

Universitatea Tehnică a Moldovei
Facultatea *Calculatoare, Informatică și Microelectronică*
Specialitatea *Tehnologii Informaționale*



Raport

la lucrarea de laborator nr. 2

Tema: “*Programarea algoritmilor cu structură ramificată și ciclică*”

Disciplina: “Programarea Calculatorului”

Varianta 4

A efectuat:

Student grupa TI-231 FR

Apareci Aurica

A verificat:

Asistent universitar

Mantaluță Marius

Chișinău 2023

Cuprins

1. Cadrul teoretic.....	3
2. Schema bloc.....	3
3. Listingul programului.....	4
4. Testarea aplicației.....	5
5. Concluzii.....	5

1. Cadrul teoretic

Tema: Programarea algoritmilor cu structură ramificată și ciclică

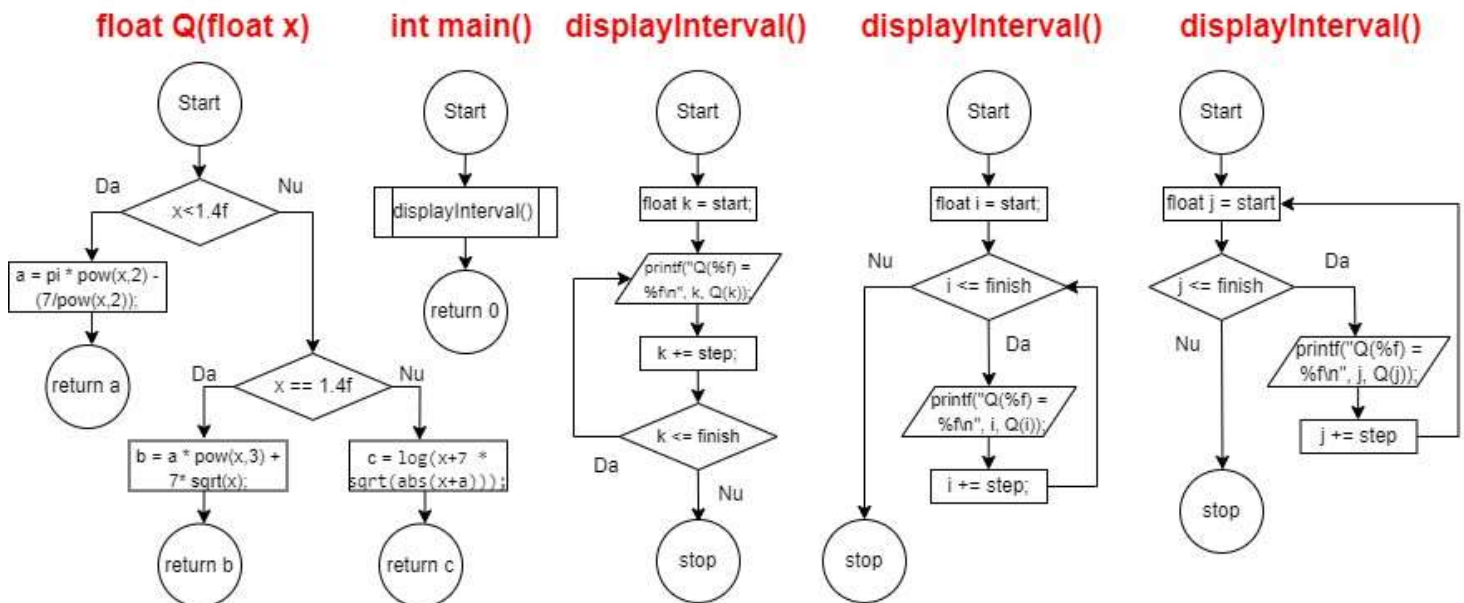
Scopul: Însușirea, folosirea și obținerea deprinderii practice de elaborare și depanare a programelor ciclice (if ... go to, for, while, do-while).

Sarcina: 1a. Să se elaboreze schema bloc și programul pentru calcularea valorii funcției conform condițiilor indicate în **Tabelul 1 (col. 1-4)**, prin introducerea valorii argumentului de la tastatură și utilizarea instrucțiunii if/else.

1b. Să se elaboreze schemele bloc și programele pentru calcularea valorilor funcției conform condițiilor indicate în **Tabelul 1 (col. 1-5)**, prin introducerea valorilor (inițială, finală și a pasului pentru modificare) ale argumentului de la tastatură și utilizarea instrucțiunilor: for, while, do while.

$Q = \begin{cases} \pi x^2 - \frac{7}{x^2} & x < 1.4 \\ ax^3 + 7\sqrt{x} & x = 1.4 \\ \ln(x + 7\sqrt{ x+a }) & x > 1.4 \end{cases}$	$\begin{matrix} x < 1.4 \\ x = 1.4 \\ x > 1.4 \end{matrix}$	$a = 1.65$	$\begin{matrix} x \in [0.7; 2] \\ \Delta x = 0.1 \end{matrix}$
---	---	------------	--

2. Schema bloc



main() – metoda principală a aplicației (punctul de execuție);

Q(float x) – metoda care calculează valoarea funcției în punctul X;

displayInterval() – metoda care afișează valorile funcției pe întreg intervalul $[0.7; 2]$;

3. Listingul programului

```
#include<stdio.h>
#include <stdlib.h>
#include <math.h>

float pi = 3.141592f;
float a = 1.65f;

float Q(float x)
{
    if(x<1.4f)
    {
        return pi * pow(x,2) - (7/pow(x,2));
    }
    else if(x == 1.4f)
    {
        return a * pow(x,3) + 7* sqrt(x);
    }
    return log(x + 7 * sqrt(abs(x+a)));
}

void displayInterval(float start, float finish, float step)
{
    //display interval using for
    for (float j = start; j <= finish; j += step)
    {
        printf("Q(%0.2f) = %0.4f\n", j, Q(j));
    }

    //display interval using while
    float i = start;
    while(i <= finish)
    {
        printf("Q(%0.2f) = %0.4f\n", i, Q(i));
        i += step;
    }

    //display interval using do while
    float k = start;
    do
    {
        printf("Q(%0.2f) = %0.4f\n", k, Q(k));
        k += step;
    } while (k <= finish);
}

int main()
{
    displayInterval(0.7f, 2.0f, 0.1f);
    return 0;
}
```

4. Testarea aplicației

Aplicația efectuează o serie de calcule pentru funcția $Q(x)$ folosind diferite bucle (for, while și do-while) pentru a afișa rezultatele într-un anumit interval de valori. În cadrul funcției displayInterval, intervalul specificat este de la 0.7 la 2.0, cu un pas de 0.1 între fiecare valoare.

Pentru $x < 1.4$:

Funcția $Q(x)$ calculează valoarea ca fiind $\pi \cdot x^2 - (7 / x^2)$.

Pentru $x = 1.4$:

Funcția $Q(x)$ calculează valoarea ca fiind $a \cdot x^3 + 7 \cdot \sqrt{x}$.

Pentru $x > 1.4$:

Funcția $Q(x)$ calculează valoarea ca fiind $\log(x + 7 \cdot \sqrt{\text{abs}(x+a)})$

```
Q(0.70) = -12.7463
Q(0.80) = -8.9269
Q(0.90) = -6.0973
Q(1.00) = -3.8584
Q(1.10) = -1.9838
Q(1.20) = -0.3372
Q(1.30) = 1.1673
Q(1.40) = 2.6045
Q(1.50) = 2.6119
Q(1.60) = 2.6192
Q(1.70) = 2.6264
Q(1.80) = 2.6336
Q(1.90) = 2.6408
Q(2.00) = 2.6479
```

5. Concluzii

În cadrul acestei lucrări de laborator, am explorat conceptele esențiale legate de programarea algoritmilor cu structură ramificată și ciclică. Am luat în considerare problema specifică de calcul a funcției $Q(x)$, care necesită diferite calcule în funcție de intervalul în care se află valoarea lui x . Am implementat soluția acestei probleme folosind limbajul de programare C și am demonstrat aplicarea corectă a structurilor ramificate (if-else) și ciclice (for, while și do-while) pentru a calcula și afișa rezultatele funcției $Q(x)$ pentru un interval specificat.

Prin această lucrare, am învățat cum să gestionăm situații în care algoritmi noștri trebuie să se comporte diferit în funcție de condițiile date. Am observat că structurile ramificate sunt esențiale pentru luarea deciziilor în programare, iar buclele ciclice sunt instrumente puternice pentru repetarea unor operații până când anumite condiții sunt îndeplinite. De asemenea, am dobândit abilități practice în programarea în limbajul C și în utilizarea funcțiilor matematice și de manipulare a datelor.