

**Міністерство освіти і науки України
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікорського»
Факультет інформатики та обчислювальної техніки
Кафедра обчислювальної техніки**

Лабораторна робота №1

з дисципліни
«Алгоритми і структури даних»

Виконав:

студент групи ІМ-43

Черепов Олександр Павлович

номер у списку групи: 32

Перевірила:

Молчанова А. А.

Київ 2024

Постановка задачі

Задано дійсне число x . Визначити значення заданої за варіантом кусочно-безперервної функції $y(x)$, якщо воно існує, або вивести на екран повідомлення про неіснування функції для заданого x .

Розв'язати задачу двома способами (створити дві блок-схеми алгоритмів та написати дві програми):

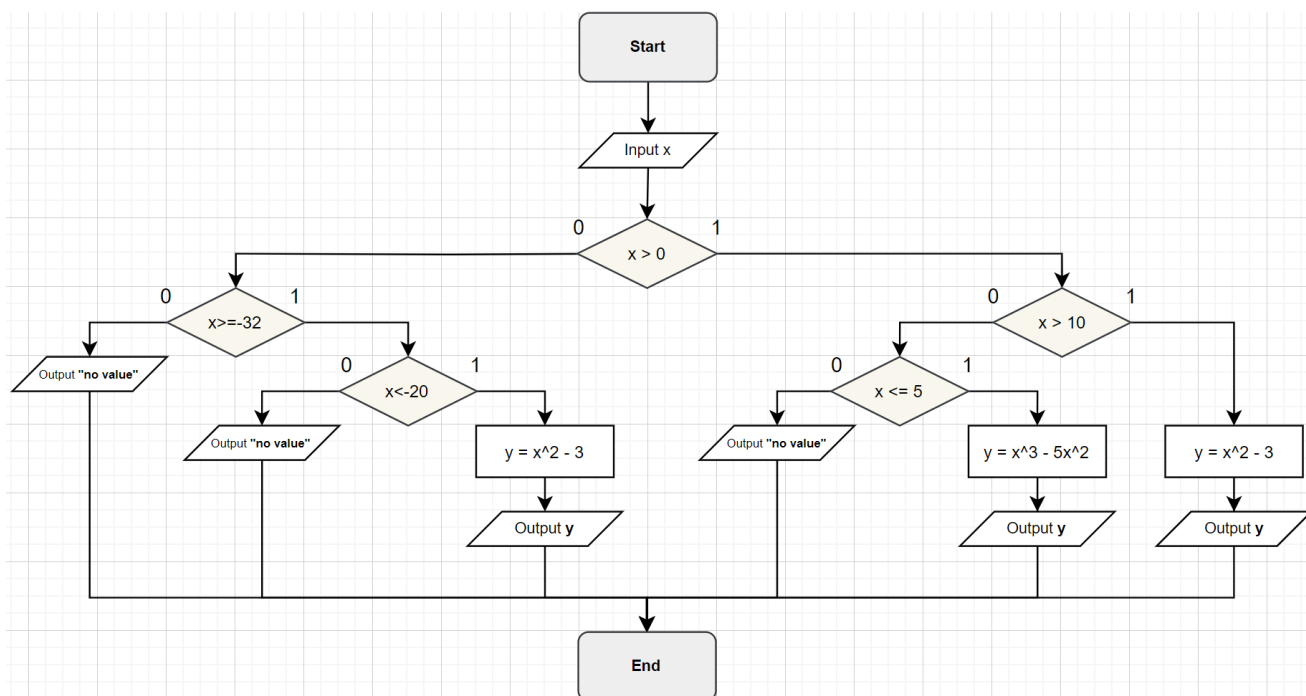
- 1) у програмі дозволяється використовувати тільки одиничні операції порівняння ($<$, $<=$, $>$, $>=$), і **не** дозволяється використовувати булеві (логічні) операції ($!$, $\&\&$, $||$ тощо);
- 2) у програмі необхідно обов'язково використати булеві (логічні) операції ($!$, $\&\&$, $||$ тощо); використання булевих операцій не повинно бути надлишковим.

Варіант № 32

$$y = \begin{cases} x^3 - 5x^2, & x \in (0, 5] \\ x^2 - 3, & x \in [-32, -20) \cup (10, +\infty) \end{cases}$$

I спосіб:

Діаграма алгоритму



Текст програми

```
#include <stdio.h>

int main(void) {
    printf("Welcome!\nInput x: ");

    float x;
    float y;

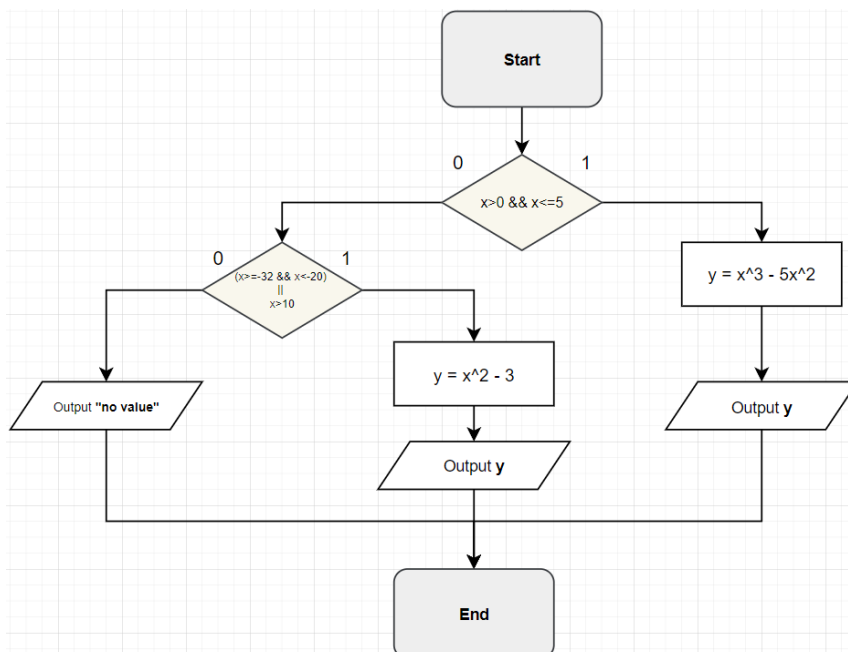
    scanf("%f", &x);

    if (x>0) {
        if (x>10) {
            y = (x*x)-3;
            printf("Found y value: %f\n", y);
        }
        else if (x<=5) {
            y = (x*x*x)-5*(x*x);
            printf("Found y value: %f\n", y);
        }
        else printf("There is no value for such x!\n");
    }
    else if (x>=-32) {
        if (x<-20) {
            y = (x*x)-3;
            printf("Found y value: %f\n", y);
        }
        else printf("There is no value for such x!\n");
    }
    else printf("There is no value for such x!\n");

    return 0;
}
```

II спосіб:

Діаграма алгоритму



Текст програми

```
#include <stdio.h>

int main(void) {
    printf("Welcome again!\nInput x: ");

    float x;
    float y;

    scanf("%f", &x);

    if (x>0 && x<=5) {
        y = (x*x*x)-5*(x*x);
        printf("Found y value: %f\n", y);
    }
    else if ((x>=-32&& x<=-20)||x>10) {
        y = (x*x)-3;
        printf("Found y value: %f\n", y);
    }
    else printf("There is no value for such x!\n ");

    return 0;
}
```

Результати тестування програм

Перевірка першої області визначення:

```
C:\archive\EducationalWork\ACД\lab1\programme\1>main.exe
Welcome!
Input x: 3
Found y value: -18.000000
C:\archive\EducationalWork\ACД\lab1\programme\2>main.exe
Welcome again!
Input x: 3
Found y value: -18.000000
```

```
C:\archive\EducationalWork\ACД\lab1\programme\1>main.exe
Welcome!
Input x: 7
There is no value for such x!
C:\archive\EducationalWork\ACД\lab1\programme\2>main.exe
Welcome again!
Input x: 7
There is no value for such x!
```

```
C:\archive\EducationalWork\ACД\lab1\programme\1>main.exe
Welcome!
Input x: -5
There is no value for such x!
C:\archive\EducationalWork\ACД\lab1\programme\2>main.exe
Welcome again!
Input x: -5
There is no value for such x!
```

Перевірка другої області визначення:

```
C:\archive\EducationalWork\ACД\lab1\programme\1>main.exe
Welcome!
Input x: -13
There is no value for such x!
C:\archive\EducationalWork\ACД\lab1\programme\2>main.exe
Welcome again!
Input x: -13
There is no value for such x!
```

```
C:\archive\EducationalWork\ACД\lab1\programme\1>main.exe
Welcome!
Input x: 13
Found y value: 166.000000
C:\archive\EducationalWork\ACД\lab1\programme\2>main.exe
Welcome again!
Input x: 13
Found y value: 166.000000
```

```
C:\archive\EducationalWork\ACД\lab1\programme\1>main.exe
Welcome!
Input x: -25
Found y value: 622.000000
C:\archive\EducationalWork\ACД\lab1\programme\2>main.exe
Welcome again!
Input x: -25
Found y value: 622.000000
```

```
C:\archive\EducationalWork\ACД\lab1\programme\1>main.exe
Welcome!
Input x: -40
There is no value for such x!
C:\archive\EducationalWork\ACД\lab1\programme\2>main.exe
Welcome again!
Input x: -40
There is no value for such x!
```

Перевірка критичних точок:

```
C:\archive\EducationalWork\ACД\lab1\programme\1>main.exe
Welcome!
Input x: 0
There is no value for such x!
C:\archive\EducationalWork\ACД\lab1\programme\2>main.exe
Welcome again!
Input x: 0
There is no value for such x!
```

```
C:\archive\EducationalWork\ACД\lab1\programme\1>main.exe
Welcome!
Input x: 5
Found y value: 0.000000
C:\archive\EducationalWork\ACД\lab1\programme\2>main.exe
Welcome again!
Input x: 5
Found y value: 0.000000
```

```
C:\archive\EducationalWork\ACД\lab1\programme\1>main.exe
Welcome!
Input x: 10
There is no value for such x!

C:\archive\EducationalWork\ACД\lab1\programme\2>main.exe
Welcome again!
Input x: 10
There is no value for such x!
```

```
C:\archive\EducationalWork\ACД\lab1\programme\1>main.exe
Welcome!
Input x: -20
There is no value for such x!

C:\archive\EducationalWork\ACД\lab1\programme\2>main.exe
Welcome again!
Input x: -20
There is no value for such x!
```

```
C:\archive\EducationalWork\ACД\lab1\programme\1>main.exe
Welcome!
Input x: -32
Found y value: 1021.000000

C:\archive\EducationalWork\ACД\lab1\programme\2>main.exe
Welcome again!
Input x: -32
Found y value: 1021.000000
```

Висновок

На лабораторній роботі я розв'язував задачу по знаходженню значення у залежно від області визначення, в яку попадає введене значення x , за допомогою мови програмування С. Задачу розв'язав двома способами: із використанням виключно одиничних операторів порівняння (1 спосіб) та з використанням логічних операцій (2 спосіб).

У ході лабораторної роботи навчився використовувати обидва види операторів для порівняння величин, а також будувати структуру та блок-схему алгоритму програми. Для себе визначив, що використання логічних операцій для порівняння є ліпшим варіантом, ніж використання одиничних операторів, через кращу оптимізацію та зменшення кількості рядків, що призводить до покращення читабельності та сприйняття коду.

Тому в подальшому буду використовувати саме логічні операції в тих випадках, де це можливо.