilor de control și ciclice în limbajul C

Scopul lucrării: Studierea tehnicilor și metodelor de utilizare a instrucțiunilor de control a condiției și a instrucțiunilor ciclice în limbajul C pentru tabularea funcției.

Sarcina (conform variantelor): Să se calculeze și să se afișeze la ecran valorile argumentului x și valorile funcției F , definită prin 3 expresii date, pentru intervalul $x1 \leq x \leq x2$ și pasul px de incrementare a argumentului x . Valorile $x1$, $x2$, px și parametrii a , b , c sunt date de intrare de tip real.

Varianta 1.

$$F = \begin{cases} \frac{c}{ax} - b & \text{pentru } x + a < 0 \text{ și } c = 0 \\ \frac{x - a}{c \sin x} & \text{pentru } x + a > 0 \text{ și } c \neq 0 \\ \frac{10x}{c - 6} & \text{în celelalte cazuri} \end{cases}$$

Varianta 2.

$$F = \begin{cases} a - \frac{\cos x}{10 + b} & \text{pentru } x < 0 \text{ și } b \neq 0 \\ \frac{x - a}{x - b} & \text{pentru } x > 0 \text{ sau } b = 0 \\ 3x^2 + \frac{a}{c} & \text{în celelalte cazuri} \end{cases}$$

Varianta 3.

$$F = \begin{cases} \frac{x + ac}{x + b} & \text{pentru } c < 0 \text{ și } b \neq 0 \\ ax^2 + b^2x + c & \text{pentru } c > 0 \text{ și } b = 0 \\ \frac{\sin x + a}{c} & \text{în celelalte cazuri} \end{cases}$$

Varianta 4.

$$F = \begin{cases} a \sin x - \frac{\ln x}{c+b} & \text{pentru } x < c \text{ și } a \neq 0 \\ \frac{x - ax^2}{x - b - \sin c} & \text{pentru } x > c \text{ și } a = 0 \\ 3x + \frac{a+x}{c^2} & \text{în celelalte cazuri} \end{cases}$$

Varianta 5.

$$F = \begin{cases} a \ln x - \frac{\cos x}{c} & \text{pentru } a + x < 0 \text{ sau } b \neq 0 \\ \sin\left(\frac{x-a}{x-b}\right) & \text{pentru } a + x > 0 \text{ și } b = 0 \\ e^x + \frac{a+x^2}{c} & \text{în celelalte cazuri} \end{cases}$$

Varianta 6.

$$F = \begin{cases} \frac{1+x}{x - \cos c} - \frac{b}{a} & \text{pentru } b - a < 0 \text{ și } c = 0 \\ \frac{a - bx}{\ln x} & \text{pentru } b - a > 0 \text{ și } c \neq 0 \\ \frac{10x+2}{c^2x - 6 - b} & \text{în celelalte cazuri} \end{cases}$$

Varianta 7.

$$F = \begin{cases} \frac{a + \ln x}{\sin c} - b^2 & \text{pentru } x - 2 > 0 \text{ și } a = 0 \\ \frac{x - \sin(x+1)}{b} & \text{pentru } x - 2 < 0 \text{ sau } a \neq 0 \\ \frac{ax+c}{\cos 2x} & \text{în celelalte cazuri} \end{cases}$$

Varianta 8.

$$F = \begin{cases} \frac{\ln x}{c+b} + \frac{x}{a} & \text{pentru } x < c+a \text{ sau } b \neq 0 \\ \frac{a - ax^2 + c^3}{\sin b} & \text{pentru } x > c+a \text{ și } b = 0 \\ \frac{a+x}{x^2 - c} & \text{în celelalte cazuri} \end{cases}$$

Varianta 9.

$$F = \begin{cases} \frac{a+b}{\ln(x-c)} & \text{pentru } x+c < 0 \text{ și } b = 0 \\ \frac{\sin a - x + 3}{c^2} & \text{pentru } x+c > 0 \text{ și } b \neq 0 \\ \frac{4x+8}{a^2x} + \sin x & \text{în celelalte cazuri} \end{cases}$$

Varianta 10.

$$F = \begin{cases} \frac{\cos x + 1}{b-x} & \text{pentru } x-10 < 0 \text{ și } a = 0 \\ \frac{ax^2 - bx + c}{\sin x} & \text{pentru } x-10 > 0 \text{ sau } a \neq 0 \\ \frac{\ln(x+c)}{2b} & \text{în celelalte cazuri} \end{cases}$$

Varianta 11.

$$F = \begin{cases} \frac{\sin a + bx}{|x^2 - c|} & \text{pentru } x+b < 0 \text{ și } c = 0 \\ \frac{\ln ax}{c^2 + b} & \text{pentru } x+b > 0 \text{ și } c \neq 0 \\ \frac{4\sqrt{x} + 8 \cos x}{a^2 + b} & \text{în celelalte cazuri} \end{cases}$$

Varianta 12.

$$F = \begin{cases} \frac{a \cos^2 x + c}{b^2 - x} & \text{pentru } c + b < 0 \text{ și } a \neq 0 \\ \frac{ax - \ln bx}{b^2 - cx} & \text{pentru } c + b > 0 \text{ și } a = 0 \\ \frac{3 \sin x + 4 \ln 2x}{a^2 + c + bx} & \text{în celelalte cazuri} \end{cases}$$

Varianta 13.

$$F = \begin{cases} c \ln ax - \frac{x^2}{a+c} & \text{pentru } x < b \text{ sau } c \neq 0 \\ \frac{x - \ln x^2}{b - \cos^2 ac} & \text{pentru } x > b \text{ și } c = 0 \\ \frac{a+x}{c^2} + e^{3x} & \text{în celelalte cazuri} \end{cases}$$

Varianta 14.

$$F = \begin{cases} \frac{|x+a|}{\ln x} + \frac{b}{c} & \text{pentru } x < a \text{ sau } b \neq 0 \\ \frac{\sin x + \cos b}{(a+c)^3} & \text{pentru } x > a \text{ și } b = 0 \\ \frac{a + \ln x}{x^2 - c^2} & \text{în celelalte cazuri} \end{cases}$$

Varianta 15.

$$F = \begin{cases} \frac{\sqrt{x+a+b^2}}{\cos x} & \text{pentru } x > c + b \text{ și } a \neq 0 \\ \frac{\ln a + \ln b}{ax + c^2} & \text{pentru } x < c + b \text{ și } a = 0 \\ \frac{a+b+x}{x^2} & \text{în celelalte cazuri} \end{cases}$$

Varianta 16.

$$F = \begin{cases} \frac{xc - b}{\cos x} & \text{pentru } x+b < 0 \text{ și } c \neq 0 \\ \frac{x - \ln b}{ax + c^2 + 2} & \text{pentru } x+b > 0 \text{ sau } c \neq 0 \\ \frac{a + \sin b + x}{x^2} & \text{în celelalte cazuri} \end{cases}$$

Varianta 17.

$$F = \begin{cases} \frac{\sin x + 1}{b - x - c} & \text{pentru } x < b \text{ sau } c \neq 0 \\ \frac{ax^2 - bxc}{\sin x} & \text{pentru } x > b \text{ și } c = 0 \\ \frac{\ln x + c}{2b} & \text{în celelalte cazuri} \end{cases}$$

Varianta 18.

$$F = \begin{cases} \frac{a + x}{x - \cos c} - \frac{b}{a} & \text{pentru } b - a < 0 \text{ și } c = 0 \\ \frac{a - bx}{x + c} & \text{pentru } b - a > 0 \text{ și } c \neq 0 \\ \frac{10 \ln x + 2}{c^2 x - 6 - b} & \text{în celelalte cazuri} \end{cases}$$

Varianta 19.

$$F = \begin{cases} \frac{a + \sin x}{\sin c} - b^2 & \text{pentru } x < a \text{ și } b \neq 0 \\ \frac{x - \cos(x+1)}{b} & \text{pentru } x > a \text{ și } b = 0 \\ \frac{ax + \ln c}{\cos 2x} & \text{în celelalte cazuri} \end{cases}$$

Varianta 20.

$$F = \begin{cases} \frac{x + \ln ac}{x + b} & \text{pentru } b - a < 0 \text{ și } c = 0 \\ ax^2 + b^2 \sin x + c & \text{pentru } b - a > 0 \text{ sau } c \neq 0 \\ \frac{\cos x + a}{c} & \text{în celelalte cazuri} \end{cases}$$

Exemplu de realizare:

$$F = \begin{cases} \frac{a^2 x - c}{b^2 + \cos x} & \text{pentru } c + b < 0 \text{ și } a \neq 0 \\ \frac{ax + \sin bx}{b^2 + cx} & \text{pentru } c + b > 0 \text{ și } a = 0 \\ \frac{3 \ln x + 2x}{a^2 + c - bx} & \text{în celelalte cazuri} \end{cases}$$

Descrierea variabilelor:

a) date de intrare (input data)

$x1, x2, px, a, b, c$ – variabile simple de tip real (float);

b) date de ieșire (output data)

x, F – variabile simple de tip real (float), rezultatele de calcul;

n – variabila simplă de tip întreg (int), numărul de ordine al rezultatelor de calcul;

c) date de lucru (working data), fiind în același timp și date de ieșire

x, F – variabile simple de tip real (float), rezultatele de calcul;

n – variabila simplă de tip întreg (int), numărul de ordine al rezultatelor de calcul.

Codul programului în limbajul C:

```
1  #include<stdio.h>
2  #include<conio.h>
3  #include<math.h>
4  int main()
5  {
6  float x1,x2,px,a,b,c;
7  float x,F;
8  int n;
9  // clrscr();
10 printf("Input x1 : ");
11 scanf("%f", &x1);
12 printf("Input x2 : ");
13 scanf("%f", &x2);
14 printf("Input px : ");
15 scanf("%f", &px);
16 printf("Input a : ");
17 scanf("%f", &a);
18 printf("Input b : ");
19 scanf("%f", &b);
20 printf("Input c : ");
21 scanf("%f", &c);
22 x = x1;
23 n = 0;
24 printf("\n\t Results : \n"); // puts("\n \t Results : ");
25 while (x < x2)
26 {
27 n = n + 1;
28 if((c + b < 0) && (a != 0))
29 {
30 F = (a*a*x - c)/(b*b +cos(x));
31 }
32 else if ((c + b > 0) && (a == 0))
33 {
34 F = (a*x + sin(b*x))/(b*b +c*x);
35 }
36 else
37 {
38 F = (3*log(x) + 2*x)/(a*a +c - b*x);
39 }
40 printf("%i: x = %.3f \t F = %.3f \n", n,x,F);
41 x = x + px;
42 }
43 getch();
44 return 0;
45 }
```

Rezultatele testării și funcționării programului:

Input x1 : 4
Input x2 : 8
Input px : 0.4
Input a : 5
Input b : 6
Input c : 7

Results :

1: x = 4.000	F = 1.520
2: x = 4.400	F = 2.365
3: x = 4.800	F = 4.471
4: x = 5.200	F = 19.182
5: x = 5.600	F = -10.230
6: x = 6.000	F = -4.344
7: x = 6.400	F = -2.870
8: x = 6.800	F = -2.199
9: x = 7.200	F = -1.814
10: x = 7.600	F = -1.565

Lucrarea de laborator nr. 2.

Tema: Prelucrarea tablourilor unidimensionale în limbajul C

Scopul lucrării: Studiarea posibilităților și mijloacelor limbajului C pentru programarea algoritmilor cu structură ramificată și ciclică la prelucrarea tablourilor unidimensionale.

Sarcină (conform variantelor):

Pentru tabloul unidimensional dat format din n elemente de tip real:

Varianta 1. Să se determine valoarea maximală dintre elementele tabloului și pozițiile elementelor cu această valoare, precum și suma tuturor elementelor negative în tablou.

Varianta 2. Să se determine valoarea minimală dintre elementele tabloului și pozițiile elementelor cu această valoare, precum și produsul tuturor elementelor pozitive în tablou.

Varianta 3. Să se determine valoarea maximală dintre elementele tabloului și pozițiile primului și ultimului element cu această valoare, precum și media aritmetică a tuturor elementelor pozitive în tablou.

Varianta 4. Să se determine valoarea minimală dintre elementele tabloului și pozițiile primului și ultimului element cu această valoare, precum și media aritmetică a tuturor elementelor negative în tablou.

Varianta 5. Să se determine valoarea maximală dintre elementele tabloului și numărul de elemente cu această valoare, precum și pozițiile acestora în tablou.

Varianta 6. Să se determine valoarea minimală dintre elementele tabloului și numărul de elemente cu această valoare, precum și pozițiile acestora în tablou.

Varianta 7. Să se determine valoarea maximală negativă dintre elementele tabloului și poziția primului element cu această valoare, precum și numărul elementelor pozitive în tablou.

Varianta 8. Să se determine valoarea minimală pozitivă dintre elementele tabloului și poziția ultimului element cu această valoare, precum și numărul elementelor negative în tablou.

Varianta 9. Să se determine valoarea maximală negativă dintre elementele tabloului și poziția ultimului element cu această valoare, precum și numărul elementelor negative în tablou.

Varianta 10. Să se determine valoarea minimală pozitivă dintre elementele tabloului și poziția primului element cu această valoare, precum și numărul elementelor pozitive în tablou.

Varianta 11. Să se determine valorile primului element minimal pozitiv și a ultimului element maximal negativ, precum și pozițiile acestora în tablou.

Varianta 12. Să se determine valorile ultimului element minimal pozitiv și a primului element maximal negativ și pozițiile acestora în tablou.

Varianta 13. Să se determine valoarea maximală dintre elementele tabloului și numărul de elemente cu această valoare, precum și media aritmetică a tuturor elementelor pozitive în tablou.

Varianta 14. Să se determine valoarea minimală dintre elementele tabloului și numărul de elemente cu această valoare, precum și media aritmetică a tuturor elementelor nenule în tablou.

Varianta 15. Să se determine valoarea maximală negativă dintre elementele tabloului și pozițiile elementelor cu această valoare în tablou.

Varianta 16. Să se determine valoarea minimală pozitivă dintre elementele tabloului și pozițiile elementelor cu această valoare în tablou.

Varianta 17. Să se determine valorile primului element minimal pozitiv și a ultimului element maximal negativ, precum și pozițiile acestora în tablou.

Varianta 18. Să se determine valorile ultimului element minimal negativ și a primului element maximal negativ, și pozițiile acestora în tablou.

Varianta 19. Să se determine numărul de elemente negative mai mici ca elementul maximal și produsul elementelor pozitive mai mari ca elementul minimal în tablou.

Varianta 20. Să se determine numărul de elemente negative ce se află la dreapta de elementul maximal și numărul de elemente pozitive ce se află la stânga de elementul minimal în tablou.

Lucrarea de laborator nr. 3.

Tema: Prelucrarea tablourilor bidimensionale în limbajul C

Scopul lucrării: Studiarea posibilităților și mijloacelor limbajului C pentru programarea algoritmilor de prelucrare a tablourilor bidimensionale.

Sarcină (conform variantelor):

Pentru tabloul bidimensional dat din n linii și m coloane:

Varianta 1. Să se determine produsul elementelor nenule din fiecare coloană și numărul elementelor negative de pe diagonala principală.

Varianta 2. Să se determine suma elementelor negative din fiecare linie și media elementului maximal și cel minimal.

Varianta 3. Să se ordoneze liniile tabloului în ordinea ascendentă a numărului de elemente egale în fiecare linie.

Varianta 4. Să se determine elementul minimal de pe diagonala secundară, suma și numărul elementelor pozitive din coloanele pare.

Varianta 5. Să se ordoneze liniile tabloului în ordine ascendentă a numărului de elemente negative în fiecare linie.

Varianta 6. Să se calculeze numărul de minimuri locale ale tabloului. Un element al tabloului este minim local, dacă este strict mai mic decât toți vecinii săi.

Varianta 7. Să se determine numărul de ordine al coloanei în care se află cea mai lungă serie de elemente identice.

Varianta 8. Să se determine minimul dintre sumele elementelor pozitive de pe diagonala principală și cea secundară.

Varianta 9. Să se ordoneze liniile tabloului în corespundere cu creșterea caracteristicii liniei. În calitate de caracteristică a liniei a se consideră suma elementelor pozitive pare ale acesteia.

Varianta 10. Să se calculeze numărul de maximuri locale ale tabloului. Un element al tabloului este maxim local, dacă este strict mai mare decât toți vecinii săi.

Varianta 11. Să se determine maximul dintre sumele elementelor negative de pe diagonala principală și cea secundară.

Varianta 12. Să se determine suma tuturor elementelor negative și media geometrică a elementelor pozitive din fiecare linie.

Varianta 13. Să se ordoneze coloanele tabloului în corespundere cu creșterea caracteristicii coloanei. În calitate de caracteristică a coloanei a se considera suma elementelor pozitive pare ale acesteia.

Varianta 14. Să se ordoneze liniile tabloului în ordinea ascendentă a numărului de elemente pozitive în fiecare linie.

Varianta 15. Să se efectueze uniformizarea tabloului. Operația de uniformizare a tabloului se efectuează prin înlocuirea fiecărui element cu media aritmetică a tuturor vecinilor.

Varianta 16. Să se ordoneze liniile tabloului în ordinea ascendentă a numărului de elemente pozitive în fiecare linie.

Varianta 17. Să se ordoneze coloanele tabloului în ordinea ascendentă a numărului de elemente negative în fiecare coloană.

Varianta 18. Să se determine suma elementelor pozitive din liniile pare și produsul elementelor mai mici ca 5 din coloanele impare.

Varianta 19. Să se determine suma elementelor negative din coloanele impare și produsul elementelor pozitive mai mici ca media aritmetică a elementelor situate pe diagonala principală.

Varianta 20. Să se determine produsul elementelor nenule din fiecare coloană și numărul elementelor pozitive de pe perimetrul tabloului.

Lucrarea de laborator nr. 4

Tema: Alocarea dinamică a memoriei pentru tablourile bidimensionale. Utilizarea funcțiilor și a pointerilor

Scopul lucrării: Programarea algoritmilor de prelucrare a tablourilor bidimensionale prin utilizarea funcțiilor, pointerilor și alocarea dinamică a memoriei pentru tablou.

Sarcină:

Pentru tabloul bidimensional dat din n linii și m coloane să se afișeze la ecran următorul meniu de opțiuni:

1. *Alocarea dinamică a memoriei pentru tablourile bidimensionale*
2. *Introducerea elementelor tabloului de la tastatură*
3. *Completarea tabloului cu valori aleatorii*
4. *Sortarea elementelor tabloului (conform variantelor)*
5. *Afișarea elementelor tabloului la ecran*
6. *Eliberarea memoriei alocate pentru tablou*
0. *Ieșire din program*

Să se elaboreze funcțiile pentru realizarea opțiunilor meniului.

Variante pentru sortare (pentru opțiunea 4 din meniu):

Varianta 1. Să se sorteze liniile tabloului în ordine ascendentă cu ajutorul metodei selecției liniare.

Varianta 2. Să se sorteze coloanele tabloului în ordine descendentă cu ajutorul metodei “bulelor”.

Varianta 3. Să se sorteze liniile tabloului în ordine descendentă cu ajutorul metodei de sortare prin selecție și schimb.

Varianta 4. Să se sorteze coloanele tabloului în ordine ascendentă cu ajutorul metodei selecției liniare.

Varianta 5. Să se sorteze liniile tabloului în ordine ascendentă cu ajutorul metodei de sortare prin inserție.

Varianta 6. Să se sorteze coloanele tabloului în ordine descendentă cu ajutorul metodei de sortare prin selecție și schimb.

Varianta 7. Să se sorteze liniile impare ale tabloului în ordine ascendentă cu ajutorul metodei selecției liniare.

Varianta 8. Să se sorteze coloanele pare ale tabloului în ordine descendentă cu ajutorul metodei “bulelor”.

Varianta 9. Să se sorteze liniile pare ale tabloului în ordine ascendentă cu ajutorul metodei de sortare prin selecție și schimb.

Varianta 10. Să se sorteze coloanele impare ale tabloului în ordine descendentă cu ajutorul metodei “bulelor”.

Varianta 11. Să se sorteze prima și ultima linie ale tabloului în ordine descendentă cu ajutorul metodei selecției liniare.

Varianta 12. Să se sorteze prima și ultima coloană ale tabloului în ordine ascendentă cu ajutorul metodei “bulelor”.

Varianta 13. Să se sorteze a doua și penultima linie ale tabloului în ordine descendentă cu ajutorul metodei de sortare prin selecție și schimb.

Varianta 14. Să se sorteze elementele tabloului pe linii de la stânga la dreapta și de sus în jos cu ajutorul metodei “bulelor”.

Varianta 15. Să se sorteze elementele tabloului pe coloane de sus în jos și de la stânga la dreapta cu ajutorul metodei selecției liniare.

Varianta 16. Să se sorteze elementele tabloului de pe perimetrul tabloului în ordine ascendentă folosind metoda de sortare prin inserție.

Varianta 17. Să se sorteze coloanele impare ale tabloului în ordine descendentă cu ajutorul metodei de sortare prin selecție liniară.

Varianta 18. Să se sorteze liniile impare ale tabloului în ordine ascendentă folosind metoda de sortare prin inserție.

Varianta 19. Să se sorteze elementele tabloului de pe diagonala secundară a tabloului în ordine descendentă folosind metoda “bulelor”.

Varianta 20. Să se sorteze prima și ultima linie ale tabloului în ordine ascendentă cu ajutorul metodei de sortare prin inserție.

Lucrarea de laborator nr. 5

Tema: Prelucrarea caracterelor și a șirurilor de caractere

Scopul lucrării: Programarea algoritmilor de prelucrare a caracterelor și a șirurilor de caractere prin utilizarea funcțiilor predefinite de prelucrare a caracterelor și a șirurilor de caractere în limbajul C.

Sarcină (conform variantelor):

Pentru propoziția dată:

- Varianta 1.* Să se determine numărul de vocale și frecvența repetării fiecăreia dintre ele.
- Varianta 2.* Să se determine cel mai “mare” cuvânt în sens lexicografic.
- Varianta 3.* Să se determine cel mai lung cuvânt.
- Varianta 4.* Să se determine cel „mai mic” cuvânt în sens lexicografic.
- Varianta 5.* Să se determine cel mai scurt cuvânt.
- Varianta 6.* Să se sorteze cuvintele în sens lexicografic.
- Varianta 7.* Să se sorteze cuvintele în ordinea descreșterii lungimii acestora.
- Varianta 8.* Să se determine frecvența repetării fiecărei litere.
- Varianta 9.* Să se determine lungimea celui mai mic interval dintre cuvinte.
- Varianta 10.* Să se determine numărul cuvintelor care încep cu simbolul dat.
- Varianta 11.* Să se determine cuvintele simetrice.
- Varianta 12.* Să se determine numărul de consoane și frecvența repetării fiecăreia dintre acestea.
- Varianta 13.* Să se sorteze cuvintele în ordinea inversă celei alfabetice.

- Varianta 14.** Să se sorteze cuvintele în ordinea creșterii lungimii acestora.
- Varianta 15.** Să se elimine din propoziție cuvintele identice.
- Varianta 16.** Să se determine corectitudinea parantezelor deschise și închise.
- Varianta 17.** Să se determine numărul de cuvinte.
- Varianta 18.** Să se determine prima și ultima literă din fiecare cuvânt.
- Varianta 19.** Să se elimine prima și ultima literă din fiecare cuvânt.
- Varianta 20.** Să se determine lungimea propoziției.

Lucrarea de laborator nr. 6

Tema: Prelucrarea tabloului de structuri și utilizarea fișierelor în limbajul C

Scopul lucrării: Programarea algoritmilor pentru prelucrarea tabloului de structuri prin utilizarea funcțiilor, pointerilor, alocării dinamice a memoriei și a fișierelor în limbajul C.

Sarcină:

Pentru tabloul unidimensional dat cu elemente de tip structură (conform variantelor) să se afișeze la ecran următorul meniu de opțiuni:

1. *Alocarea dinamică a memoriei pentru tabloul de structuri.*
2. *Introducerea elementelor tabloului de la tastatură.*
3. *Căutarea elementului tabloului.*
4. *Sortarea tabloului.*
5. *Redactarea elementului tabloului.*
6. *Adăugarea unui element nou la sfârșit.*
7. *Eliminarea elementului indicat din tablou.*
8. *Inserarea unui element nou.*
9. *Salvarea elementelor tabloului în fișier.*
10. *Citirea elementelor tabloului din fișier.*
11. *Extragerea elementelor tabloului la ecran.*
12. *Eliberarea memoriei alocate pentru tablou.*
0. *Ieșire din program.*

Să se elaboreze funcțiile pentru implementarea opțiunilor meniului.

Variante pentru structuri:

1. Structura *Carte* cu câmpurile: autorul, denumirea, anul ediției, numărul de pagini, tirajul.

2. Structura *Film* cu câmpurile: denumirea, țara, regizorul, genul filmului, anul.
3. Structura *Hotel* cu câmpurile: denumirea, adresa, telefonul, categoria, numărul de locuri.
4. Structura *Cinematograf* cu câmpurile: denumirea, adresa, telefonul, numărul de locuri, prețul biletelor.
5. Structura *Marfă* cu câmpurile: denumirea, țara, producătorul, articolul, prețul.
6. Structura *Automobil* cu câmpurile: modelul, țara, data fabricării, puterea motorului, costul.
7. Structura *Disciplină* cu câmpurile: denumirea, anul, facultatea, departamentul, numărul de ore.
8. Structura *Muzeu* cu câmpurile: denumirea, adresa, telefonul, numărul de exponate, prețul biletului.
9. Structura *Casă* cu câmpurile: adresa, telefonul, suprafața, numărul de odăi, costul.
10. Structura *Stat* cu câmpurile: denumirea, capitala, suprafața, populația, produsul intern brut.
11. Structura *Aviarută* cu câmpurile: numărul, destinație, ora plecării, ora sosirii, prețul biletului.
12. Structura *Farmacie* cu câmpurile: denumirea, adresa, telefonul, orele de lucru, volumul de medicamente.
13. Structura *Restaurant* cu câmpurile: denumirea, adresa, telefonul, categoria, numărul de locuri, specializarea.
14. Structura *Universitate* cu câmpurile: denumirea, adresa, telefonul, numărul de facultăți, numărul de studenți.

15. Structura *Computer* cu câmpurile: modelul, procesorul, memoria, viteza, prețul.
16. Structura *Canal TV* cu câmpurile: denumirea, țara, telefonul, genul, raitigul.
17. Structura *Biblioteca* cu câmpurile: denumirea, adresa, telefonul, numărul de săli, numărul de cărți.
18. Structura *Imobil* cu câmpurile: proprietarul, tipul, adresa, suprafața, costul.
19. Structura *Capitală* cu câmpurile: denumirea, țară, populația, suprafața, primarul.
20. Structura *Supermarket* cu câmpurile: denumirea, adresa, telefonul, orele de lucru, volumul vânzărilor.

Lucrarea de laborator nr. 7

Tema: Implementarea tipului abstract de date „Tablou de structuri” în limbajul C

Scopul lucrării: Obținerea deprinderilor practice de implementare și de utilizare a tipului abstract de date (TAD) “Tablou de structuri” în limbajul C.

Sarcină: Să se scrie trei fișiere-text în limbajul C pentru implementarea și utilizarea TAD “Tablou de structuri”:

1. Fișier antet cu extensia **.h**, care conține specificarea structurii de date a elementului tabloului (conform variantelor) și prototipurile funcțiilor de prelucrare a tabloului de structuri.
2. Fișier cu extensia **.cpp** sau **.c**, care conține implementările (codurile) funcțiilor, declarate în fișierul antet.
3. Fișier al utilizatorului, funcția **main()** pentru prelucrarea tabloului de structuri cu afișarea la ecran a următorului meniu de opțiuni:
 1. *Alocarea dinamică a memoriei pentru tablou.*
 2. *Introducerea informației despre elementele tabloului de la tastatură.*
 3. *Afișarea informației despre elementele tabloului la ecran.*
 4. *Căutarea elementului în tablou.*
 5. *Modificarea câmpurilor a unui element din tablou.*
 6. *Înterschimbarea a două elemente indicate din tablou.*
 7. *Sortarea tabloului.*
 8. *Adăugarea unui element nou la sfârșitul tabloului.*
 9. *Adăugarea unui element nou la începutul tabloului.*

10. *Inserarea unui element nou după elementul indicat al tabloului.*
11. *Inserarea unui element nou înainte elementul indicat al tabloului.*
12. *Ștergerea elementului indicat al tabloului.*
13. *Salvarea informației despre elementele tabloului în fișier.*
14. *Citirea informației despre elementele tabloului din fișier.*
15. *Eliberarea memoriei alocate pentru tablou.*
0. *Ieșire din program.*

Variante pentru structuri :

A se vedea **Variante pentru structuri** din lucrarea de laborator nr.6 (p. 22-24).

Lucrarea de laborator nr. 8

Tema: Implementarea tipului abstract de date „Listă unidirecțională” („Listă simplu înlănțuită”) în limbajul C. Partea I.

Scopul lucrării: Obținerea deprinderilor practice de implementare și de utilizare a tipului abstract de date (TAD) “Listă simplu înlănțuită” în limbajul C cu asigurarea operațiilor de prelucrare de bază ale listei.

Sarcină: Să se scrie trei fișiere-text în limbajul C pentru implementarea și utilizarea TAD “Listă simplu înlănțuită” cu asigurarea operațiilor de prelucrare de bază ale listei:

1. Fișier antet cu extensia **.h**, care conține specificarea structurii de date a elementului listei simplu înlănțuite (conform variantelor) și prototipurile funcțiilor de prelucrare de bază ale listei.
2. Fișier cu extensia **.cpp** sau **.c**, care conține implementările (codurile) funcțiilor declarate în fișierul antet.
3. Fișier al utilizatorului, funcția **main()** pentru prelucrarea listei cu afișarea la ecran a următorului meniu de opțiuni de bază:
 1. *Crearea listei în memoria dinamică.*
 2. *Introducerea informației despre elementele listei de la tastatură.*
 3. *Afișarea informației despre elementele listei la ecran.*
 4. *Căutarea elementului în listă.*
 5. *Modificarea câmpurilor unui element din listă.*
 6. *Determinarea adresei ultimului element din listă.*
 7. *Determinarea lungimii listei (numărul de elemente).*
 8. *Înterschimbarea a două elemente indicate în listă.*

9. *Sortarea listei.*
10. *Eliberarea memoriei alocate pentru listă.*
0. *Ieșire din program.*

Indicație: *La realizarea operației de afișare la ecran a informației referitor la elementele listei a se prevedea afișarea adreselor din memorie ale elementelor curent și următor din listă.*

Variante pentru structuri:

A se vedea **Variante pentru structuri** din lucrarea de laborator nr.6 (p. 22-24).

Lucrarea de laborator nr. 9

Tema: Implementarea tipului abstract de date „Listă unidirecțională” („Listă simplu înlănțuită”) în limbajul C.

Partea II

Scopul lucrării: Obținerea deprinderilor practice de implementare și de utilizare a tipului abstract de date (TAD) “Listă simplu înlănțuită” în limbajul C cu asigurarea operațiilor de prelucrare ale listei.

Sarcină: Să se scrie trei fișiere-text în limbajul C pentru implementarea și utilizarea TAD “Listă simplu înlănțuită” cu asigurarea operațiilor de prelucrare ale listei:

4. Fișier antet cu extensia **.h**, care conține specificarea structurii de date a elementului listei simplu înlănțuite (conform variantelor) și prototipurile funcțiilor de prelucrare a listei.
5. Fișier cu extensia **.cpp** sau **.c**, care conține implementările funcțiilor (codurile funcțiilor) declarate în fișierul antet.
6. Fișier al utilizatorului, funcția **main()** pentru prelucrarea listei cu afișarea la ecran a următorului meniu de opțiuni:
 1. *Crearea listei în memoria dinamică.*
 2. *Introducerea informației despre elementele listei de la tastatură.*
 3. *Afișarea informației despre elementele listei la ecran.*
 4. *Căutarea elementului în listă.*
 5. *Modificarea câmpurilor unui element din listă.*
 6. *Determinarea adresei ultimului element din listă.*
 7. *Determinarea lungimii listei (numărul de elemente).*
 8. *Înterschimbarea a două elemente indicate din listă.*

9. *Sortarea listei.*
10. *Adăugarea unui element nou la sfârșitul listei.*
11. *Adăugarea unui element nou la începutul listei.*
12. *Inserarea unui element nou după elementul indicat al listei.i*
13. *Inserarea unui element nou înaintea elementului indicat în listă.*
14. *Ștergerea elementului indicat în listă.*
15. *Divizarea listei în două liste.*
16. *Concatenarea a două liste.*
17. *Salvarea informației despre elementele listei în fișier.*
18. *Citirea informației despre elementele listei din fișier.*
19. *Eliberarea memoriei alocate pentru listă.*
0. *Ieșirea din program.*

Indicație: *La realizarea operației de afișare la ecran a informației referitor la elementele listei a se prevedea afișarea adreselor din memorie ale elementelor curent și următor din listă.*

Variante pentru structuri:

A se vedea **Variante pentru structuri** din lucrarea de laborator nr.6 (p. 22-24).

Lucrarea de laborator nr. 10

Tema: Implementarea tipului abstract de date „Arbore binar”.
Utilizarea algoritmilor iterativi

Scopul lucrării: Obținerea deprinderilor practice de implementare și de utilizare a tipului abstract de date (TAD) „Arbore binar” cu asigurarea operațiilor de prelucrare de bază ale **arborelui binar oarecare** prin parcurgerea nodurilor arborelui “în lățime” și “în adâncime”, folosind structurile respective de date “coadă” și “stivă”.

Sarcină: Să se scrie trei fișiere-text în limbajul C pentru implementarea și utilizarea TAD „Arbore binar” cu asigurarea operațiilor de prelucrare de bază ale **arborelui binar oarecare** prin parcurgerea nodurilor arborelui cu ajutorul algoritmilor iterativi:

1. Fișier antet cu extensia **.h**, care conține specificarea structurii de date a nodului arborelui binar (conform variantelor) și a structurilor de date ale elementelor cozii și stivei, precum și prototipurile funcțiilor de prelucrare ale arborelui binar oarecare, bazate pe algoritmi iterativi.
2. Fișier cu extensia **.cpp** sau **.c**, care conține implementările funcțiilor (codurile funcțiilor), declarate în fișierul antet.
3. Fișier al utilizatorului, funcția **main()** pentru prelucrarea arborelui binar oarecare cu afișarea la ecran a următorului meniu de opțiuni:
 1. *Crearea nodurilor arborelui binar oarecare în memoria dinamică și introducerea informației despre nodurile arborelui de la tastatură în mod interactive.*
 2. *Afișarea informației despre nodurile arborelui la ecran.*

3. Căutarea nodului în arbore.
4. Modificarea informației unui nod din arbore.
5. Determinarea numărului de noduri.
6. Determinarea înălțimii arborelui.
7. Eliberarea memoriei alocate pentru arbore.
0. Ieșirea din program.

Indicații:

1. La realizarea operației de afișare la ecran a informației referitor la nodurile arborelui a se prevedea și afișarea adreselor din memorie ale nodului curent precum și fiilor stâng și drept ai nodului curent.
2. Funcțiile de prelucrare ale arborelui binar oarecare trebuie implementate folosind **algoritmi iterativi** în 2 versiuni:
 - a) **utilizând coada** - pentru parcurgerea arborelui “în lățime” (parcurgerea arborelui pe niveluri);
 - b) **utilizând stiva** – pentru parcurgerea arborelui “în adâncime” (parcurgerea “în inordine”: rădăcină -> subarbore stâng(drept) -> subarbore drept(stâng). O excepție face funcția de eliberare a memoriei dinamice alocate pentru arbore (trebuie folosită parcurgerea “în postordine”: subarbore stâng(drept) -> subarbore drept(stâng) -> rădăcină).

Variante pentru structuri:

A se vedea **Variante pentru structuri** din lucrarea de laborator nr.6 (p. 22-24).

Lucrarea de laborator nr.11

Tema: Implementarea tipului abstract de date „Arbore binar”.
Utilizarea algoritmilor recursivi

Scopul lucrării: Obținerea deprinderilor practice de implementare și de utilizare a tipului abstract de date (TAD) „Arbore binar” cu asigurarea operațiilor de prelucrare de bază ale **arborelui binar oarecare** prin parcurgerea recursivă a nodurilor arborelui, folosind algoritmi recursivi.

Sarcină: Să se scrie trei fișiere-text în limbajul C pentru implementarea și utilizarea TAD „Arbore binar” cu asigurarea operațiilor de prelucrare de bază ale **arborelui binar oarecare** prin parcurgerea nodurilor arborelui cu ajutorul algoritmilor recursivi:

1. Fișier antet cu extensia **.h**, care conține specificarea structurii de date a nodului arborelui binar (conform variantelor) și prototipurile funcțiilor de prelucrare ale arborelui binar oarecare, bazate pe algoritmi recursivi.
2. Fișier cu extensia **.cpp** sau **.c**, care conține implementările funcțiilor (codurile funcțiilor), declarate în fișierul antet.
3. Fișier al utilizatorului, funcția **main()** pentru prelucrarea arborelui binar oarecare cu afișarea la ecran a următorului meniu de opțiuni:
 1. *Crearea nodurilor arborelui binar oarecare în memoria dinamică și introducerea informației despre nodurile arborelui de la tastatură în mod interactive.*
 2. *Afișarea informației despre nodurile arborelui la ecran.*
 3. *Căutarea nodului în arbore.*
 4. *Modificarea informației a unui nod din arbore.*

5. *Determinarea numărului de noduri.*
6. *Determinarea înălțimii arborelui.*
7. *Eliberarea memoriei alocate pentru arbore.*
0. *Ieșirea din program.*

Indicații:

1. *La realizarea operației de afișare la ecran a informației referitor la nodurile arborelui a se prevedea și afișarea adreselor din memorie ale nodului curent, precum și a fiilor stâng și drept ai nodului curent.*
2. *Funcțiile de prelucrare ale arborelui binar trebuie implementate folosind **algoritmi recursivi** pentru parcurgerea arborelui “în adâncime”: rădăcină -> subarbore stâng(drept) -> subarbore drept(stâng). O excepție face funcția de eliberare a memoriei dinamice alocate pentru arbore (trebuie folosită parcurgerea “în postordine”: subarbore stâng(drept) -> subarbore drept(stâng) -> rădăcină).*
3. *Funcția de afișare a informației despre nodurile arborelui la ecran trebuie implementată pentru toate parcurgerile recursive posibile ale arborelui binar*

Variante pentru structuri:

A se vedea **Variante pentru structuri** din lucrarea de laborator nr.6 (p. 22-24).

Lucrarea de laborator nr.12

Tema: Analiza empirică a algoritmilor de sortare și de căutare

Scopul lucrării: Studiarea posibilităților și mijloacelor limbajului C de programare a algoritmilor de sortare și de căutare pentru tablouri unidimensionale și obținerea deprinderilor de analiză empirică a algoritmilor.

Sarcină: Să se scrie un program în limbajul C pentru analiza empirică a algoritmului propus (după variantă) cu crearea funcției de căutare sau sortare pentru tabloul unidimensional de n elemente cu afișarea la ecran a următorului meniu de opțiuni:

1. Tabloul demonstrativ de n elemente ($5 \leq n \leq 20$).
2. Tabloul cu valori aleatorii ($n=10000$, $n=100000$, $n=1000000$).
3. Tabloul sortat crescător ($n=10000$, $n=100000$, $n=1000000$).
4. Tabloul sortat descrescător ($n=10000$, $n=100000$, $n=1000000$).
5. Analiza empirică a funcției de sortare **qsort()** sau a funcției de căutare **bsearch()** din biblioteca standard a limbajului C.
0. Ieșire din program.

Analiza empirică a algoritmului constă în: a) determinarea timpului de rulare, numărului de comparații, numărului de schimbări (sau deplasări) de elemente pentru tablouri cu 3 diferite seturi de valori și cu 3 diferite volume de elemente; b) compararea și analiza rezultatelor obținute utilizând funcția creată și funcția din biblioteca standard a limbajului C; c) tragerea concluziilor.

Variante de algoritmi:

1. Căutare secvențială.
2. Căutare binară cu algoritmul iterativ.
3. Sortare bulelor în ordine ascendentă.
4. Sortare prin selecție liniară în ordine descendentă.

5. Sortare prin selecție și schimb în ordine ascendentă.
6. Sortare prin inserție în ordine descendentă.
7. Sortare shaker în ordine ascendentă.
8. Căutare binară cu algoritmul recursiv.
9. Sortare Shell în ordine ascendentă.
10. Sortare rapidă în ordine descendentă.
11. Sortare prin interclasare (merge sort).
12. Sortare qsort ().
13. Căutare bsearch ().
14. Sortare prin coada de prioritate.
15. Sortare heap în ordine descendentă.
16. Sortare utilizând arbore binar de căutare.
17. Sortare prin interclasare (merge sort) în ordine ascendentă.
18. Sortare Shell în ordine descendentă.
19. Sortare prin inserție în ordine ascendentă.
20. Sortare heap în ordine descendentă.

Cerințe față de întocmirea raportului asupra lucrării de laborator

Raportul asupra fiecărei lucrări de laborator include:

1. Foaia de titlu

2. Conținutul raportului:

- Numărul de ordine al lucrării de laborator
- Tema
- Scopul lucrării
- Sarcina
- Numărul și datele variantei
- Mersul lucrării:
 - o succintă și scurtă descriere a noțiunilor teoretice și metodelor folosite;
 - descrierea variabilelor de intrare, de ieșire și de lucru;
 - descrierea algoritmului (schema-bloc a algoritmului);
 - codul programului în limbajul C;
 - rezultatele testării și funcționării programului;
 - verificarea și analiza rezultatelor .
- Concluzii
- Bibliografie

BIBLIOGRAFIE

1. **Logofătu Doina.** Bazele programării în C. Aplicații. Iași, Polirom, 2006.
2. **Jamsa Kris, Klander Lurs** Totul despre C și C++. Manual fundamental de programare în C și C++. București: Teora, 2004.
3. **Heileman Gregory L.** Data Structures, Algorithms and Object Oriented Programming. McGraw – Hill, 1996.
4. **Trembley Jean-Paul, Sorensen Paul G.** An introduction to Data Structures with Applications. McGraw – Hill, 1984.
5. **Knuth D.E.** Tratat de programarea calculatoarelor. Algoritmi fundamentali. București, Ed. Tehnica, 1974.
6. **Craus, Mitica.** Structuri de date și algoritmi. Iași, Editura "Gh. Asachi", 2002.
7. **Cristian Razvan Uscatu si alți.** Algoritmi și tehnici de programare: aplicații. București, Editura ASE, 2015.
8. **Ion Gh. Rosca și alți.** Bazele programării calculatoarelor: Teorie și aplicații în C. București, Editura ASE, 2006.
9. **Buzurniuc, Stefan.** Inițiere în limbajul C. Chișinău, Evrica, 2004.
10. **Buzurniuc, Stefan.** Programarea în limbajul C. Chișinău, ASEM, 2008.
11. Programarea Calculatoarelor – andrei.clubcisco.ro/Universitatea Politehnica din București, 2016. <http://andrei.clubcisco.ro/cursuri/anul-1/semestrul-1/programarea-calculatoarelor.html>
12. **Constantinescu, Nicolae.** Bazele programării procedurale: Limbajul C. – Craiova, Universitaria, 2012.
13. **Istrati, Sergiu G.** Programare: Inițializare în limbaje C și C++. – Chișinău, 2003. <http://library.utm.md>

CUPRINS

Lucrarea de laborator nr. 1.	
Utilizarea instrucțiunilor condiționale și ciclice în limbajul C.....	3
Lucrarea de laborator nr. 2.	
Prelucrarea tablourilor unidimensionale în limbajul C.....	11
Lucrarea de laborator nr. 3.	
Prelucrarea tablourilor bidimensionale în limbajul C.....	14
Lucrarea de laborator nr. 4.	
Alocarea dinamică a memoriei pentru tablourile bidimensionale. Utilizarea funcțiilor și a pointerilor.....	17
Lucrarea de laborator nr. 5.	
Prelucrarea caracterelor și a șirurilor de caractere în limbajul C.....	20
Lucrarea de laborator nr. 6.	
Prelucrarea tabloului de structuri și utilizarea fișierelor în limbajul C....	22
Lucrarea de laborator nr. 7.	
Implementarea tipului abstract de date „Tablou de structuri” în limbajul C	25
Lucrarea de laborator nr. 8.	
Implementarea tipului abstract de date „Listă unidirecțională” („Listă simplu înlănțuită”) în limbajul C. Partea I	27
Lucrarea de laborator nr. 9.	
Implementarea tipului abstract de date „Listă unidirecțională” („Listă simplu înlănțuită”) în limbajul C. Partea II.....	29
Lucrarea de laborator nr. 10.	
Implementarea tipului abstract de date „Arbore binar”. Utilizarea algoritmilor iterativi	31
Lucrarea de laborator nr. 11.	
Implementarea tipului abstract de date „Arbore binar”. Utilizarea algoritmilor recursivi.....	33
Lucrarea de laborator nr. 12.	
Analiza emperică a algoritmilor de sortare și de căutare.....	35
Cerințe față de întocmirea raportului asupra lucrării de laborator.....	37
BIBLIOGRAFIE	38

PROGRAMAREA CALCULATOARELOR, STRUCTURI DE DATE ȘI ALGORITMI ÎN LIMBAJUL C

CULEGERE DE PROBLEME *pentru executarea lucrărilor de laborator și de control*

Autori: Mihail Kulev
Liudmila Luchianova
Lucia Stadler

E. Gheorghîșteanu

Bun de tipar 29.08.19	Formatul hârtiei 60x84 1/16
Hârtie ofset. Tipar RISO	Tirajul 100 ex.
Coli de tipar 2,5	Comanda nr. 71

UTM, 2004, Chișinău, bd. Ștefan cel Mare și Sfânt, 168
Editura "Tehnica-UTM"
2045, Chișinău, str. Studenților, 9/9