

**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»

Виконала:

студентка групи ІМ-43
Козаченко Софія Олександрівна
номер у списку групи: 14

Перевірила:

Молчанова А. А.

Київ 2024

Завдання

Задано дійсне число x . Визначити значення заданої за варіантом кусочнобезперервної 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):

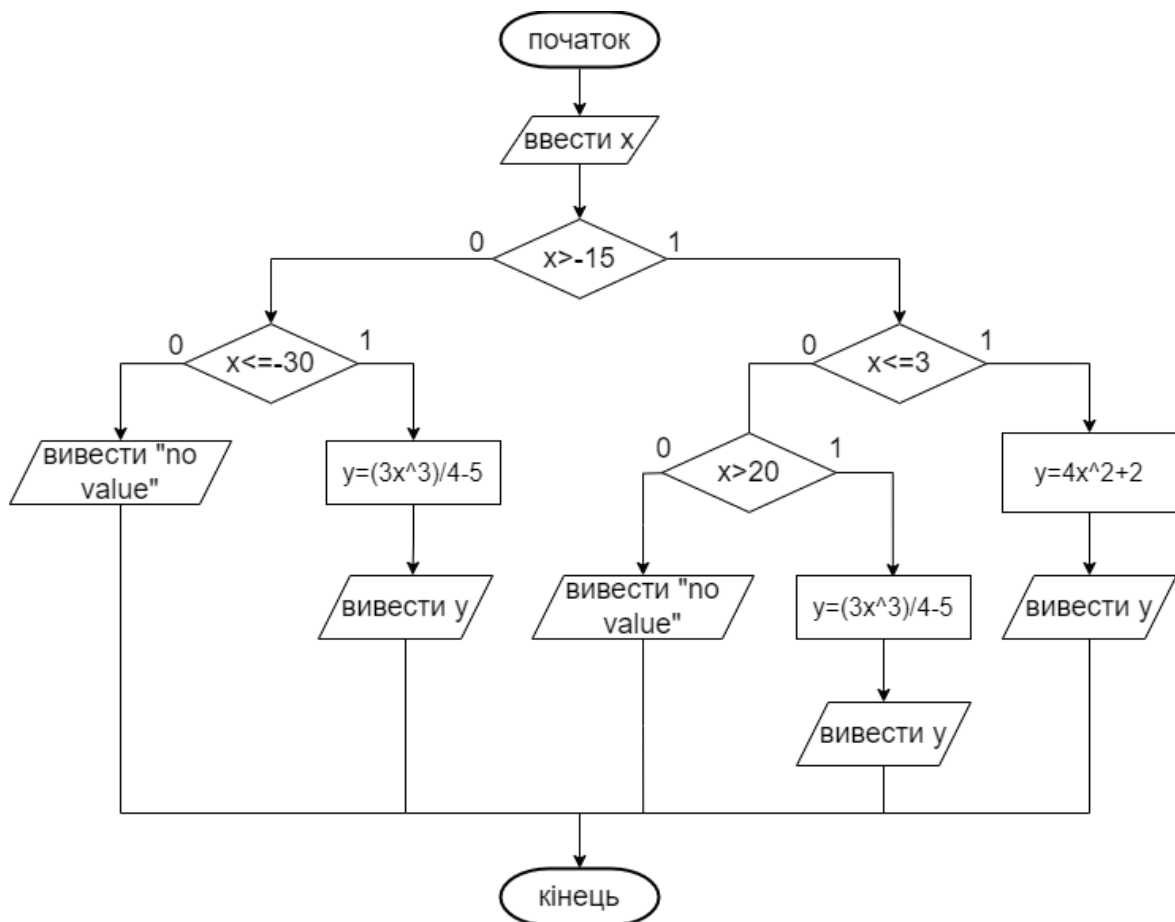
1) у програмі дозволяється використовувати тільки одиничні операції порівняння ($<$, $<=$, $>$, $>=$), і не дозволяється використовувати булеві (логічні) операції ($!$, $\&\&$, \parallel тощо);

2) у програмі необхідно обов'язково використати булеві (логічні) операції ($!$, $\&\&$, \parallel тощо); використання булевих операцій не повинно бути надлишковим.

Варіант 14:

$$y = \begin{cases} 4x^2 + 2, & x \in (-15, 3] \\ 3x^3/4 - 5, & x \in (-\infty, -30] \cup (20, +\infty) \end{cases}$$

Діаграма алгоритму(1 спосіб):



Текст програми(1 спосіб):

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

int main()
{
    float x;
    float y;

    printf("input x\n");
    scanf("%f", &x);

    if(x>-15)
    {
        if(x<=3)
        {
            y=4*x*x+2;
            printf("y=%f\n", y);
            printf("x belongs to the interval (-15;3] \n");
        }
        else
        {
            if(x>20)
            {
                y=3*x*x*x/4-5;
                printf("y=%f\n", y);
                printf("x belongs to the interval (20;+infinity)
\n");
            }
            else
            {
                printf("no value\n");
            }
        }
    }

    else
    {
        if(x<=-30)
```

```

        {
            y=3*x*x*x/4-5;
            printf("y=%f\n", y);
            printf("x belongs to the interval (-infinity;-30] \n");
        }
        else
        {
            printf("no value\n");
        }
    }
    return 0;
}

```

Результати тестування програми(1 спосіб)

```

input x
-32
y=-24581.000000
x belongs to the interval (-infinity;-30]

Process returned 0 (0x0)   execution time : 4.434 s
Press any key to continue.

```

```

input x
-21
no value

Process returned 0 (0x0)   execution time : 15.073 s
Press any key to continue.

```

```

input x
-9
y=326.000000
x belongs to the interval (-15;3]

Process returned 0 (0x0)   execution time : 3.846 s
Press any key to continue.

```

```

input x
6
no value

Process returned 0 (0x0)   execution time : 16.669 s
Press any key to continue.

```

```

input x
29
y=18286.750000
x belongs to the interval (20;+infinity)

Process returned 0 (0x0)   execution time : 3.740 s
Press any key to continue.

```

Результати тестування критичних точок:

```
input x
-30
y=-20255.000000
x belongs to the interval (-infinity;-30]

Process returned 0 (0x0)   execution time : 4.422 s
Press any key to continue.
```

```
input x
-15
no value

Process returned 0 (0x0)   execution time : 8.082 s
Press any key to continue.
```

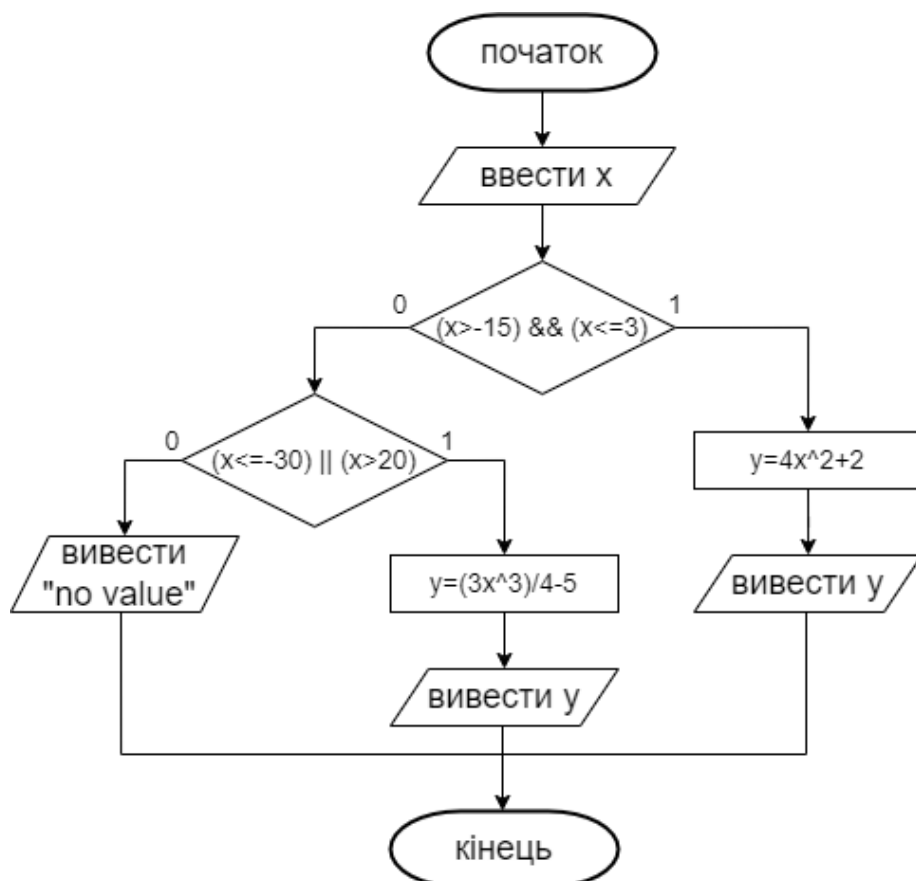
```
input x
3
y=38.000000
x belongs to the interval [-15;3]

Process returned 0 (0x0)   execution time : 3.574 s
Press any key to continue.
```

```
input x
20
no value

Process returned 0 (0x0)   execution time : 3.714 s
Press any key to continue.
```

Діаграма алгоритму(2 спосіб):



Текст програми(2 способ):

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

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

```
int main()
```

```
{
```

```
    float x;
```

```
    float y;
```

```
    printf("input x\n");
```

```
    scanf("%f", &x);
```

```
    if((x>-15) && (x<=3))
```

```
    {
```

```
        y=4*x*x+2;
```

```
        printf("y=%f\n", y);
```

```
        printf("x belongs to the interval (-15;3] \n");
```

```
    }
```

```
    else
```

```
    {
```

```
        if((x<=-30) || (x>20))
```

```
        {
```

```
            y=3*x*x*x/4-5;
```

```
            printf("y=%f\n", y);
```

```
            printf("x belongs to the interval (-infinity;-30) u (20;+infinity) \n");
```

```
        }
```

```
    else
```

```
    {
```

```

        printf("no value\n");
    }

}

return 0;
}

```

Результати тестування програми(2 способ):

```

input x
-36
y=-34997.000000
x belongs to the interval (-infinity;-30) u (20;+infinity)

Process returned 0 (0x0)   execution time : 4.017 s
Press any key to continue.

```

```

input x
-24
no value

Process returned 0 (0x0)   execution time : 4.260 s
Press any key to continue.

```

```

input x
0
y=2.000000
x belongs to the interval [-15;3]

Process returned 0 (0x0)   execution time : 21.056 s
Press any key to continue.

```

```

input x
11
no value

Process returned 0 (0x0)   execution time : 1.747 s
Press any key to continue.

```

```

input x
51
y=99483.250000
x belongs to the interval (-infinity;-30) u (20;+infinity)

Process returned 0 (0x0)   execution time : 6.990 s
Press any key to continue.

```

Результати тестування критичних точок

```

input x
-30
y=-20255.000000
x belongs to the interval (-infinity;-30) u (20;+infinity)

Process returned 0 (0x0)   execution time : 5.556 s
Press any key to continue.

```

```
input x
-15
no value

Process returned 0 (0x0)   execution time : 7.198 s
Press any key to continue.
```

```
input x
3
y=38.000000
x belongs to the interval [-15;3]

Process returned 0 (0x0)   execution time : 3.884 s
Press any key to continue.
```

```
input x
20
no value

Process returned 0 (0x0)   execution time : 8.678 s
Press any key to continue.
```

Висновок :

Виконуючи цю лабораторну роботу, я засвоїла теоретичний матеріал на тему “Розгалужені алгоритми ” та набула практичних навичок у використанні керуючих конструкцій розгалуження та булевих операцій. Я виконала цю роботу двома способами, з використанням одиничних та булевих операцій. І у висновку, я зрозуміла, що булеві операції спрощують написання конструкцій розгалуження, а за допомогою одиничних операцій я розібралась, як саме ці конструкції працюють.