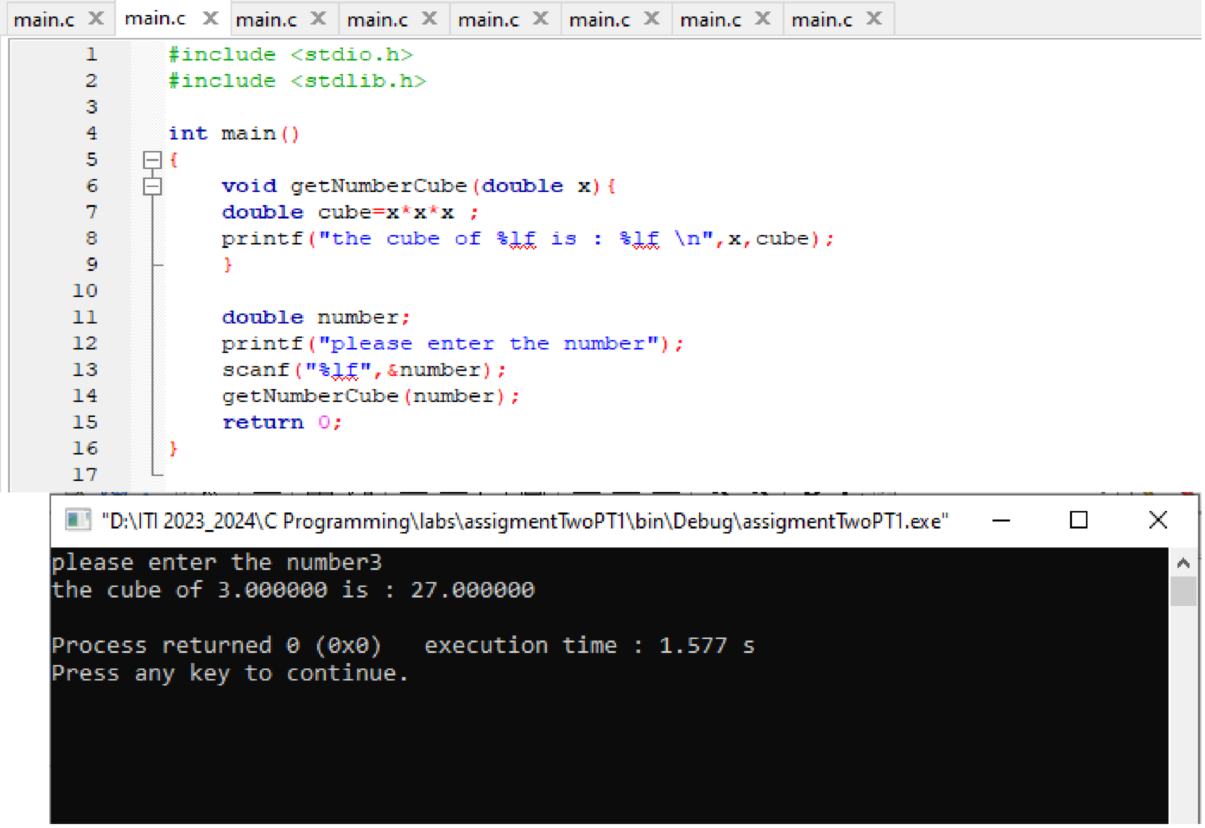
Assignment 2:

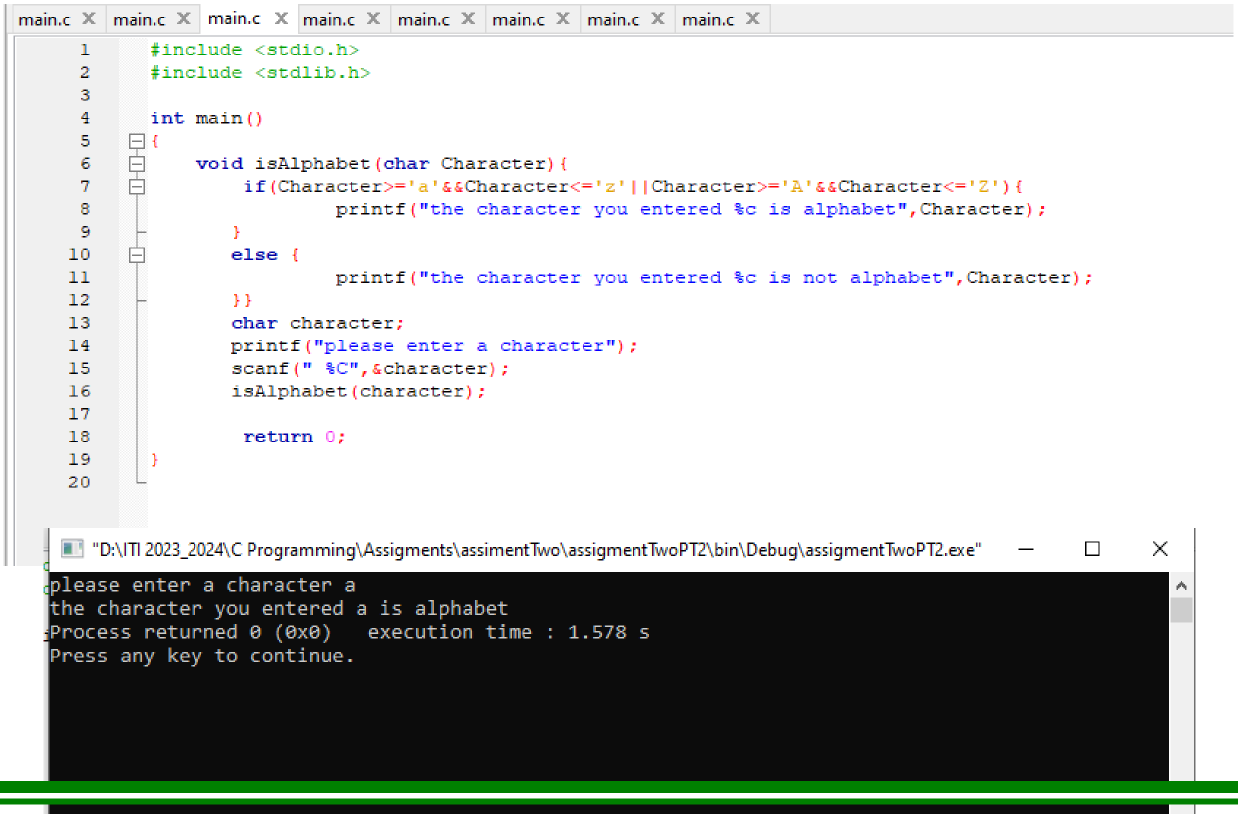
(1) Write a C Function that prints the cube of any number.


main.c ✕
main.c ✕
main.c ✕
main.c ✕
main.c ✕
main.c ✕
main.c ✕
main.c ✕
"D:\ITI 2023_2024\C Programming\labs\assigmentTwoPT1\bin\Debug\assigmentTwoPT1.exe"

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

int main()
{
    void getNumberCube(double x){
    double cube=x*x*x ;
    printf("the cube of %lf is : %lf \n",x,cube);
    }

    double number;
    printf("please enter the number");
    scanf("%lf",&number);
    getNumberCube(number);
    return 0;
}

```

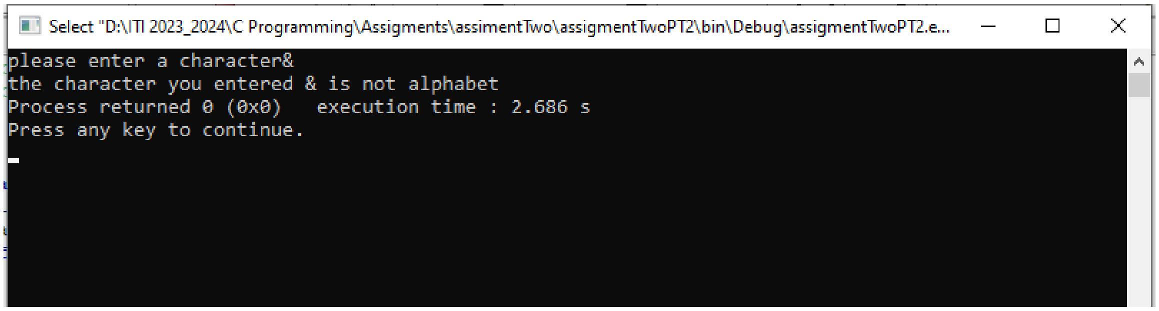
```
please enter the number3
the cube of 3.000000 is : 27.000000

Process returned 0 (0x0)   execution time : 1.577 s
Press any key to continue.
```

(2) Write a C Function that takes one character and checks if it alphabet or not.


main.c ✕
main.c ✕
main.c ✕
main.c ✕
main.c ✕
main.c ✕
main.c ✕
main.c ✕
"D:\ITI 2023_2024\C Programming\Assigments\assimentTwo\assigmentTwoPT2\bin\Debug\assigmentTwoPT2.exe"

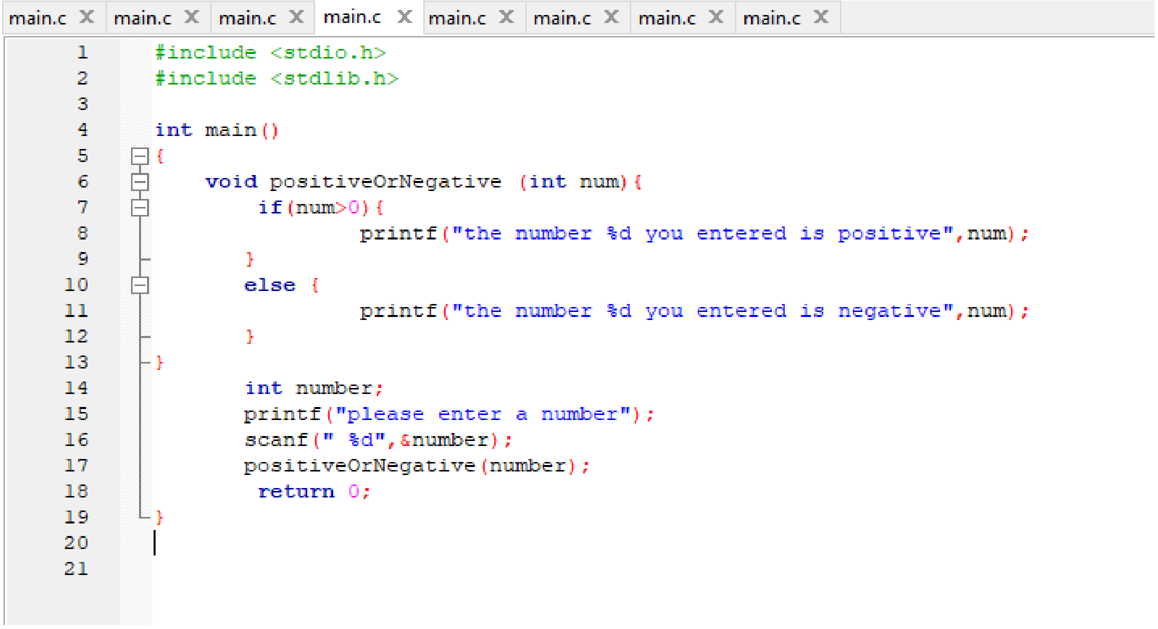
```c
#include <stdio.h>
#include <stdlib.h>

int main()
{
    void isAlphabet(char Character){
        if(Character>='a'&&Character<='z'||Character>='A'&&Character<='Z'){
                printf("the character you entered %c is alphabet",Character);
       }
       else {
                printf("the character you entered %c is not alphabet",Character);
       }}
       char character;
       printf("please enter a character");
       scanf(" %C",&character);
       isAlphabet(character);

        return 0;
}

```

```
please enter a character a
the character you entered a is alphabet
Process returned 0 (0x0)   execution time : 1.578 s
Press any key to continue.
```

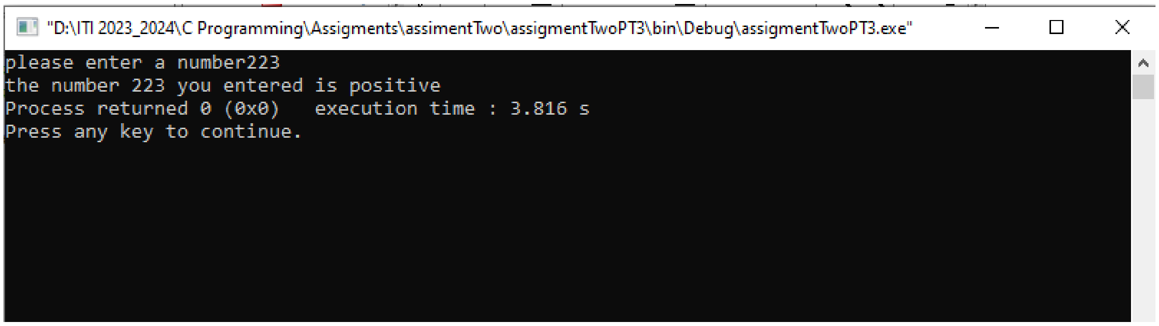

Select "D:\ITI 2023_2024\C Programming\Assigments\assimentTwo\assigmentTwoPT2\bin\Debug\assigmentTwoPT2.e...

```
please enter a character&
the character you entered & is not alphabet
Process returned 0 (0x0)   execution time : 2.686 s
Press any key to continue.
```

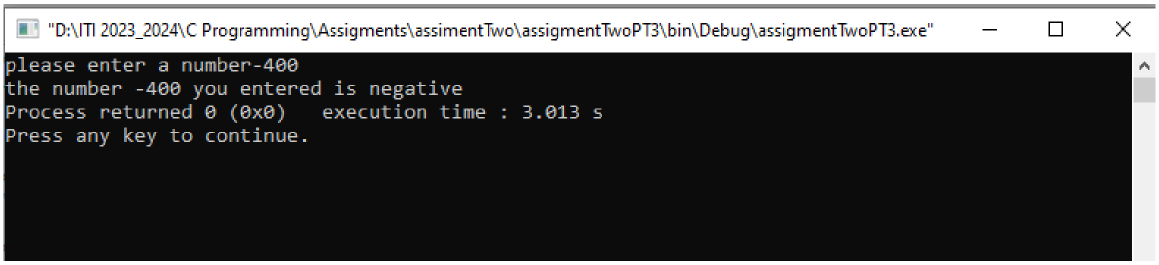
(3) Write a C Function that check if the number if positive or negative.



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

int main()
{
    void positiveOrNegative (int num){
        if(num>0){
                printf("the number %d you entered is positive",num);
       }
       else {
                printf("the number %d you entered is negative",num);
       }
}
       int number;
       printf("please enter a number");
       scanf(" %d",&number);
       positiveOrNegative(number);
        return 0;
}

```

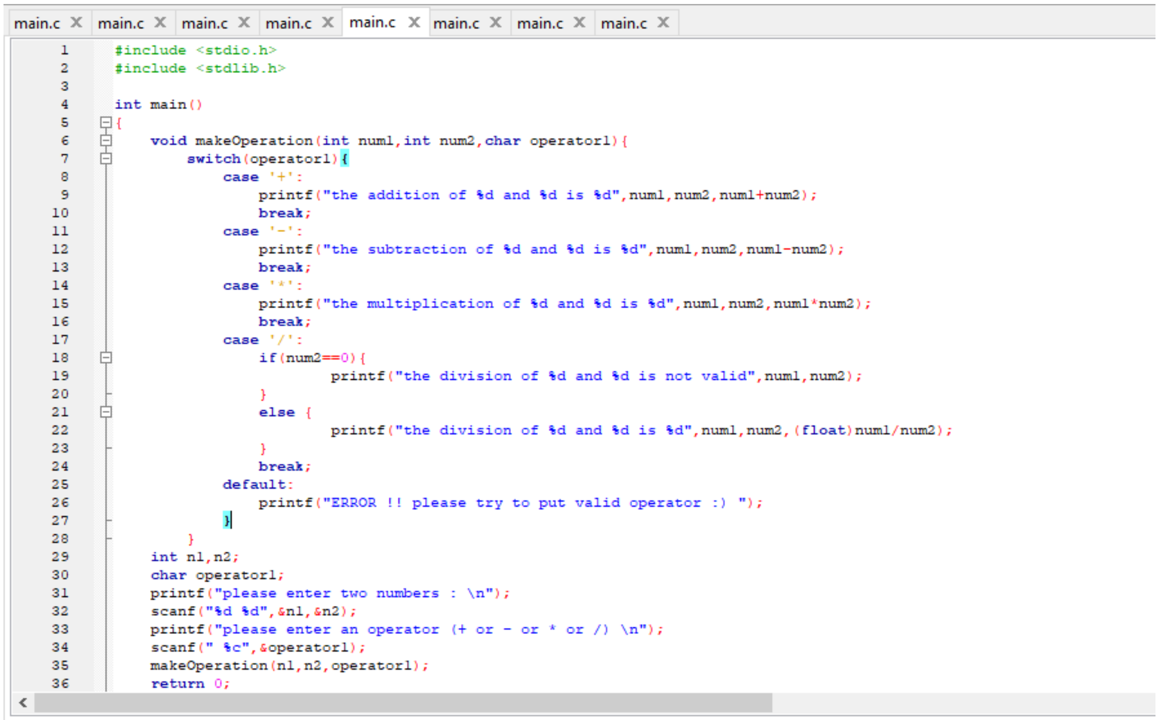

"D:\ITI 2023_2024\C Programming\Assigments\assimentTwo\assigmentTwoPT3\bin\Debug\assigmentTwoPT3.exe"

```
please enter a number223
the number 223 you entered is positive
Process returned 0 (0x0)   execution time : 3.816 s
Press any key to continue.
```

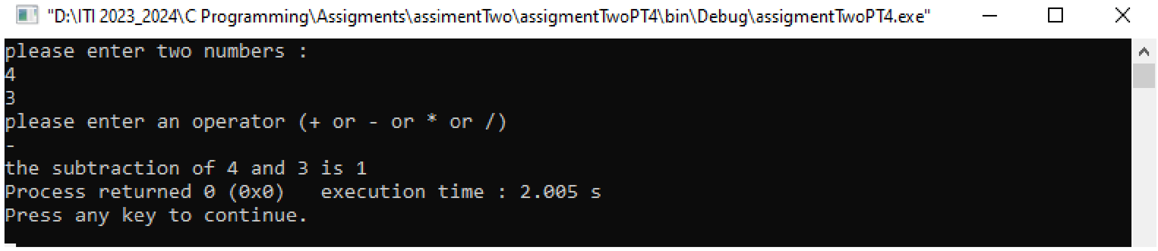

"D:\ITI 2023_2024\C Programming\Assigments\assimentTwo\assigmentTwoPT3\bin\Debug\assigmentTwoPT3.exe"

```
please enter a number-400
the number -400 you entered is negative
Process returned 0 (0x0)   execution time : 3.013 s
Press any key to continue.
```

(4) Write a C Function that return the addition or subtraction or multiplication or division for two numbers. The function should take the required operation and two numbers as arguments. It also should check that the input operation is one of those operation that mentioned before and if not it should return error. The function should be implemented using switch case.

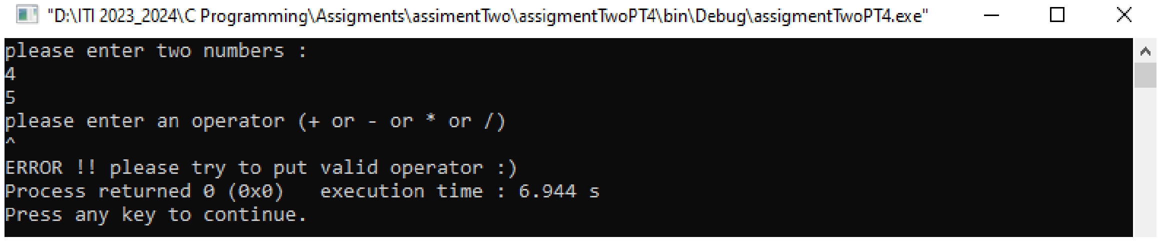


```c
#include <stdio.h>
#include <stdlib.h>

int main()
{
    void makeOperation(int num1,int num2,char operator1){
        switch(operator1){
            case '+':
                printf("the addition of %d and %d is %d",num1,num2,num1+num2);
                break;
            case '-':
                printf("the subtraction of %d and %d is %d",num1,num2,num1-num2);
                break;
            case '*':
                printf("the multiplication of %d and %d is %d",num1,num2,num1*num2);
                break;
            case '/':
                if(num2==0){
                        printf("the division of %d and %d is not valid",num1,num2);
                }
                else {
                        printf("the division of %d and %d is %d",num1,num2,(float)num1/num2);
                }
                break;
            default:
                printf("ERROR !! please try to put valid operator :) ");
            }
        }
    int n1,n2;
    char operator1;
    printf("please enter two numbers : \n");
    scanf("%d %d",&n1,&n2);
    printf("please enter an operator (+ or - or * or /) \n");
    scanf(" %c",&operator1);
    makeOperation(n1,n2,operator1);
    return 0;
```



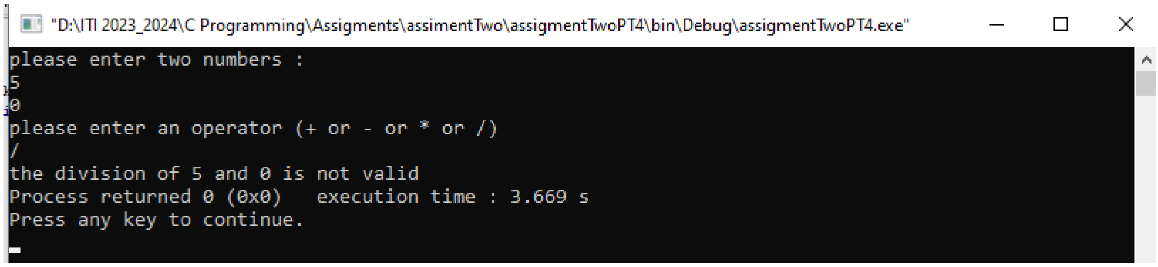
"D:\ITI 2023_2024\C Programming\Assigments\assimentTwo\assigmentTwoPT4\bin\Debug\assigmentTwoPT4.exe"

```
please enter two numbers :
4
3
please enter an operator (+ or - or * or /)
-
the subtraction of 4 and 3 is 1
Process returned 0 (0x0)   execution time : 2.005 s
Press any key to continue.
```



"D:\ITI 2023_2024\C Programming\Assigments\assimentTwo\assigmentTwoPT4\bin\Debug\assigmentTwoPT4.exe"

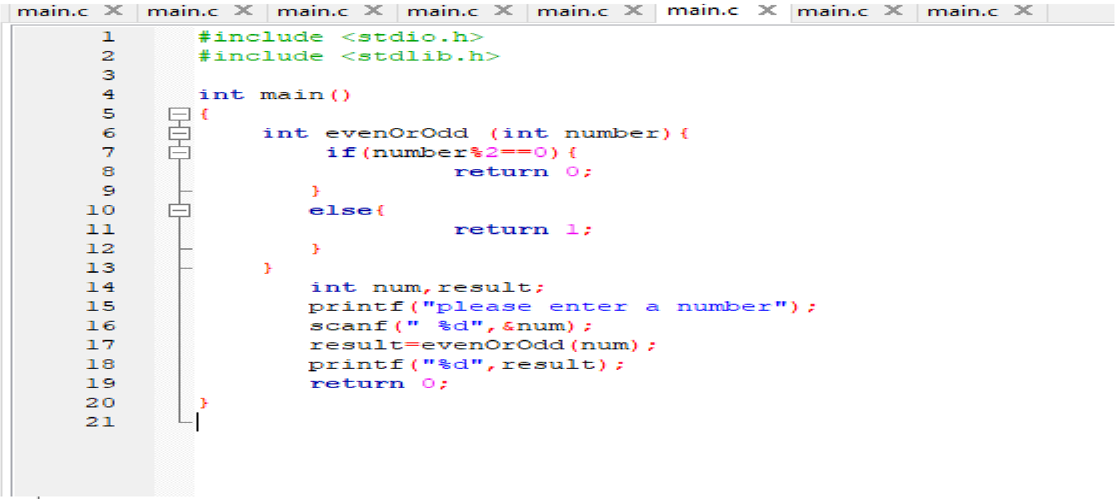
```
please enter two numbers :
4
5
please enter an operator (+ or - or * or /)
^
ERROR !! please try to put valid operator :)
Process returned 0 (0x0)   execution time : 6.944 s
Press any key to continue.
```



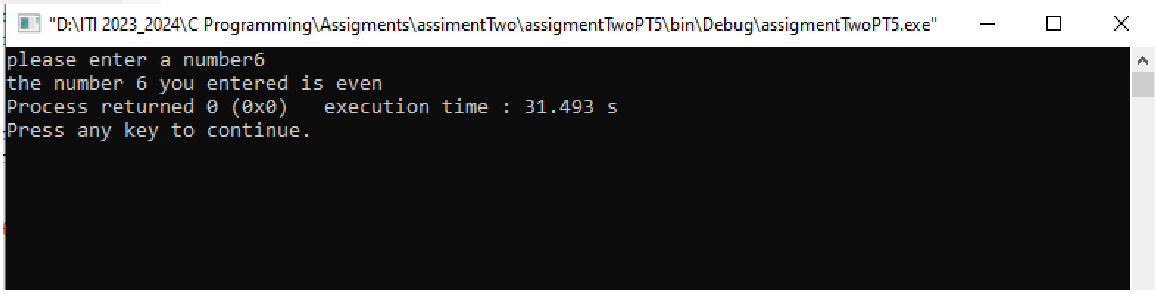
"D:\ITI 2023_2024\C Programming\Assigments\assimentTwo\assigmentTwoPT4\bin\Debug\assigmentTwoPT4.exe"

```
please enter two numbers :
5
0
please enter an operator (+ or - or * or /)
/
the division of 5 and 0 is not valid
Process returned 0 (0x0)   execution time : 3.669 s
Press any key to continue.
```

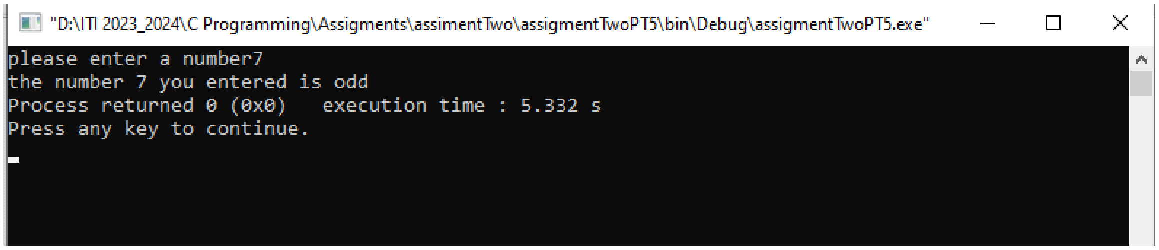
(5) Write a C function to check if the input is an even number or an odd number, if even number return 0 if odd number return 1. Example: Input 7, Output = 1 (Odd) Input 6, Output = 0 (Even)



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

int main()
{
    int evenOrOdd (int number){
        if(number%2==0){
                return 0;
       }
       else{
                return 1;
       }
    }
       int num,result;
       printf("please enter a number");
       scanf(" %d",&num);
       result=evenOrOdd(num);
       printf("%d",result);
       return 0;
}
```

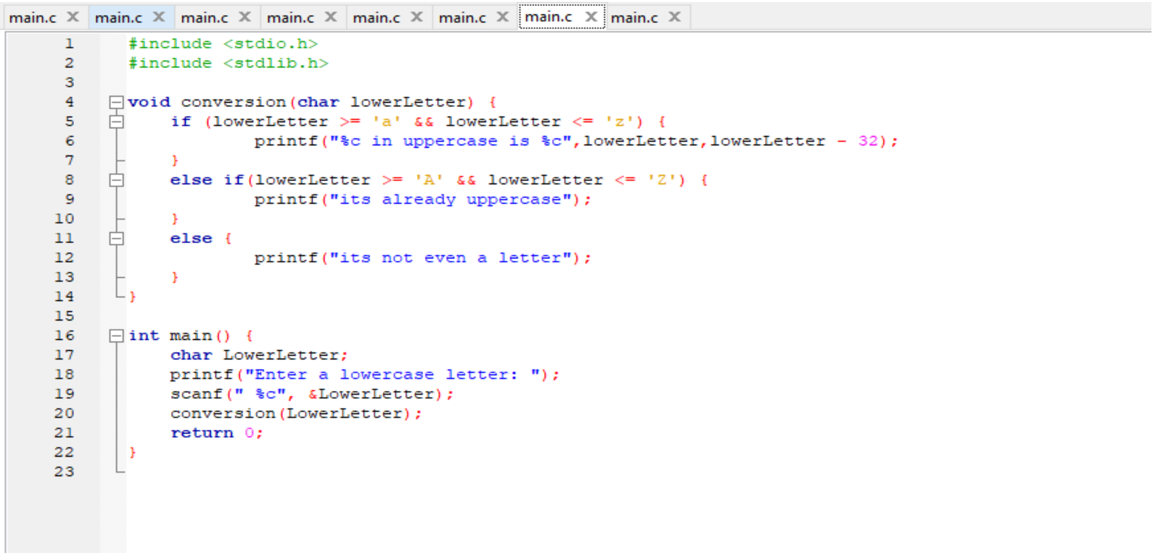

"D:\ITI 2023_2024\C Programming\Assigments\assimentTwo\assigmentTwoPT5\bin\Debug\assigmentTwoPT5.exe"

```
please enter a number6
the number 6 you entered is even
Process returned 0 (0x0)   execution time : 31.493 s
Press any key to continue.
```

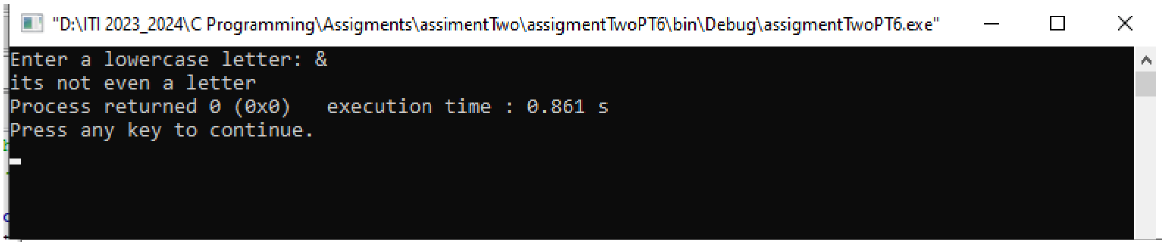

"D:\ITI 2023_2024\C Programming\Assigments\assimentTwo\assigmentTwoPT5\bin\Debug\assigmentTwoPT5.exe"

```
please enter a number7
the number 7 you entered is odd
Process returned 0 (0x0)   execution time : 5.332 s
Press any key to continue.
```

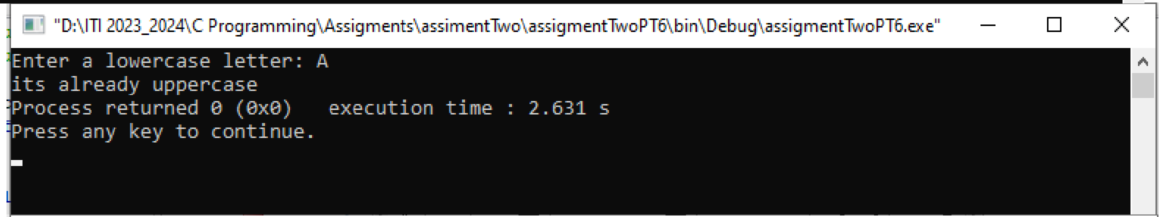
(6) Write C Function that converts the any letter from lowercase to uppercase.



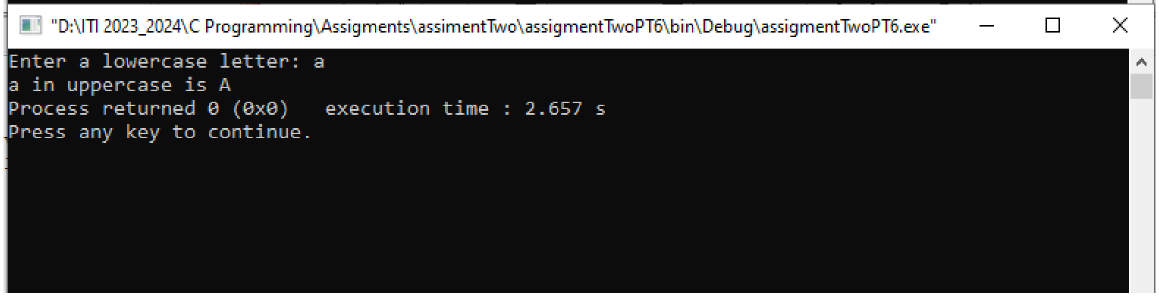
```c
#include <stdio.h>
#include <stdlib.h>

void conversion(char lowerLetter) {
    if (lowerLetter >= 'a' && lowerLetter <= 'z') {
        printf("%c in uppercase is %c",lowerLetter,lowerLetter - 32);
    }
    else if(lowerLetter >= 'A' && lowerLetter <= 'Z') {
        printf("its already uppercase");
    }
    else {
        printf("its not even a letter");
    }
}

int main() {
    char LowerLetter;
    printf("Enter a lowercase letter: ");
    scanf(" %c", &LowerLetter);
    conversion(LowerLetter);
    return 0;
}
```


"D:\ITI 2023_2024\C Programming\Assigments\assimentTwo\assigmentTwoPT6\bin\Debug\assigmentTwoPT6.exe"

```
Enter a lowercase letter: &
its not even a letter
Process returned 0 (0x0)   execution time : 0.861 s
Press any key to continue.
```

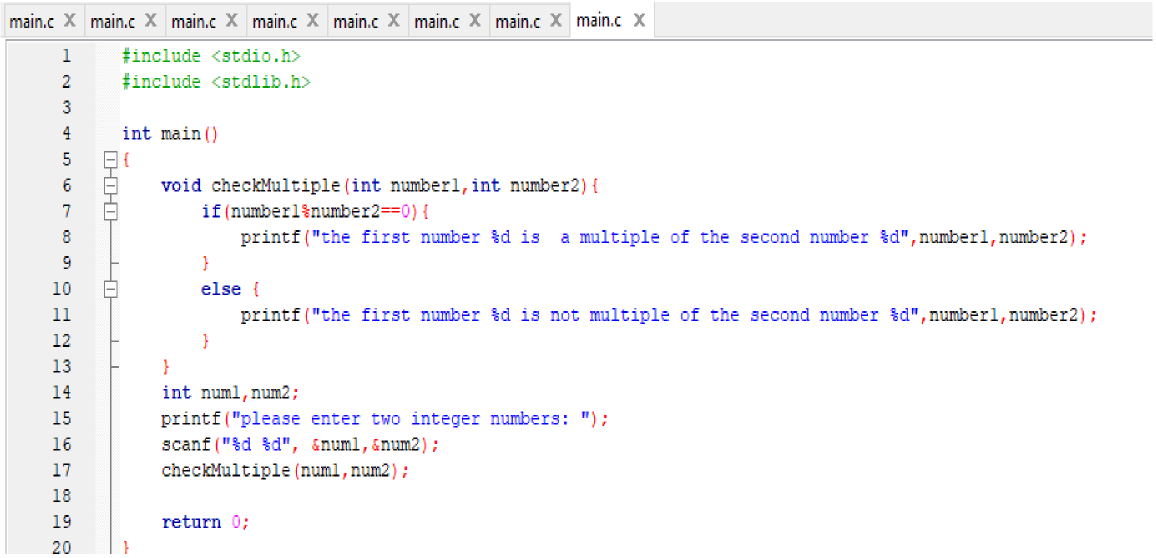

"D:\ITI 2023_2024\C Programming\Assigments\assimentTwo\assigmentTwoPT6\bin\Debug\assigmentTwoPT6.exe"

```
Enter a lowercase letter: A
its already uppercase
Process returned 0 (0x0)   execution time : 2.631 s
Press any key to continue.
```

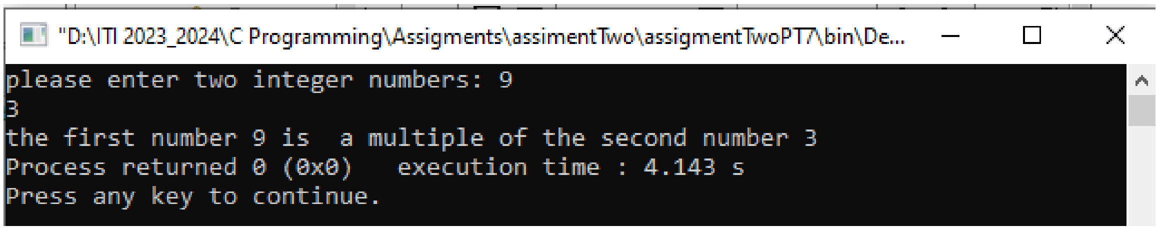

"D:\ITI 2023_2024\C Programming\Assigments\assimentTwo\assigmentTwoPT6\bin\Debug\assigmentTwoPT6.exe"

```
Enter a lowercase letter: a
a in uppercase is A
Process returned 0 (0x0)   execution time : 2.657 s
Press any key to continue.
```

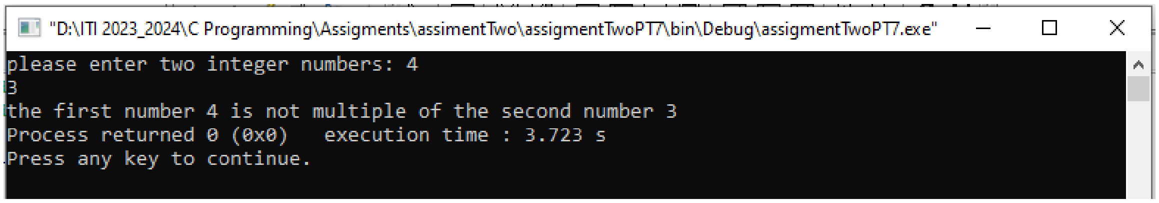
(7) Write a C Function that reads two integers and checks if the first is multiple of the second.


main.c X main.c X main.c X main.c X main.c X main.c X main.c X main.c X

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

int main()
{
    void checkMultiple(int number1,int number2){
        if(number1%number2==0){
            printf("the first number %d is  a multiple of the second number %d",number1,number2);
        }
        else {
            printf("the first number %d is not multiple of the second number %d",number1,number2);
        }
    }
    int num1,num2;
    printf("please enter two integer numbers: ");
    scanf("%d %d", &num1,&num2);
    checkMultiple(num1,num2);

    return 0;
}
```

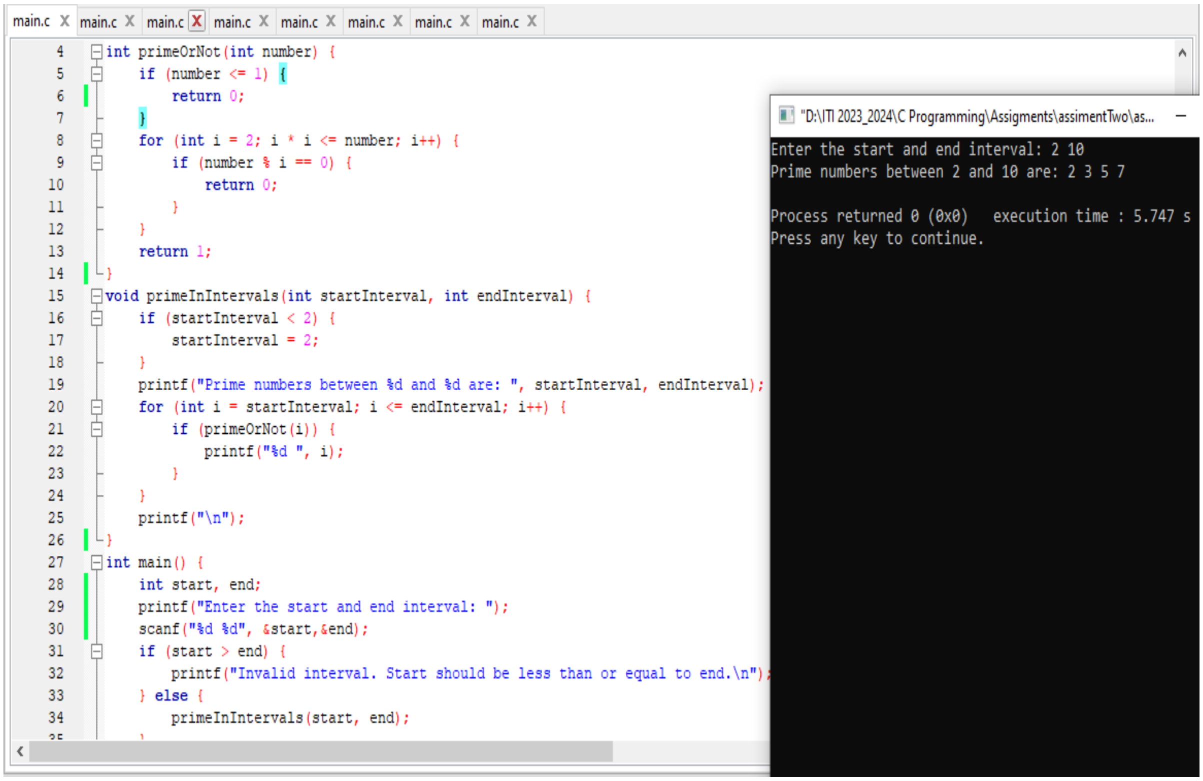

"D:\ITI 2023_2024\C Programming\Assigments\assimentTwo\assigmentTwoPT7\bin\De...

```
please enter two integer numbers: 9
3
the first number 9 is  a multiple of the second number 3
Process returned 0 (0x0)   execution time : 4.143 s
Press any key to continue.
```


"D:\ITI 2023_2024\C Programming\Assigments\assimentTwo\assigmentTwoPT7\bin\Debug\assigmentTwoPT7.exe"

```
please enter two integer numbers: 4
3
the first number 4 is not multiple of the second number 3
Process returned 0 (0x0)   execution time : 3.723 s
Press any key to continue.
```

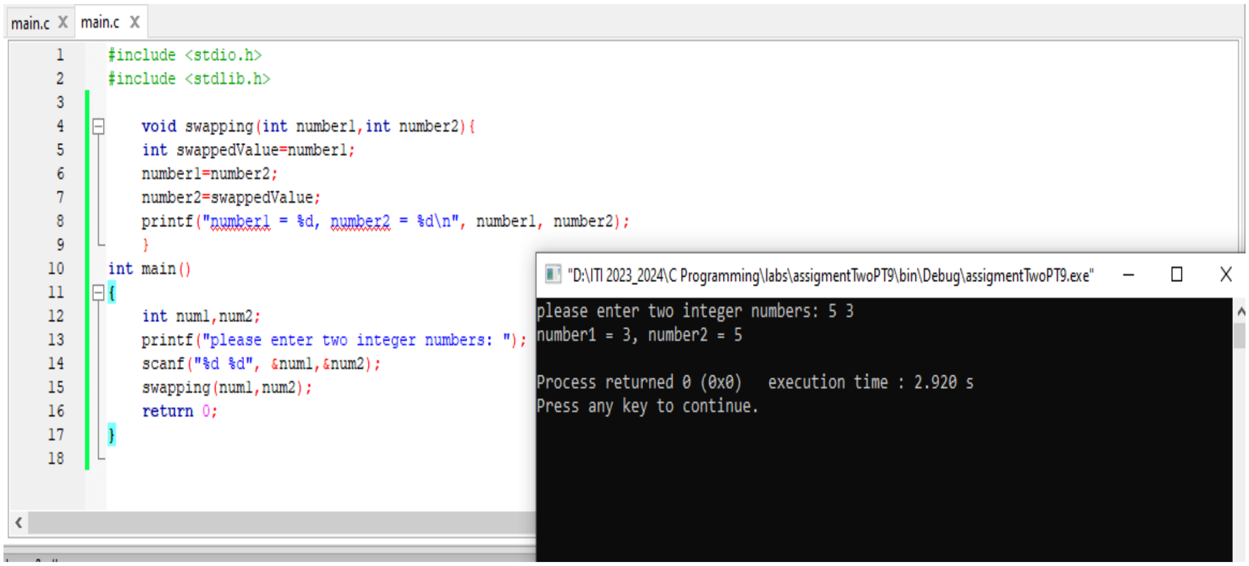
(8)Write a C Function that display Prime Numbers between Intervals (two numbers) by Making Function.



```c
int primeOrNot(int number) {
    if (number <= 1) {
        return 0;
    }
    for (int i = 2; i * i <= number; i++) {
        if (number % i == 0) {
            return 0;
        }
    }
    return 1;
}
void primeInIntervals(int startInterval, int endInterval) {
    if (startInterval < 2) {
        startInterval = 2;
    }
    printf("Prime numbers between %d and %d are: ", startInterval, endInterval);
    for (int i = startInterval; i <= endInterval; i++) {
        if (primeOrNot(i)) {
            printf("%d ", i);
        }
    }
    printf("\n");
}
int main() {
    int start, end;
    printf("Enter the start and end interval: ");
    scanf("%d %d", &start,&end);
    if (start > end) {
        printf("Invalid interval. Start should be less than or equal to end.\n");
    } else {
        primeInIntervals(start, end);
```

```
Enter the start and end interval: 2 10
Prime numbers between 2 and 10 are: 2 3 5 7

Process returned 0 (0x0)   execution time : 5.747 s
Press any key to continue.
```

(9)Write a C Function that swaps the value of two integer numbers.


main.c X
main.c X
"D:\ITI 2023_2024\C Programming\labs\assigmentTwoPT9\bin\Debug\assigmentTwoPT9.exe"

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

    void swapping(int number1,int number2){
    int swappedValue=number1;
    number1=number2;
    number2=swappedValue;
    printf("number1 = %d, number2 = %d\n", number1, number2);
    }
int main()
{

    int num1,num2;
    printf("please enter two integer numbers: ");
    scanf("%d %d", &num1,&num2);
    swapping(num1,num2);
    return 0;
}
```

```
please enter two integer numbers: 5 3
number1 = 3, number2 = 5

Process returned 0 (0x0)   execution time : 2.920 s
Press any key to continue.
```

(10) You are designing a poster which prints out numbers with a unique style applied to each of them. The styling is based on the number of closed paths or holes present in a giver number. The number of holes that each of the digits from 0 to 9 have are equal to the number of closed paths in the digit. Their values are:

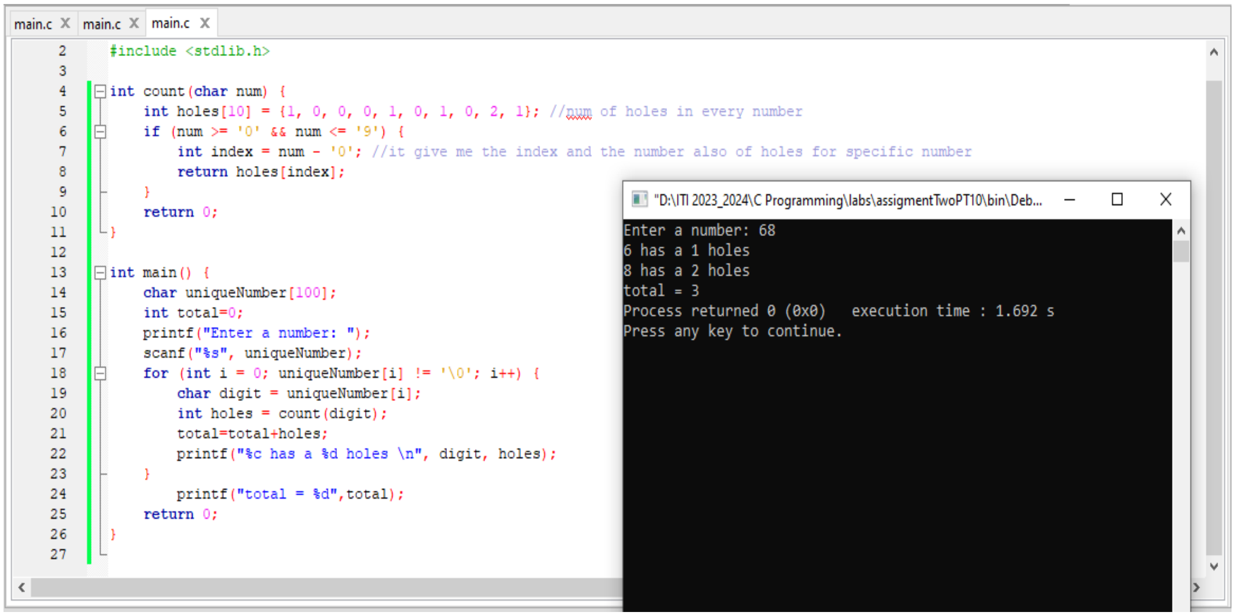
 1, 2, 3, 5 and 7 = 0 holes.

 0, 4, 6, and 9 = 1 hole.

 8 = 2 holes.

Given a number, you must determine the sum of the number of holes for all of its digits. For example, the number 819 has 3 holes.

Function Description Complete the function countHoles. The function must return an integer denoting the total number of holes in num.

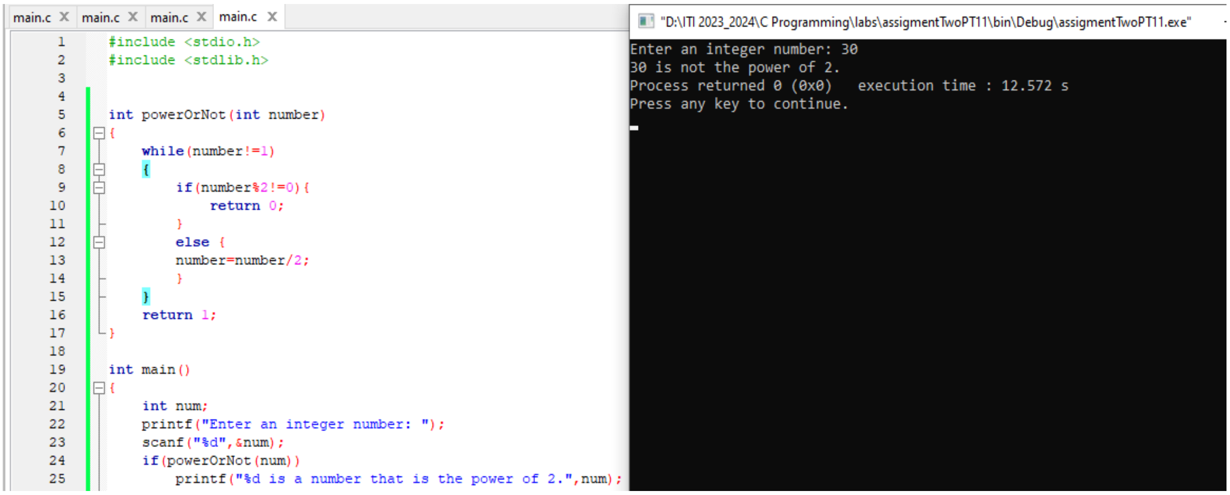

main.c X
main.c X
main.c X
"D:\ITI 2023_2024\C Programming\labs\assigmentTwoPT10\bin\Deb...

```c
#include <stdlib.h>

int count(char num) {
    int holes[10] = {1, 0, 0, 0, 1, 0, 1, 0, 2, 1}; //num of holes in every number
    if (num >= '0' && num <= '9') {
        int index = num - '0'; //it give me the index and the number also of holes for specific number
        return holes[index];
    }
    return 0;
}

int main() {
    char uniqueNumber[100];
    int total=0;
    printf("Enter a number: ");
    scanf("%s", uniqueNumber);
    for (int i = 0; uniqueNumber[i] != '\0'; i++) {
        char digit = uniqueNumber[i];
        int holes = count(digit);
        total=total+holes;
        printf("%c has a %d holes \n", digit, holes);
    }
        printf("total = %d",total);
    return 0;
}
```

```
Enter a number: 68
6 has a 1 holes
8 has a 2 holes
total = 3
Process returned 0 (0x0)   execution time : 1.692 s
Press any key to continue.
```

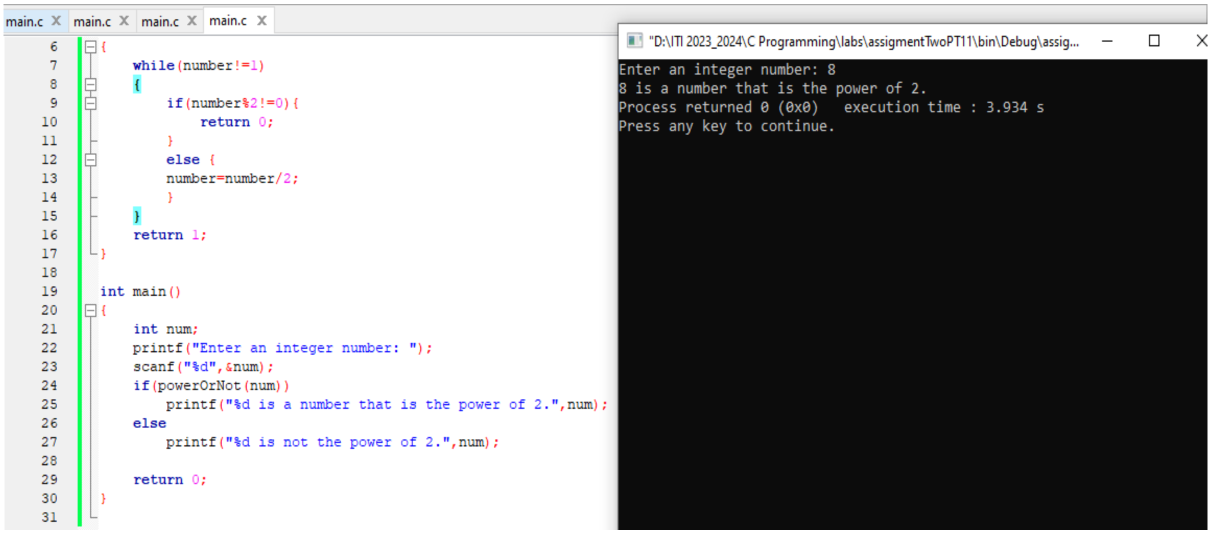
(11) Write a C function that returns 1 if the input number is a power of 2 and return 0 if the input number is power of 2.

For example: 0, 2 and 16 are power of 2. 1, 10 and 30 are not power of 2.


main.c
main.c
main.c
main.c
"D:\ITI 2023_2024\C Programming\labs\assigmentTwoPT11\bin\Debug\assigmentTwoPT11.exe"

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


int powerOrNot(int number)
{
    while(number!=1)
    {
        if(number%2!=0){
            return 0;
        }
        else {
        number=number/2;
        }
    }
    return 1;
}

int main()
{
    int num;
    printf("Enter an integer number: ");
    scanf("%d",&num);
    if(powerOrNot(num))
        printf("%d is a number that is the power of 2.",num);
```

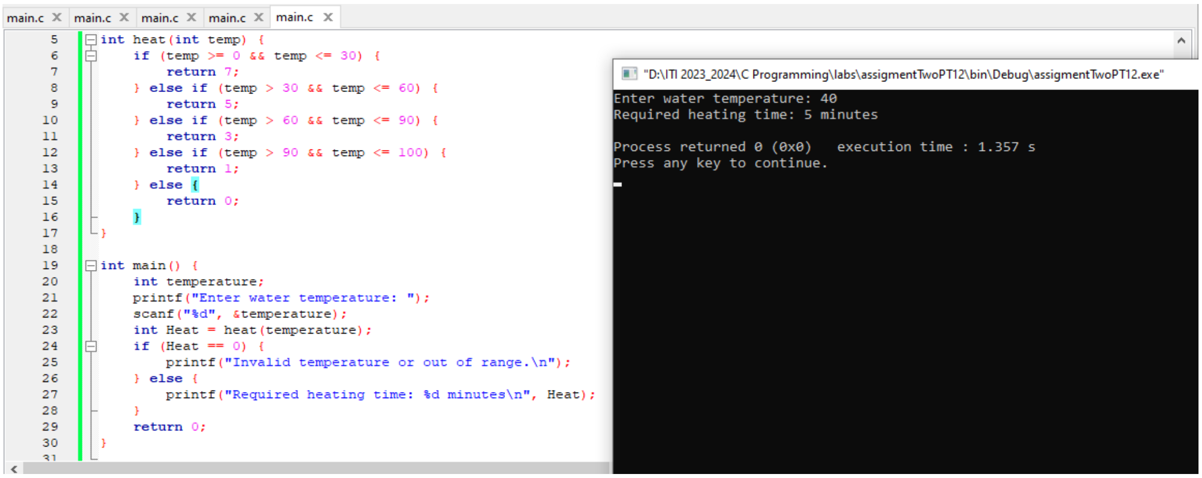
```
Enter an integer number: 30
30 is not the power of 2.
Process returned 0 (0x0)   execution time : 12.572 s
Press any key to continue.
```


main.c
main.c
main.c
main.c
"D:\ITI 2023_2024\C Programming\labs\assigmentTwoPT11\bin\Debug\assig...

```c
{
    while(number!=1)
    {
        if(number%2!=0){
            return 0;
        }
        else {
        number=number/2;
        }
    }
    return 1;
}

int main()
{
    int num;
    printf("Enter an integer number: ");
    scanf("%d",&num);
    if(powerOrNot(num))
        printf("%d is a number that is the power of 2.",num);
    else
        printf("%d is not the power of 2.",num);

    return 0;
}
```

```
Enter an integer number: 8
8 is a number that is the power of 2.
Process returned 0 (0x0)   execution time : 3.934 s
Press any key to continue.
```

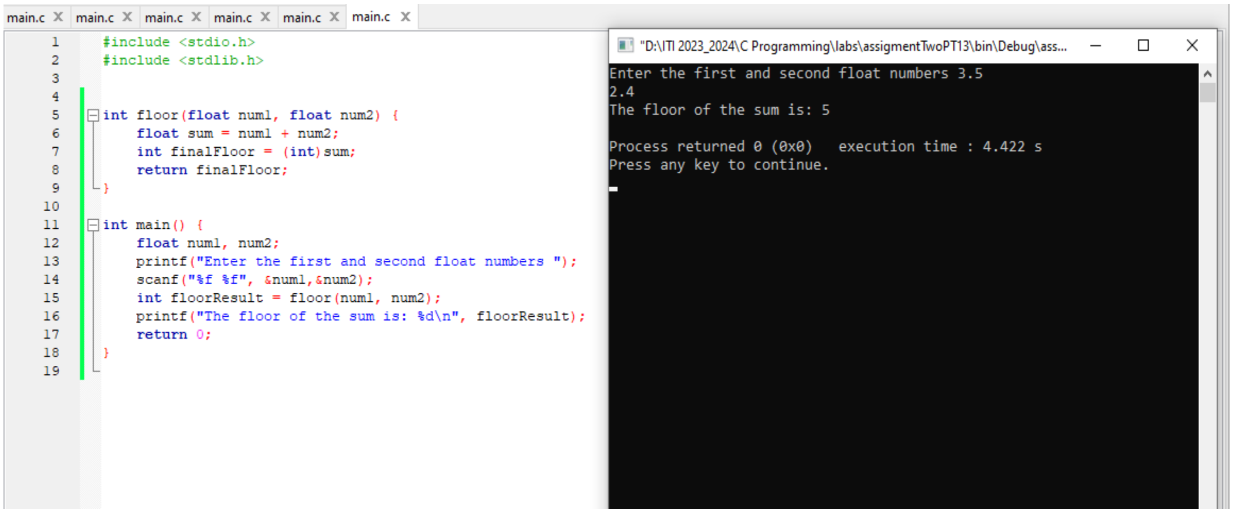
(12) Write a C function that calculates the required heater activation time according to the input temperature of water. - if input temperature is from 0 to 30, then required heating time = 7 mins. - if input temperature is from 30 to 60, then required heating time = 5 mins. - if input temperature is from 60 to 90, then required heating time = 3 mins. - if input temperature is more than 90, then required heating time = 1 mins. - if temperature is invalid (more than 100), return 0 Example: Input = 10 ⍰ output = 7 Input = 35 ⍰ output = 5


main.c ×
main.c ×
main.c ×
main.c ×
main.c ×
"D:\ITI 2023_2024\C Programming\labs\assigmentTwoPT12\bin\Debug\assigmentTwoPT12.exe"

```c
int heat(int temp) {
    if (temp >= 0 && temp <= 30) {
        return 7;
    } else if (temp > 30 && temp <= 60) {
        return 5;
    } else if (temp > 60 && temp <= 90) {
        return 3;
    } else if (temp > 90 && temp <= 100) {
        return 1;
    } else {
        return 0;
    }
}

int main() {
    int temperature;
    printf("Enter water temperature: ");
    scanf("%d", &temperature);
    int Heat = heat(temperature);
    if (Heat == 0) {
        printf("Invalid temperature or out of range.\n");
    } else {
        printf("Required heating time: %d minutes\n", Heat);
    }
    return 0;
}
```

```
Enter water temperature: 40
Required heating time: 5 minutes

Process returned 0 (0x0)   execution time : 1.357 s
Press any key to continue.
```

(13) In this challenge write a function to add two floating numbers. Determine the integer floor of the sum. The floor is the truncated float value, anything after the decimal point is dropped. For instance floor(1.1+3.05)=floor(4.15)=4

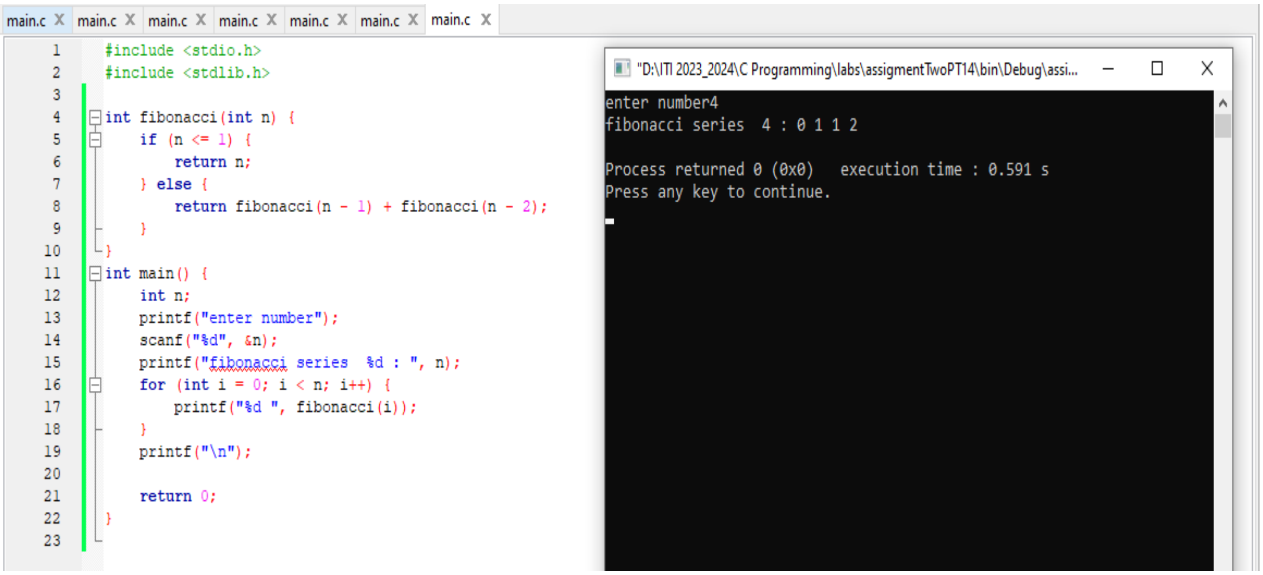

main.c ×
main.c ×
main.c ×
main.c ×
main.c ×
main.c ×
"D:\ITI 2023_2024\C Programming\labs\assigmentTwoPT13\bin\Debug\ass...

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


int floor(float num1, float num2) {
    float sum = num1 + num2;
    int finalFloor = (int)sum;
    return finalFloor;
}

int main() {
    float num1, num2;
    printf("Enter the first and second float numbers ");
    scanf("%f %f", &num1,&num2);
    int floorResult = floor(num1, num2);
    printf("The floor of the sum is: %d\n", floorResult);
    return 0;
}
```

```
Enter the first and second float numbers 3.5
2.4
The floor of the sum is: 5

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

(14) Write a C Function that calculate the Fibonacci series using recursive method.

The Fibonacci Series : 0,1,1,2,3,5,8,13,21,...


main.c X
main.c X
main.c X
main.c X
main.c X
main.c X
main.c X
"D:\ITI 2023_2024\C Programming\labs\assigmentTwoPT14\bin\Debug\assi...

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

int fibonacci(int n) {
    if (n <= 1) {
        return n;
    } else {
        return fibonacci(n - 1) + fibonacci(n - 2);
    }
}
int main() {
    int n;
    printf("enter number");
    scanf("%d", &n);
    printf("fibonacci series  %d : ", n);
    for (int i = 0; i < n; i++) {
        printf("%d ", fibonacci(i));
    }
    printf("\n");

    return 0;
}
```

```
enter number4
fibonacci series  4 : 0 1 1 2

Process returned 0 (0x0)   execution time : 0.591 s
Press any key to continue.
```