

**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
Костеніч Степан Станіславович
номер у списку групи: 17

Перевірила:

Молчанова А. А.

Київ 2024

Постановка задачі

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.

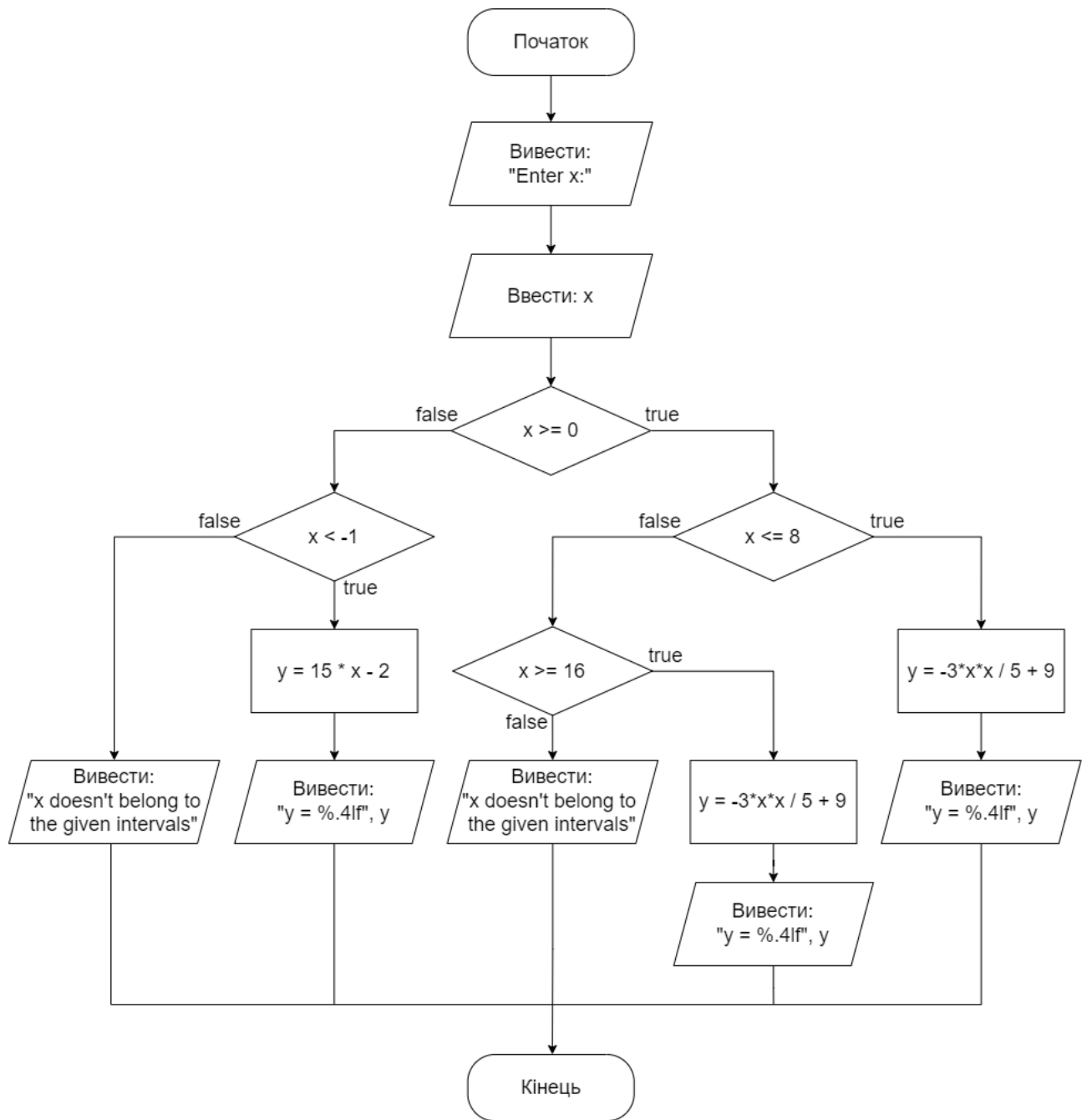
При проєктуванні програм *слід врахувати наступне*:

- 1) обчислення значень функції слід виконати без використання функцій математичної бібліотеки;
- 2) програми мають бути написані мовою програмування C.

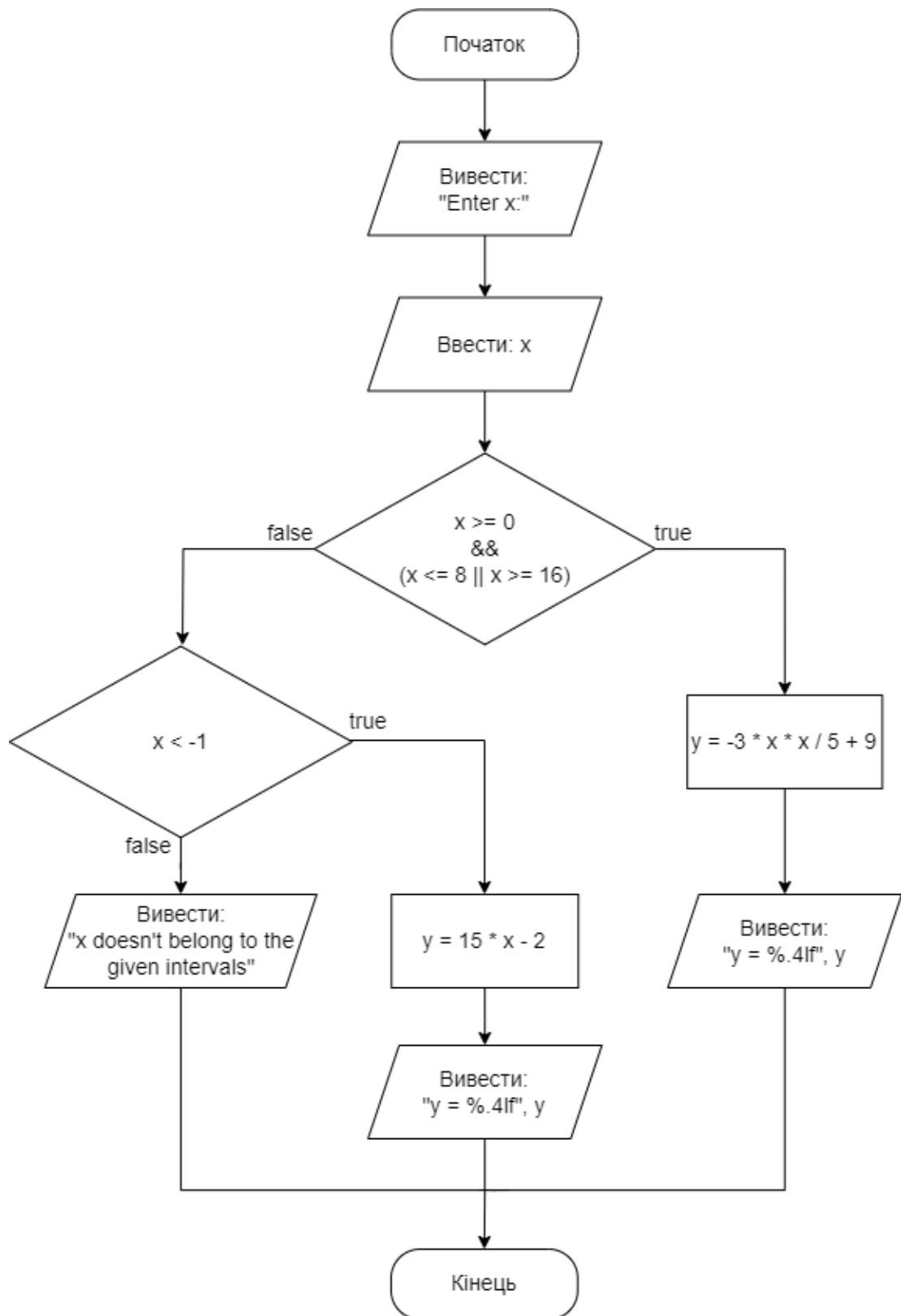
Варіант №17

$$y = \begin{cases} -3x^2/5 + 9, & x \in [0, 8] \cup [16, +\infty) \\ 15x - 2, & x \in (-\infty, -1) \end{cases}$$

Діаграма алгоритму №1



Діаграма алгоритму №2



Текст програми №1

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

```
int main(void) {
    double x;
    double y;

    printf("\nEnter x: \n");
    scanf("%lf", &x);

    if (x >= 0) {
        if (x <= 8) {
            y = -3 * x * x / 5 + 9;
            printf("y = %.4lf", y);
        } else if (x >= 16) {
            y = -3 * x * x / 5 + 9;
            printf("y = %.4lf", y);
        } else {
            printf("x doesn't belong to the given intervals");
        }
    } else if (x < -1) {
        y = 15 * x - 2;
        printf("y = %.4lf", y);
    } else {
        printf("x doesn't belong to the given intervals");
    }

    return 0;
}
```

Текст програми №2

```
#include <stdio.h>

int main(void) {
    double x;
    double y;

    printf("\nEnter x: \n");
    scanf("%lf", &x);

    if (x >= 0 && (x <= 8 || x >= 16)) {
        y = -3.0 * x * x / 5.0 + 9.0;
        printf("y = %.4lf", y);
    } else if (x < -1) {
        y = 15 * x - 2;
        printf("y = %.4lf", y);
    } else {
        printf("x doesn't belong to the given intervals");
    }

    return 0;
}
```

Результати тестування програми №1

```
Enter x:  
-2  
y = -32.0000  
Process finished with exit code 0
```

$$-2 + 15 \times (-2)$$

-32

```
Enter x:  
-1.1  
y = -18.5000  
Process finished with exit code 0
```

$$-2 + 15 \times (-1.1)$$

-18.5

```
Enter x:  
1  
y = 8.4000  
Process finished with exit code 0
```

$$9 - 3 \times 1^2 \div 5$$

8.4

```
Enter x:  
0.1  
y = 8.9940  
Process finished with exit code 0
```

$$9 - 3 \times 0.1^2 \div 5$$

8.994

```
Enter x:  
7  
y = -20.4000  
Process finished with exit code 0
```

$$9 - 3 \times 7^2 \div 5$$

-20.4

```
Enter x:  
7.65  
y = -26.1135  
Process finished with exit code 0
```

$$9 - 3 \times 7.65^2 \div 5$$

-26.1135

```
Enter x:  
10  
x doesn't belong to the given intervals  
Process finished with exit code 0
```

```
Enter x:  
17  
y = -164.4000  
Process finished with exit code 0
```

$$9 - 3 \times 17^2 \div 5$$

-164.4

```
Enter x:  
16.15  
y = -147.4935  
Process finished with exit code 0
```

$$9 - 3 \times 16.15^2 \div 5$$

-147.4935

```
Enter x:  
200.65  
y = -24147.2535  
Process finished with exit code 0
```

$$9 - 3 \times 200.65^2 \div 5$$

-24,147.2535

```
Enter x:  
-1  
x doesn't belong to the given intervals  
Process finished with exit code 0
```

```
Enter x:  
0  
y = 9.0000  
Process finished with exit code 0
```


$$9 - 3 \times 0^2 \div 5$$

9

Enter x:

8

y = -29.4000

Process finished with exit code 0

$$9 - 3 \times 8^2 \div 5$$

-29.4

Enter x:

16

y = -144.6000

Process finished with exit code 0

$$9 - 3 \times 16^2 \div 5$$

-144.6

Результати тестування програми №2

```
Enter x:  
-1.5  
y = -24.5000  
Process finished with exit code 0
```

$$-2 + 15 \times (-1.5) \\ -24.5$$

```
Enter x:  
0.7  
y = 8.7060  
Process finished with exit code 0
```

$$9 - 3 \times 0.7^2 \div 5 \\ 8.706$$

```
Enter x:  
7.7  
y = -26.5740  
Process finished with exit code 0
```

$$9 - 3 \times 7.7^2 \div 5 \\ -26.574$$

```
Enter x:  
10  
x doesn't belong to the given intervals  
Process finished with exit code 0
```

```
Enter x:  
16.7  
y = -158.3340  
Process finished with exit code 0
```

$$9 - 3 \times 16.7^2 \div 5 \\ -158.334$$

```
Enter x:  
-1  
x doesn't belong to the given intervals  
Process finished with exit code 0
```

Enter x:

0

y = 9.0000

Process finished with exit code 0

$$9 - 3 \times 0^2 \div 5$$

9

Enter x:

8

y = -29.4000

Process finished with exit code 0

$$9 - 3 \times 8^2 \div 5$$

-29.4

Enter x:

16

y = -144.6000

Process finished with exit code 0

$$9 - 3 \times 16^2 \div 5$$

-144.6

Висновок

Під час виконання лабораторної роботи № 1.1 «Розгалужені алгоритми» я засвоїв теоретичний матеріал та набув практичних навичок використання керуючих конструкцій розгалуження та булевих операцій.

Я зрозумів синтаксис використання оператора `if – else`, і, зробивши завдання дома способами, усвідомив корисність та зручність залучення булевих операторів. На прикладі двох способів стало явно видно громіздкість коду з використанням тільки одиничних операцій порівняння. Також такий код було складніше читати та легше було помилитися при написанні. При застосуванні у другому способі булевих операторів код став меншим, компактнішим та більш читабельним, полегшивши роботу.

Також при написанні коду ближче познайомився із синтаксисом `scanf()` для `float` і `double`. Формати `%f` та `%lf` відповідно. Зрозумів різницю розмірів `float` і `double` під час тестування програми з великими числами. `Double` є точнішим за `float` і може зберігати 64 біти. І для обмеження кількості цифр після коми ознайомився з потрібним форматом для `printf()`. Щоб вказати, скільки цифр виводити крапки, потрібно використати специфікатор формату `%.4lf` (для `double`).

Отже, виконання лабораторної роботи № 1.1 «Розгалужені алгоритми» було корисним, дозволило закріпити теоретичні знання та набути практичних навичок в області програмування мовою C.