Object Oriented Programming

Lab Manual 2

Introduction

In this course, we will learn a new programming language that is known as C# (C Sharp). Now that you have successfully installed and set up the Visual Studio on your computer and basic operations in this programming language, now let's learn about implementing conditional and repetition structures operations in C#.

Let's do some coding.

Conditional Operation in C#

We have learned in the previous manual about basic operations in C#. In this section, we will implement the conditional **IF-Else** structure.

```
if (condition)
{
        // statements
}
else
{
        // statements
}
```

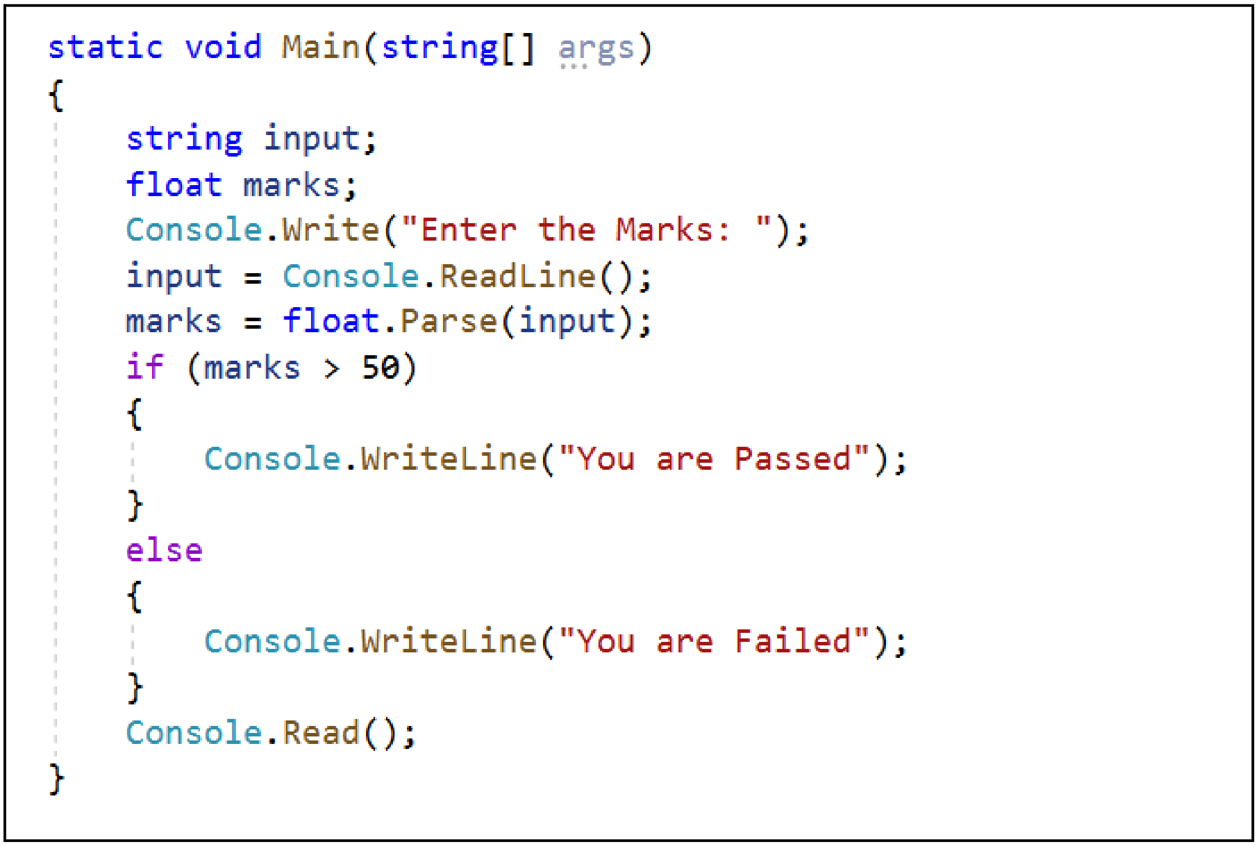
In C#, this is the syntax of IF-Else Conditional structure.

Task: To understand this concept, try writing a program that prints “You are passed” if the user enters a number greater than 50 otherwise it should display “You are Failed”.

Solution:

Write the following code into the main function of the code and execute the program by clicking on the start button.

Code:



```
static void Main(string[] args)
{
    string input;
    float marks;
    Console.Write("Enter the Marks: ");
    input = Console.ReadLine();
    marks = float.Parse(input);
    if (marks > 50)
    {
        Console.WriteLine("You are Passed");
    }
    else
    {
        Console.WriteLine("You are Failed");
    }
    Console.Read();
}
```

Now, you can check that this program will perform the required functionality by printing "**You are Passed**" if the user enters a number greater than 50. However, it prints "**You are Failed**" if the input is less than 50.

Congratulations, you have learned something new Again !!!!

Let's Continue the fun coding Ride.

Loops in C#

Similarly, the repetition structures are the same as in c++, just like we have learned about the Conditional Structures in C#.

Counting Loop

The syntax of **for loop** is mentioned below.

Syntax:

```
for (initialization; condition; increment)
{
        // statements
```

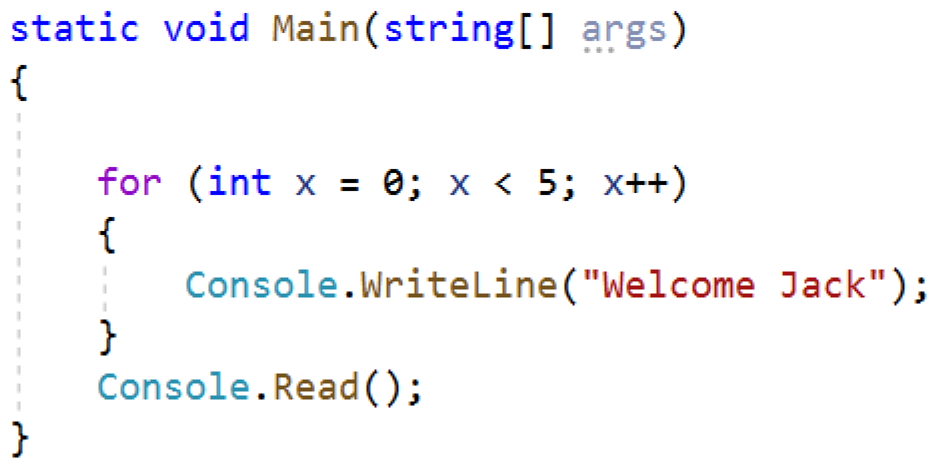
```
    }
```

Task: To understand this concept, try writing a program that prints “Welcome Jack” 5 times on the computer screen. .

Solution:

Write the following code into the main function of the code and execute the program by clicking on the start button.

Code:



```
static void Main(string[] args)
{

    for (int x = 0; x < 5; x++)
    {
        Console.WriteLine("Welcome Jack");
    }
    Console.Read();
}
```

This program gives you a classical example of a counting loop where it prints five times the same statement of “**Console.WriteLine(“Welcome Jack”)**” by using the **FOR Loop**.

Conditional Loop

The syntax of the **while loop** is as follows.

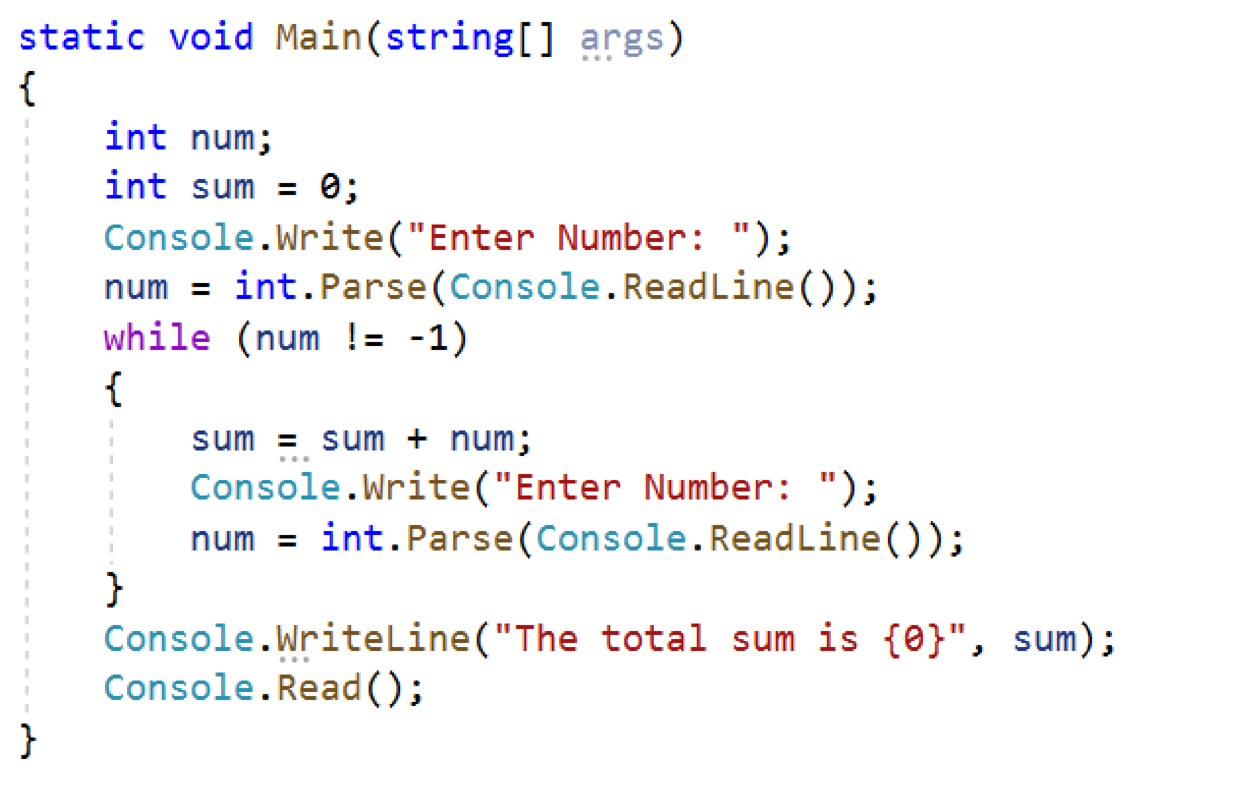
Syntax:

```
while (condition)
{
        // statements
}
```

Task: To understand this concept, try writing a program that keeps taking a number from the user until he enters -1 and shows the sum of all inputted numbers.

Solution:
Write the following code into the main function of the code and execute the program by clicking on the start button.

Code:



```
static void Main(string[] args)
{
    int num;
    int sum = 0;
    Console.Write("Enter Number: ");
    num = int.Parse(Console.ReadLine());
    while (num != -1)
    {
        sum = sum + num;
        Console.Write("Enter Number: ");
        num = int.Parse(Console.ReadLine());
    }
    Console.WriteLine("The total sum is {0}", sum);
    Console.Read();
}
```

This program will keep taking input from the user until the user enters -1. Finally, at the end of the program, it prints the total sum of all inputted numbers.

Do while Loop

The syntax of the **do-while** loop is mentioned below.

Syntax:

```
do
{
        // statements
}
while (condition);
```

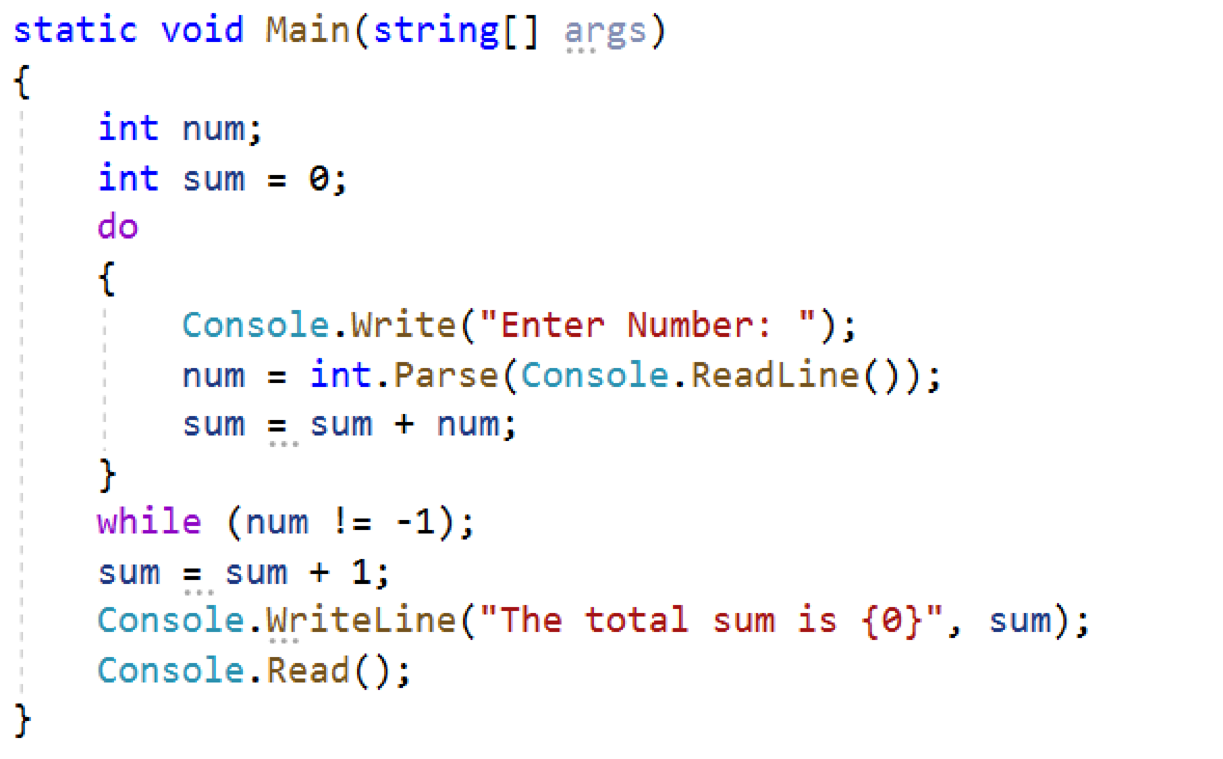
Task: To understand this concept, perform the previous task again but with a do-while loop this time.

Write a program that keeps taking a number from the user until he enters -1 and shows the sum of all inputted numbers.

Solution:

Write the following code into the main function of the code and execute the program by clicking on the start button.

Code:



```
static void Main(string[] args)
{
    int num;
    int sum = 0;
    do
    {
        Console.Write("Enter Number: ");
        num = int.Parse(Console.ReadLine());
        sum = sum + num;
    }
    while (num != -1);
    sum = sum + 1;
    Console.WriteLine("The total sum is {0}", sum);
    Console.Read();
}
```

Execute this code to see that it will keep accepting input until you enter -1 and at the end, it will print the total sum of all the inputted numbers.

Its okay students, you can take a 5-minute break as you have successfully learned about the conditional and repetition structures in C#.

Arrays in C#

The fundamental concept is the same as c++ however there are some minor changes that are defined below.

Syntax:

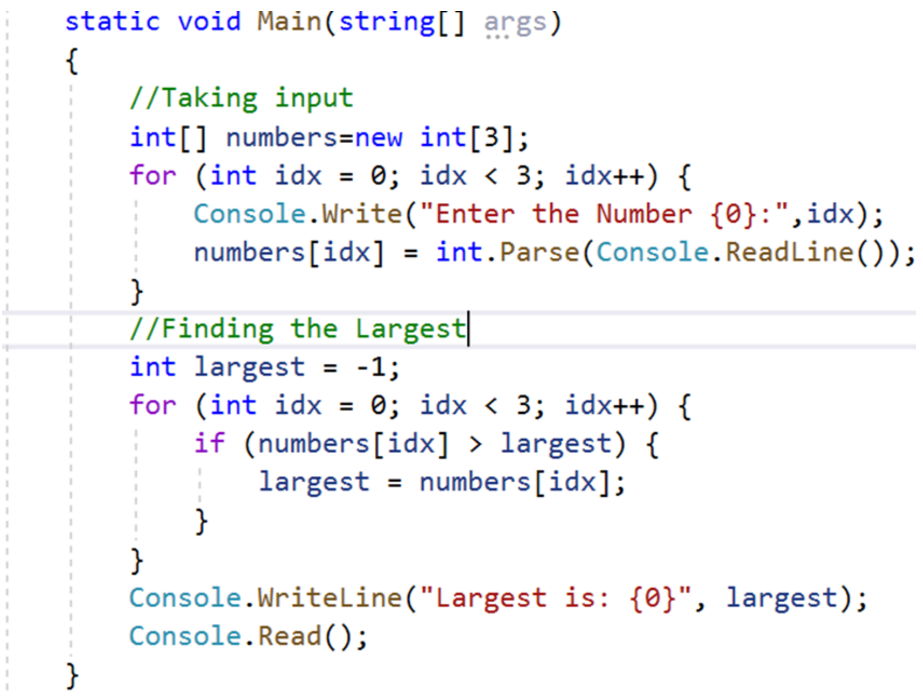
int[] variable_name = new int[5];

Task: To understand this concept, perform the below-mentioned task by using arrays. Write a program that takes 3 numbers as input and prints the largest of them.

Solution:

Write the following code into the main function of the code and execute the program by clicking on the start button.

Code:



```
static void Main(string[] args)
{
    //Taking input
    int[] numbers=new int[3];
    for (int idx = 0; idx < 3; idx++) {
        Console.Write("Enter the Number {0}:",idx);
        numbers[idx] = int.Parse(Console.ReadLine());
    }
    //Finding the Largest
    int largest = -1;
    for (int idx = 0; idx < 3; idx++) {
        if (numbers[idx] > largest) {
            largest = numbers[idx];
        }
    }
    Console.WriteLine("Largest is: {0}", largest);
    Console.Read();
}
```

Take Away Tasks

1. Lilly is N years old. For each birthday she receives a present. For each odd birthday (1, 3, 5, …, n) she receives toys
for each even birthday (2, 4, 6, …, n) she receives money. For her second birthday, she received 10.00 USD, and the amount is increased by 10.00 USD for each following even birthday (2 -> 10, 4 -> 20, 6 -> 30, etc.).

Over the years Lilly has secretly saved her money. Lilly's brother, in the years when she received money, took 1.00 USD from each of the amounts.
Lilly has sold the toys, received over the years, each one for P USD, and added the sum to the amount of saved money.
With the money, she wanted to buy a washing machine for X USD. Write a Function that takes lily’s age, price of washing machine, and price of toy and calculates how much money she has saved and if it is enough to buy a washing machine.

Input:
We read from the console 3 numbers, each on a separate line:
1. Lilly's age – integer in the range of [1 … 77].
2. Price of the washing machine – the number in the range of [1.00 … 10,000.00].
3. Unit price of each toy – integer in the range of [0 … 40].

Output
Print on the console one single line:
If Lilly's money is enough:
“Yes! {N}” – where N is the remaining money after the purchase
If the money is not enough:
“No! {M}” – where M is the insufficiency amount

Input	Output	Comments
10 170.00 6	Yes! 5.00	For the first birthday she gets a toy; 2nd -> 10 USD; 3rd -> toy; 4th -> 10 + 10 = 20 USD; 5th -> toy; 6th -> 20 + 10 = 30 USD; 7th -> toy; 8th -> 30 + 10 = 40 USD; 9th -> toy; 10th -> 40 + 10 = 50 USD. She has saved: 10 + 20 + 30 + 40 + 50 = 150 USD. She sold 5 toys for 6 USD each = 30 USD. Her brother took 1 USD 5 times = 5 USD. Remaining amount: 150 + 30 – 5 = 175 USD. 175 >= 170 (price of the washing machine): she managed to buy it and is left with 175-170 = 5 USD.

21 1570.98 3	No! 997.98	She has saved 550 USD. She has sold 11 toys, 3 USD each = 33 USD. Her brother has taken for 10 years 1 USD each year = 10 USD. Remaining amount: 550 + 33 - 10 = 573 USD. 573 < 1570.98: she did not manage to buy a washing machine. The insufficiency amount is: 1570.98 - 573 = 997.98 USD.

Good Luck and Best Wishes !!

Happy Coding ahead :)