

НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ БІОРЕСУРСІВ І ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ

Факультет інформаційних технологій

ЛАБОРАТОРНА РОБОТА №5

Створення власних функцій

Виконала:
студентка групи ІПЗ-210076
Соколовська Софія

ВАРІАНТ – 18

Тема: Створення власних функцій.

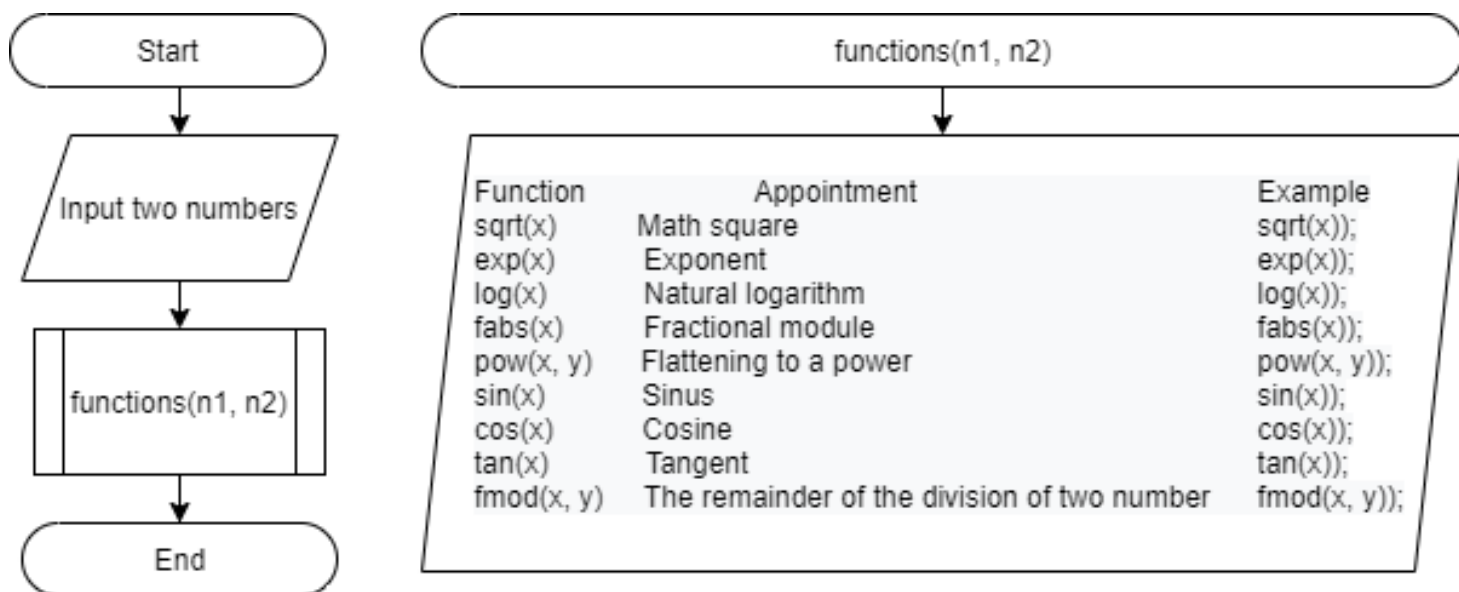
Мета: Опанування практичними навичками розробки програм із використанням функцій. Навчитися визначати функції та їхні прототипи, правильно організовувати виклик функції.

Завдання 1

Деякі загальні математичні обчислення програміст може реалізувати у своїй програмі за допомогою функцій стандартної математичної бібліотеки **math** мови C. Напишіть програму, яка перевіряє виклики функцій математичної бібліотеки та виводить результати у вигляді таблиці з такими фіксованими рядком та стовпцем:

Function	Appointment	Example
sqrt(x)		
exp(x)		
log(x)		
fabs(x)		
pow(x,y)		
sin(x)		
cos(x)		
tan(x)		
fmod(x)		

- 1) На рис. 1.1 наведена блок-схема алгоритму програми, яка виводить таблицю на екран та виконує дії.



2) Код програми:

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

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

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

```
int functions (float x, float y)
```

$$\{$$

```
printf("\nFunction\tAppointment\tExample");
```

```
printf("\nsqrt(x) \tMath square          \t%0.2f", sqrt(x));
```

```
printf("\nexp(x) \tExponent          \t%0.2f", exp(x));
```

```
printf("\nlog(x) \tNatural logarithm \t%0.2f", log(x));
```

```
printf("\nfabs(x) \tFractional module \t%0.2f", fabs(x));
```

[illegible]

```
printf("\nsin(x) \tSinus          \t%0.2f", sin(x));
```

```
printf("\ncos(x) \tCosine\t%0.2f", cos(x));
```

```
printf("\ntan(x) \tTangent \t%0.2f", tan(x));
```

```
printf("\nmod(x, y) \tThe remainder of the division of two numbers\t%.2f",
fmod(x, y));
```

$$\}$$

```
int main()
```

$$\{$$

```
float n1;
```

```
float n2;
```

```
printf("Input two numbers\n");
```

```
scanf("\n%f", &n1);
```

```
scanf("\n%f", &n2);
```

```
functions(n1, n2);  
  
}
```

3) На рис 1.2 наведений результат програми.

 D:\University\subjects\±ЁюуЁрьєтрээ \lab5\Debug\lab5.exe

Input two numbers

8

2

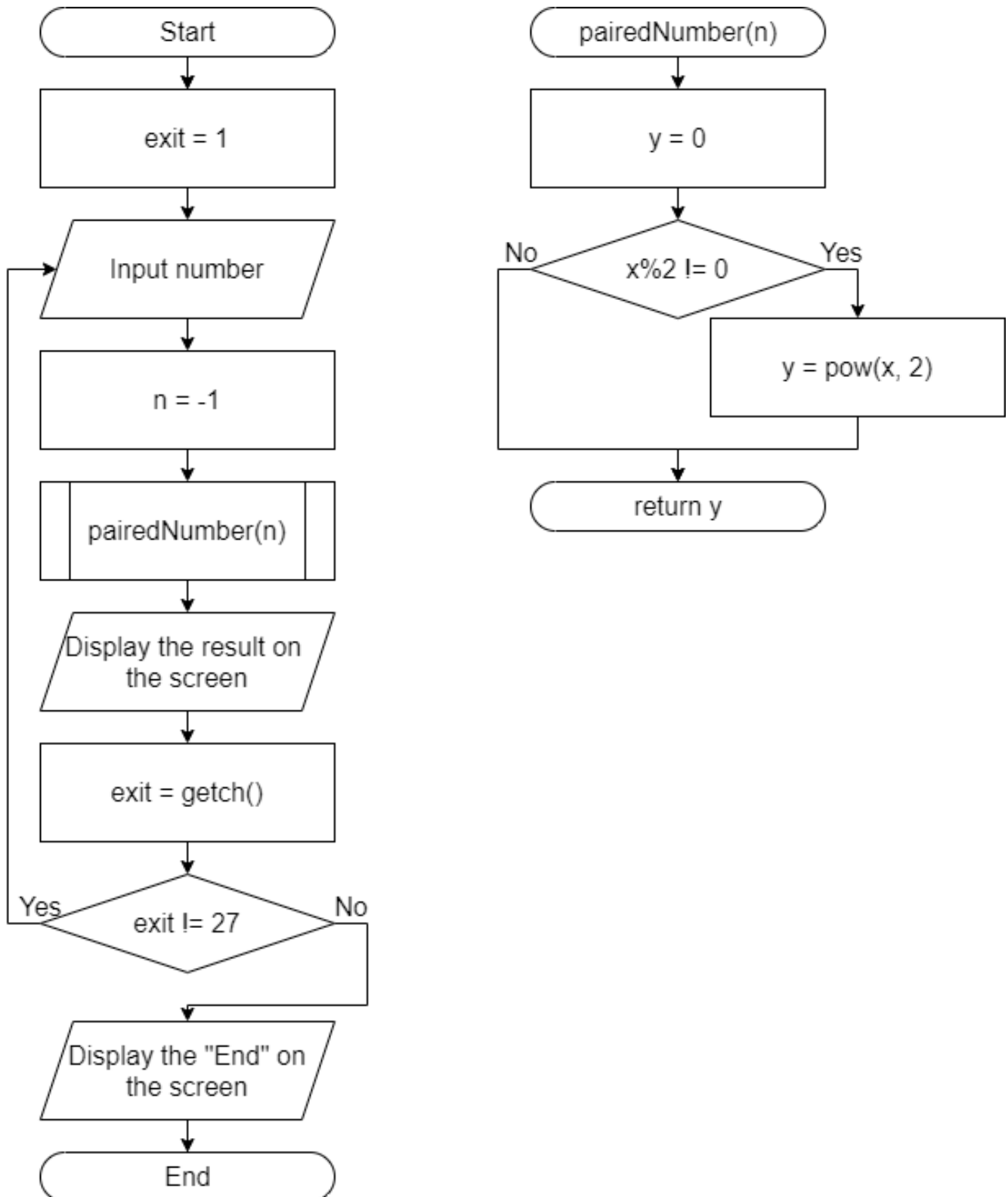
Function	Appointment	Example
sqrt(x)	Math square	2.83
exp(x)	Exponent	2980.96
log(x)	Natural logarithm	2.08
fabs(x)	Fractional module	8.00
pow(x, y)	Flattening to a power	64.00
sin(x)	Sinus	0.99
cos(x)	Cosine	-0.15
tan(x)	Tangent	-6.80
fmod(x, y)	The remainder of the division of two numbers	0.00

Process returned 0 (0x0) execution time : 16.011 s
Press any key to continue.

Завдання 2

Напишіть функцію, яка отримує ціле число та визначає, якщо це число непарне, то повертає це число в квадраті, інакше 0. Програма завершується, коли користувач натисне клавішу **ESC**.

- 1) На рис. 2.1 наведена блок-схема алгоритму програми, яка визначає парність/непарність числа.



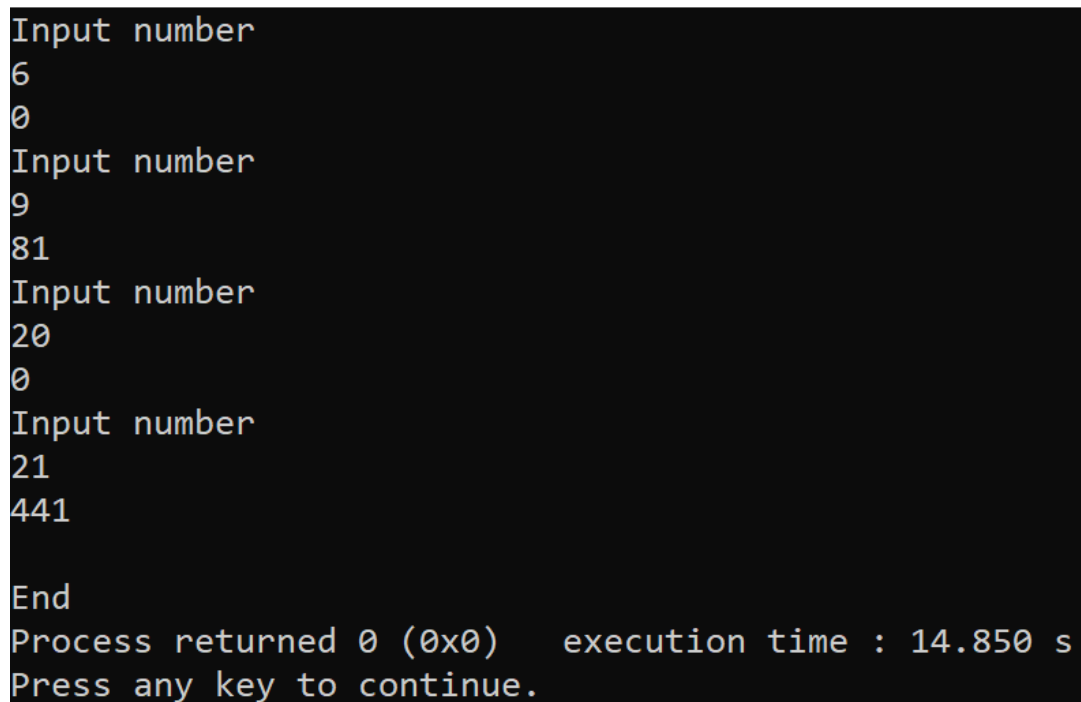
2) Код програми:

```
#include <stdio.h>
#include <stdlib.h>
#define esc 27

int main()
{
    int exit;
    do {
        printf("Input number\n");
        int n = -1;
        scanf("%i", &n);
        printf("%i\n", pairedNumber(n));
        exit = getch();
    } while(exit != 27);
    printf("\nEnd");
}

int pairedNumber(int x)
{
    int y = 0;
    if(x % 2 != 0) {
        y = pow(x, 2);
    }
    return y;
}
```

3) На рис. 2.2 наведений результат програми.



```
Input number
6
0
Input number
9
81
Input number
20
0
Input number
21
441

End
Process returned 0 (0x0)   execution time : 14.850 s
Press any key to continue.
```

Завдання 3

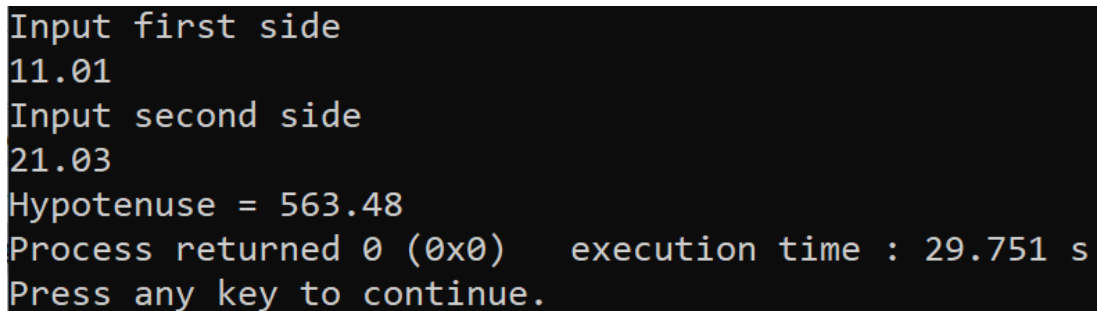
Напишіть прототипи таких функцій:

- а) функції **hypotenuse**, яка отримує два аргументи **side1**, **side2**, які мають тип з плаваючою крапкою подвійної точності, та повертає результат у вигляді значення з плаваючою крапкою подвійної точності;

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

float hypotenuse(float x, float y) {
    float h;
    h = pow(x, 2) + pow(y, 2);
    return h;
}

int main()
{
    float side1;
    float side2;
    float H;
    printf("Input first side\n");
    scanf("%f", &side1);
    printf("Input second side\n");
    scanf("%f", &side2);
    H = hypotenuse(side1, side2);
    printf("Hypotenuse = %0.2f", H);
}
```



```
Input first side
11.01
Input second side
21.03
Hypotenuse = 563.48
Process returned 0 (0x0)   execution time : 29.751 s
Press any key to continue.
```

- б) функції **help**, яка не отримує аргументів і нічого не повертає;

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

int main()
{
    //main function
}

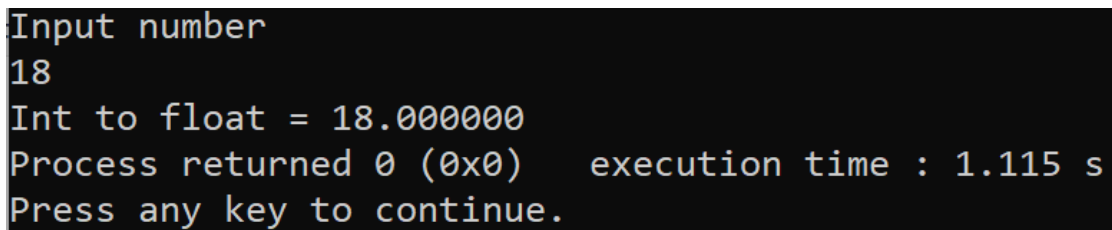
int help();
```

с) функції **intToFloat**, яка отримує цілий параметр **number** та повертає результат типу з плаваючою крапкою.

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

float intToFloat(int x) {
    float y = x;
    return y;
}

int main()
{
    int n;
    printf("Input number\n");
    scanf("%i", &n);
    printf("%f", intToFloat(n));
}
```



```
Input number
18
Int to float = 18.000000
Process returned 0 (0x0)   execution time : 1.115 s
Press any key to continue.
```

Висновок: В ході лабораторної роботи я опанувала практичними навичками розробки програм із використанням функцій. Навчилася визначати функції та їхні прототипи, правильно організовувати виклик функції.