**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: Перевірила:

студент групи IM-43 Молчанова А. А.  
Олексійчук Станіслав Юрійович  
номер у списку групи: 23

Київ 2024

**Постановка задачі**

Задано дiйсне число x. Визначити значення заданої за варiантом кусочно безперервної функцiї y(x), якщо воно iснує, або вивести на екран повiдомлення про неiснування функцiї для заданого x.

Розв’язати задачу двома способами (створити двi блок-схеми алгоритмiв та написати двi програми):

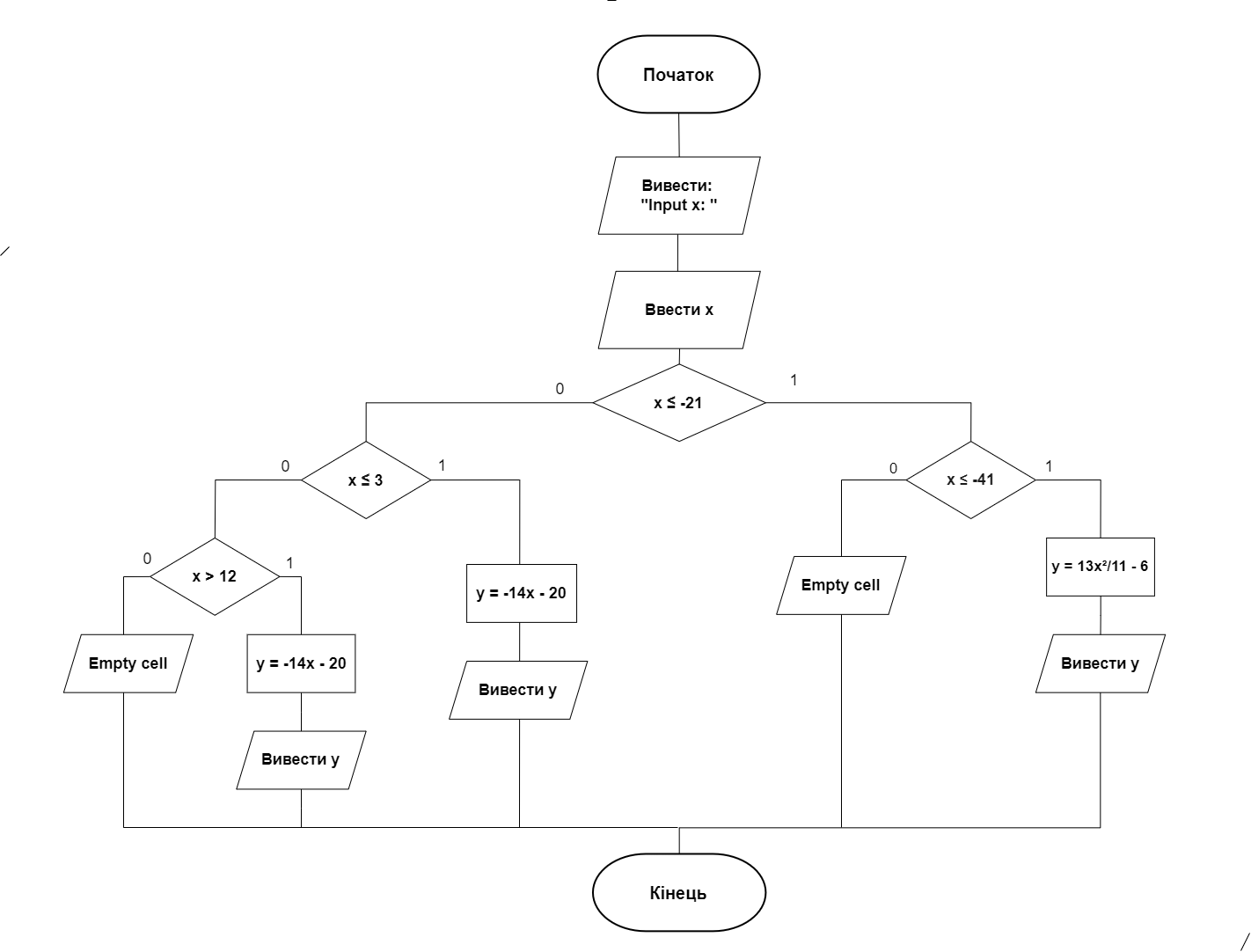
1. у програмi дозволяється використовувати тiльки одиничнi операцiї порiвняння (<, <=, >, >=), i не дозволяється використовувати булевi (логiчнi) операцiї (!, &&, || тощо);
2. у програмi необхiдно обов’язково використати булевi (логiчнi) операцiї (!, &&, || тощо); використання булевих операцiй не повинно бути надлишковим.

При проєктуваннi програм слiд врахувати наступне:

1. обчислення значень функцiї слiд виконати без використання функцiй математичної бiблiотеки;
2. програми мають бути написанi мовою програмування С.

***Варіант 23:***

***1 підпункт (not boolean)***

**Діаграма алгоритму**

**Текст програми**

#include <stdio.h>

#include <stdlib.h>

int main()

{

float x;

double y;

printf("Input x: ");

scanf("%f", &x);

if (x<=-21)

{

printf("Number %f",x);

printf(" is less than or equal to -21.\n");

if (x<=-41)

{

y=(double)13\*x\*x/(double)11 - (double)6;

printf("It is also less than or equal to -41! So... ");

printf("Variable \"y\" = %f",y);

printf("!");

}

else

{

printf("It is also greater than -41! ");

printf("So... Variable \"y\" belongs to the empty cell!");

}

}

else

{

printf("Number %f",x);

printf(" is greater than -21.\n");

if (x<=3)

{

y=(double)-14\*x - (double)20;

printf("It is also less than or equal to 3! So... ");

printf("Variable \"y\" = %f",y);

printf("!");

}

else

{

printf("It is also greater than 3");

if (x>12)

{

y=(double)-14\*x - (double)20;

printf(" and greater than 12! So... ");

printf("Variable \"y\" = %f",y);

printf("!");

}

else

{

printf(", but less than or equal to 12! So... ");

printf("Variable \"y\" belongs to the empty cell!");

}

}

}

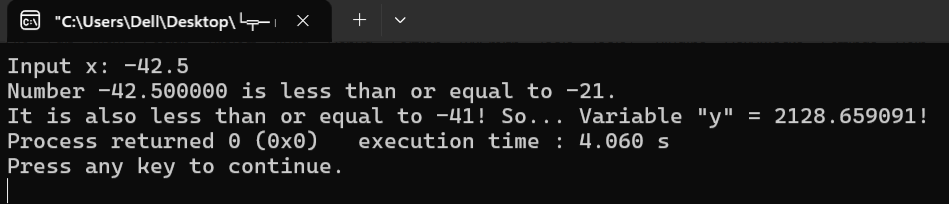
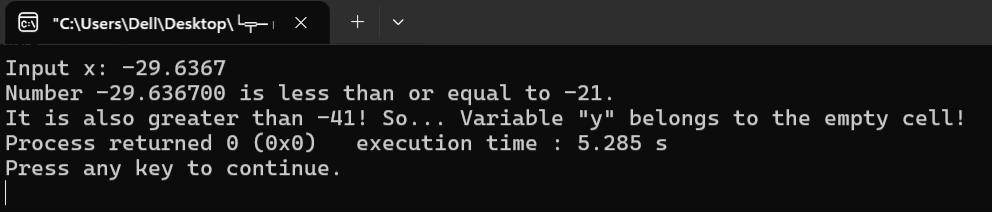
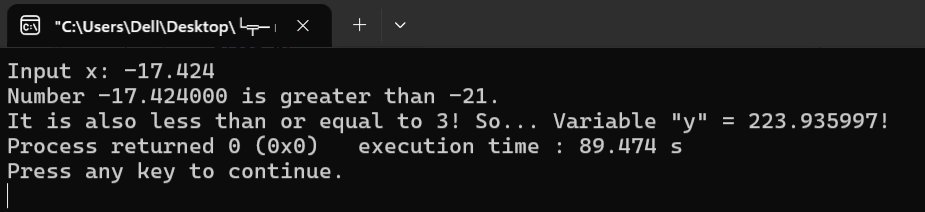
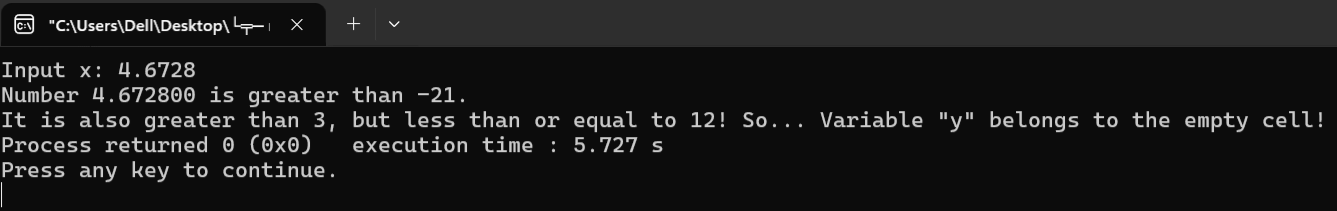
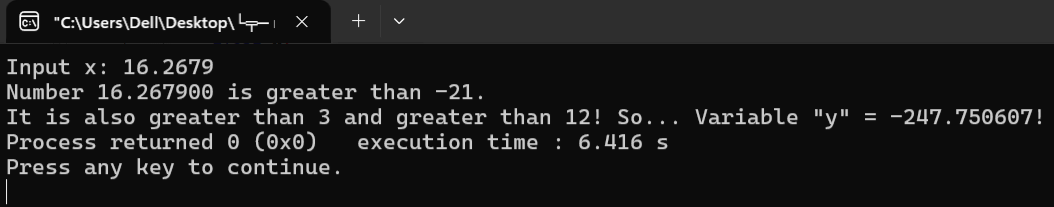
return 0;

}

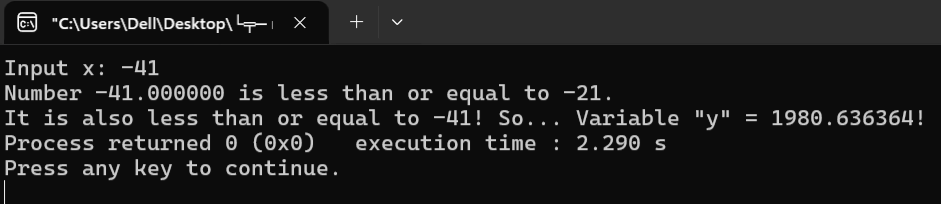
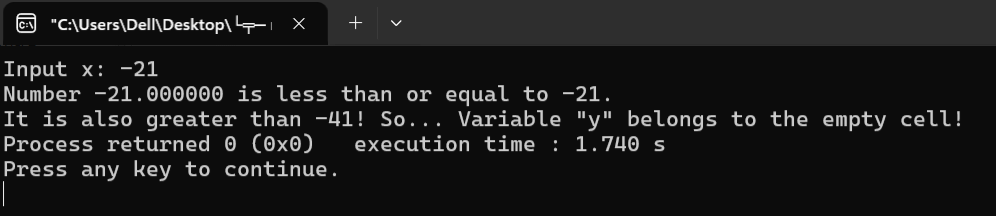
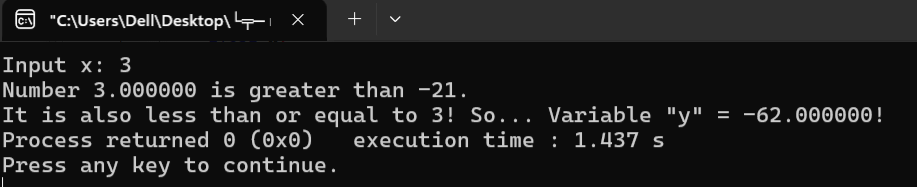
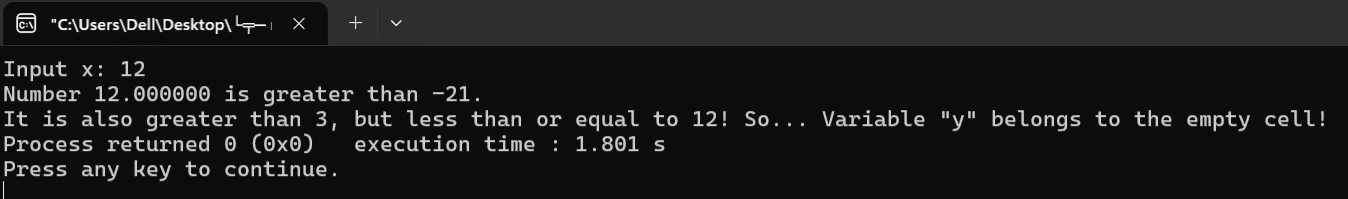
**Результати тестування програми**

Зважаючи на задачу (систему), видно, що в нас є три проміжки, для яких виконуватиметься функція.

Тобто для тестів потрібно перевірити числа x, які:

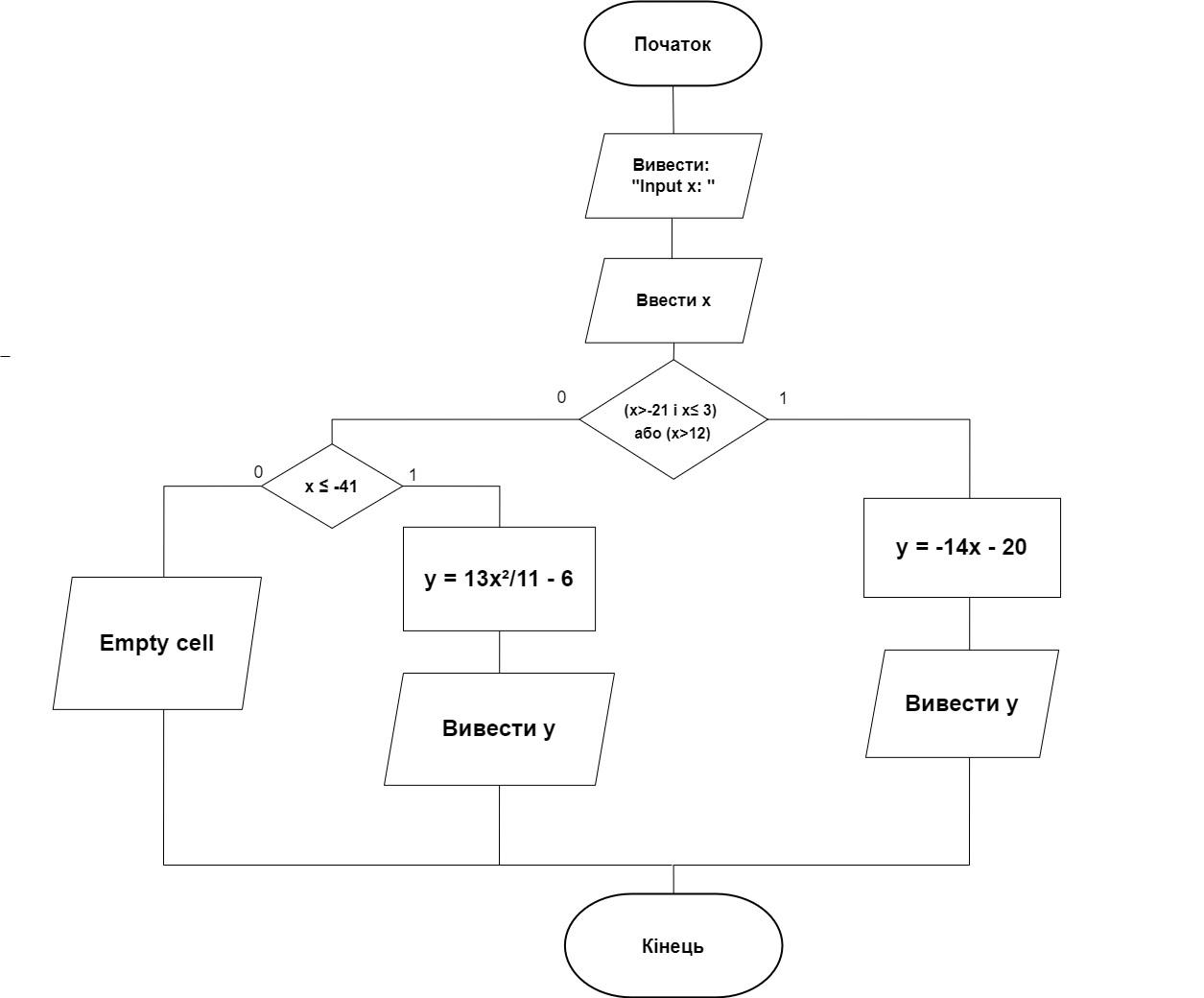
1. x ≤ -41, тобто x [∈](https://symbl.cc/ru/html-entities/) (-∞; -41] (потрапляє, );
2. -41 < x ≤ -21, тобто x [∈](https://symbl.cc/ru/html-entities/) (-41; -21] (не потрапляє);
3. -21 < x ≤ 3, тобто x [∈](https://symbl.cc/ru/html-entities/) (-21; 3] (потрапляє, );
4. 3 < x ≤ 12, тобто x [∈](https://symbl.cc/ru/html-entities/) (3; 12] (не потрапляє);
5. x > 12, тобто x [∈](https://symbl.cc/ru/html-entities/) (12; +∞) (потрапляє, ).
6. 
7. 
8. 
9. 
10. 

Тепер перевірмо критичні точки, аби розібрати можливі варіанти розвитку подій:

1. x = -41, тобто x [∈](https://symbl.cc/ru/html-entities/) (-∞; -41] (потрапляє, );
2. x = -21, тобто x [∈](https://symbl.cc/ru/html-entities/) (-41; -21] (не потрапляє);
3. x = 3, тобто x [∈](https://symbl.cc/ru/html-entities/) (-21; 3] (потрапляє, );
4. x = 12, тобто x [∈](https://symbl.cc/ru/html-entities/) (3; 12] (не потрапляє).
5. 
6. 
7. 
8. 

***2 підпункт (boolean)***

**Діаграма алгоритму**



**Текст програми**

#include <stdio.h>

#include <stdlib.h>

int main()

{

float x;

double y;

printf("Input x: ");

scanf("%f", &x);

if ((x>-21 && x<=3) || (x>12))

{

y=(double)-14\*x - (double)20;

printf("Number %f",x);

printf(" is greater than -21 and less than or equal to 3.");

printf(" Or it can be greater than 12. So...\n");

printf("Variable \"y\" = %f",y);

printf("!");

}

else if (x<=-41)

{

y=(double)13\*x\*x/(double)11 - (double)6;

printf("Number %f",x);

printf(" is less than or equal to -41. So...\n");

printf("Variable \"y\" = %f",y);

printf("!");

}

else

{

printf("Number %f",x);

printf(" is greater than -41 and less than or equal to -21.");

printf(" Or it can be greater than 3 and less than or equal to 12. So...\n");

printf("Variable \"y\" belongs to the empty cell!");

}

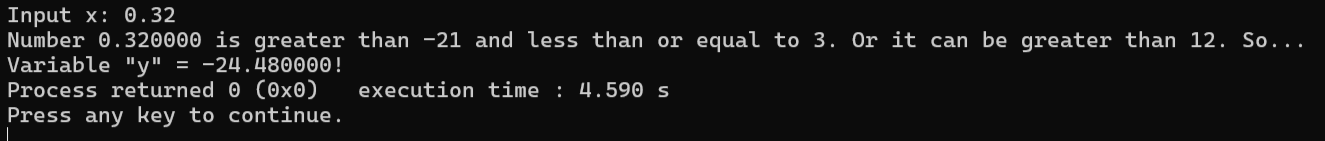
return 0;

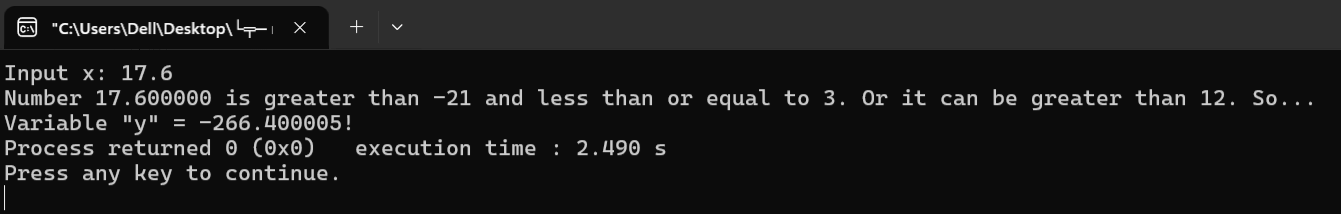
}

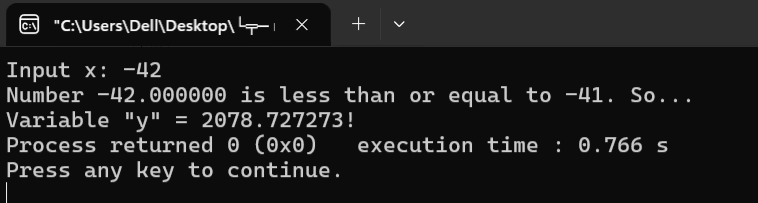
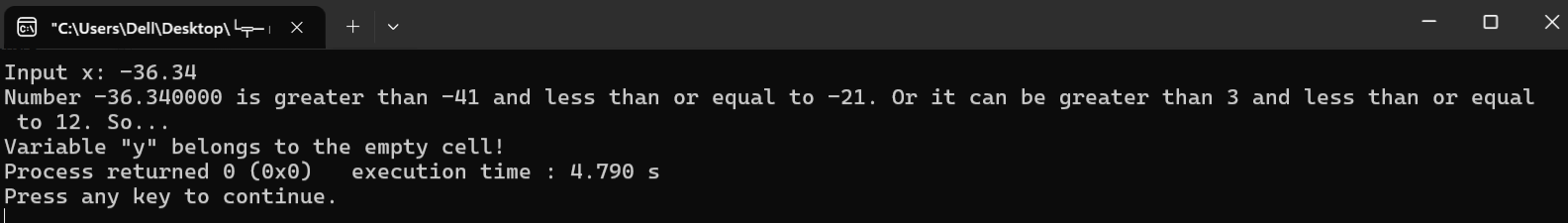
**Результати тестування програми**

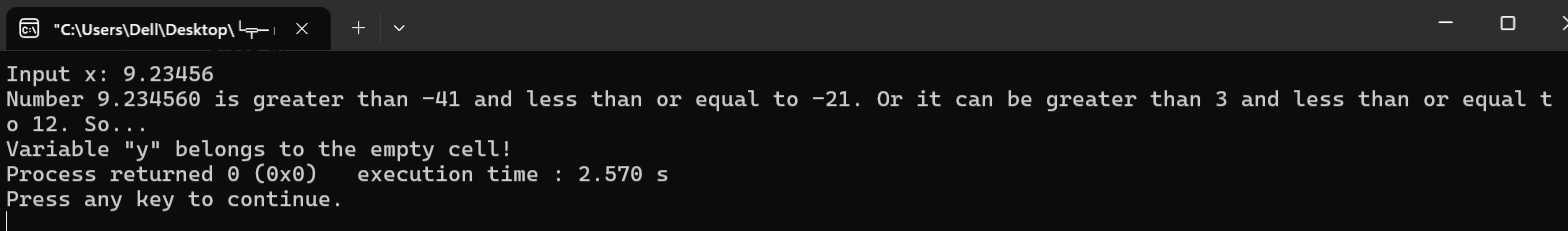
Зважаючи на задачу (систему), видно, що в нас є три проміжки, для яких виконуватиметься функція.

Тобто для тестів потрібно перевірити числа x, які:

1. x [∈](https://symbl.cc/ru/html-entities/) (-21; 3] ∪ (12; +∞) (потрапляє, ), перевірмо дві точки для кожного проміжку;
2. x [∈](https://symbl.cc/ru/html-entities/) (-∞; -41] (потрапляє, );
3. x [∈](https://symbl.cc/ru/html-entities/) (-41; -21] ∪ (3; 12] (не потрапляє), перевірмо дві точки для кожного проміжку.
4. а) 

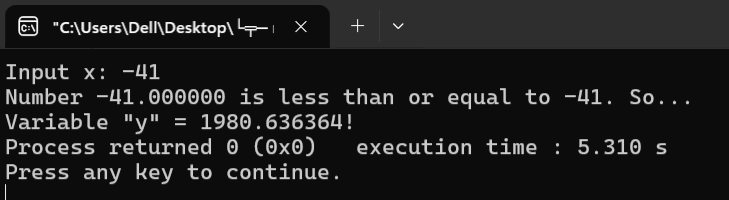
б) 

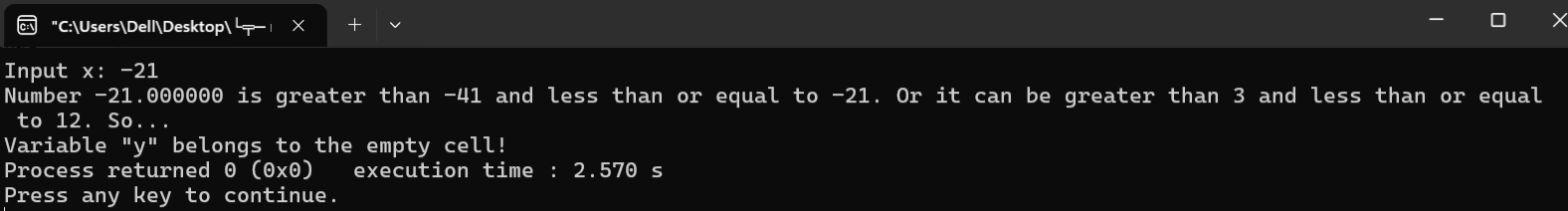
1. 
2. а) 

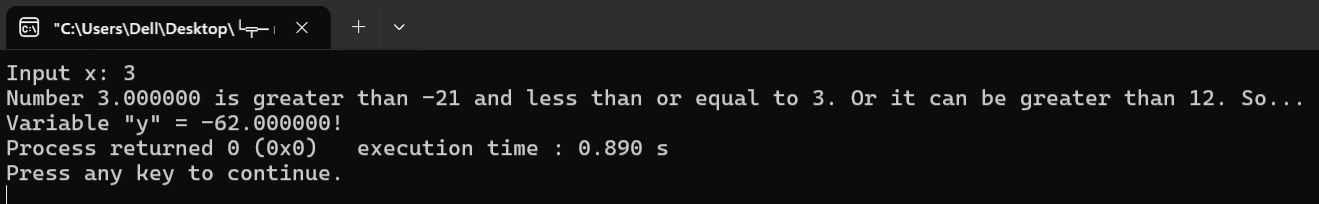
б) 

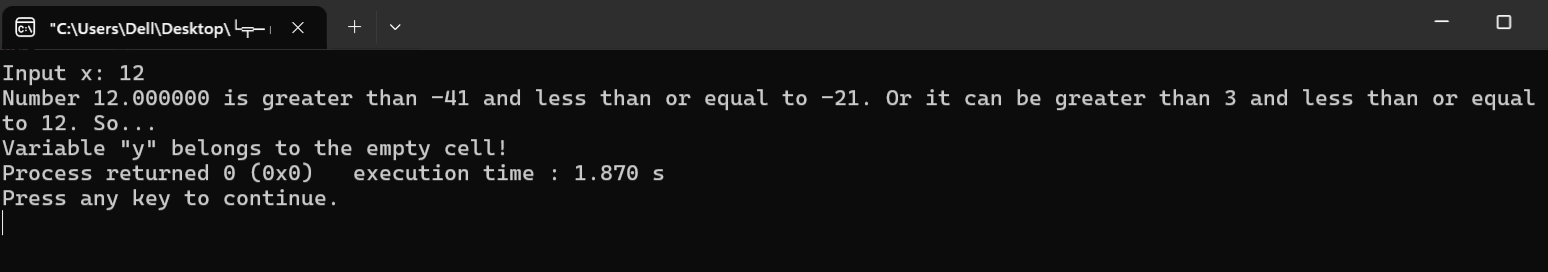
Тепер перевірмо критичні точки, аби розібрати можливі варіанти розвитку подій:

1. x = -41, тобто x [∈](https://symbl.cc/ru/html-entities/) (-∞; -41] (потрапляє, );
2. x = -21, тобто x [∈](https://symbl.cc/ru/html-entities/) (-41; -21] (не потрапляє);
3. x = 3, тобто x [∈](https://symbl.cc/ru/html-entities/) (-21; 3] (потрапляє, );
4. x = 12, тобто x [∈](https://symbl.cc/ru/html-entities/) (3; 12] (не потрапляє).

1) 

2) 

3) 

4) 

***Висновок:*** я навчився писати елементарні програми на мові програмування С, використовуючи середовище CodeBlocks; зрозумів, як працюють деякі типи даних, наприклад int, float, double; навчився розв’язувати систему варіацій функції, використовуючи умовні оператори з розгалуженням; почав використовувати функції, булеві (логічні) та одиничні операції порівняння; навчився аналізувати I/O програми; навчився відображати умовне розгалуження з допомогою блок-схем;