



Bahria University, Islamabad

Department of Software Engineering

Computer Programming Lab

(Fall-2023)

Teacher: Engr. M Waleed Khan

Student : Muhammad Hamza

Enrollment : 01-131232-057

Lab Journal: 05

Date: 29-10-22023

Task No:	Task Wise Marks		Documentation Marks		Total Marks (20)
	Assigned	Obtained	Assigned	Obtained	
1	3		5		
2	3				
3	3				
4	3				
5	3				

Comments:

Signature

Lab No: 05 – Do-While loop

Introduction:

Loops are repetitive structures and we use them if we want to repeat a particular set of statements in a particular way.

Tools Used

Visual studio.

Problem 1:

Write a program to print in the descending order first twenty natural numbers on the computer screen by using “do-while” loop.

Code:

```
#include<iostream>
using namespace std;
int main() {

    int natural_number = 20;

    do
    {
        cout << natural_number;

        natural_number--;
        cout << endl;

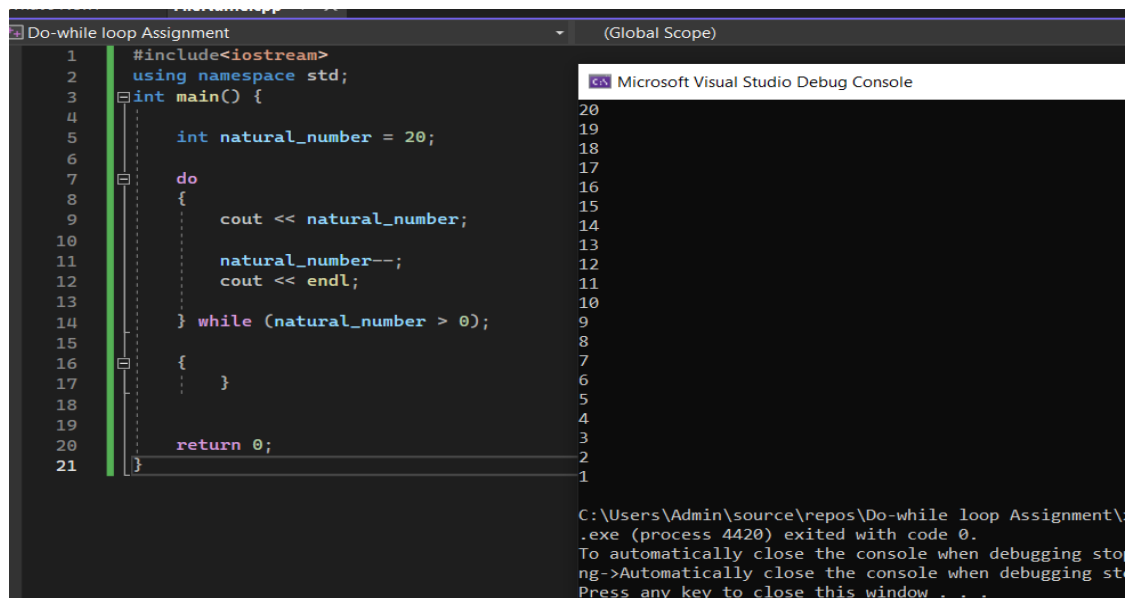
    } while (natural_number > 0);

    {

    }

    return 0;
}
```

Screenshot:



The screenshot displays the Visual Studio IDE with a C++ program titled "Do-while loop Assignment". The code is as follows:

```
1 #include<iostream>
2 using namespace std;
3 int main() {
4
5     int natural_number = 20;
6
7     do
8     {
9         cout << natural_number;
10
11         natural_number--;
12         cout << endl;
13
14     } while (natural_number > 0);
15
16     {
17     }
18
19     return 0;
20 }
21
```

The Microsoft Visual Studio Debug Console on the right shows the output of the program, displaying the numbers 20 through 1 in descending order, each on a new line. The console also shows the process exiting with code 0 and a prompt to press any key to close the window.

Problem 2:

Write a program to compute and print the factorial of the given number using the “do-while” loop.

Code:

```
#include<iostream>
using namespace std;
int main() {

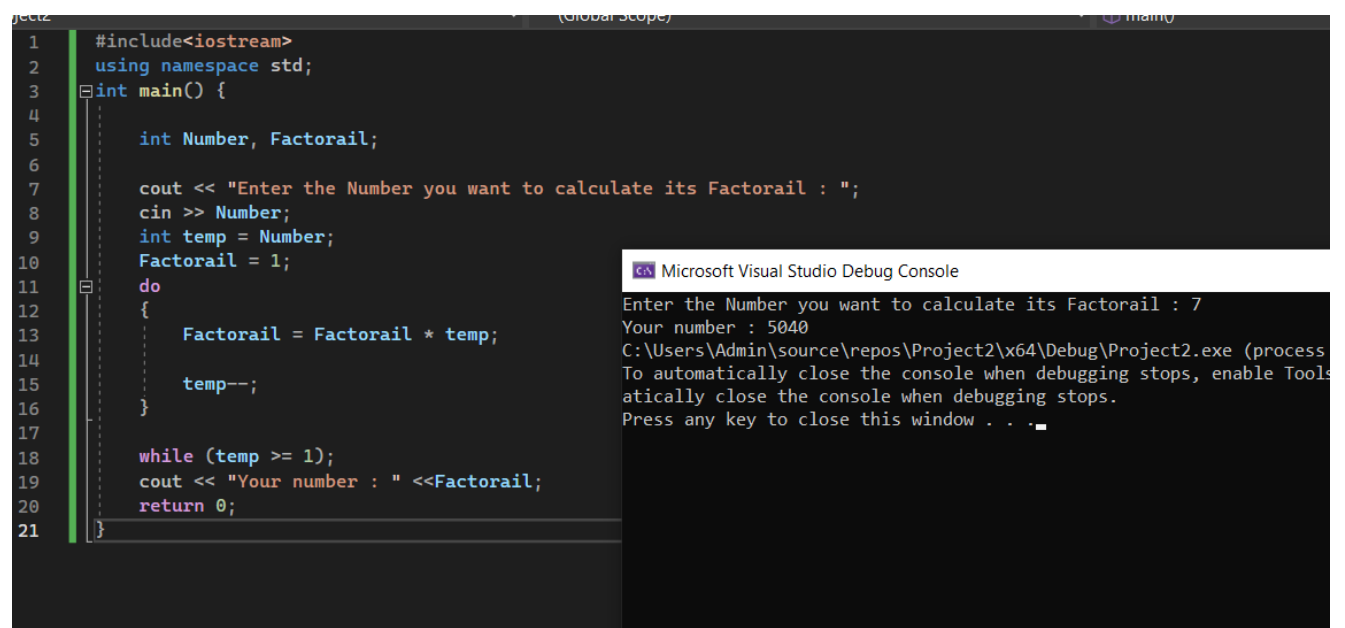
    int Number, Factorail;

    cout << "Enter the Number you want to calculate its Factorail : ";
    cin >> Number;
    int temp = Number;
    Factorail = 1;
    do
    {
        Factorail = Factorail * temp;

        temp--;
    }

    while (temp >= 1);
    cout << "Your number : " <<Factorail;
    return 0;
}
```

Screenshot:



Problem 3:

Write a program to convert the given decimal number into octal number using the “do-while” loop.

Code:

```
#include<iostream>
using namespace std;
int main() {
    int Number, reminder;
    cout << "Enter number : ";
    cin >> Number;

    int octal = 0;
    int value = 1;

    do
    {
        reminder = Number % 8;
        octal = octal + (reminder * value);
        Number = Number / 8;
        value = value * 10;

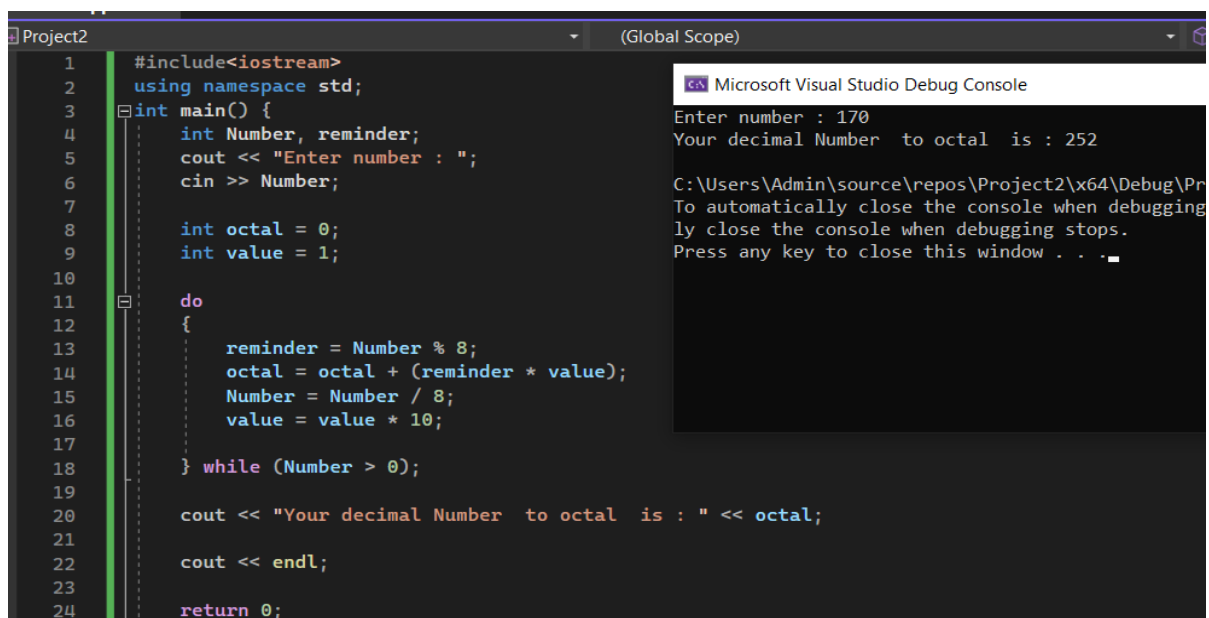
    } while (Number > 0);

    cout << "Your decimal Number to octal is : " << octal;

    cout << endl;

    return 0;
}
```

Screenshot:



The screenshot displays the Visual Studio IDE with a C++ project named 'Project2'. The code editor shows the implementation of a decimal-to-octal conversion program using a do-while loop. The code is as follows:

```
1 #include<iostream>
2 using namespace std;
3 int main() {
4     int Number, reminder;
5     cout << "Enter number : ";
6     cin >> Number;
7
8     int octal = 0;
9     int value = 1;
10
11     do
12     {
13         reminder = Number % 8;
14         octal = octal + (reminder * value);
15         Number = Number / 8;
16         value = value * 10;
17
18     } while (Number > 0);
19
20     cout << "Your decimal Number to octal is : " << octal;
21
22     cout << endl;
23
24     return 0;
}
```

The Microsoft Visual Studio Debug Console on the right shows the program's execution. It prompts the user to enter a number, and the user has entered 170. The program then outputs the octal equivalent, 252. The console text is as follows:

```
Enter number : 170
Your decimal Number to octal is : 252

C:\Users\Admin\source\repos\Project2\x64\Debug\Pr
To automatically close the console when debugging
ly close the console when debugging stops.
Press any key to close this window . . .
```

Problem 4:

Four-Function Calculator.

Code:

```
#include<iostream>
using namespace std;
int main()
{
    float number1, number2;
    char Choice = 'y' / 'n';
    char Operation;

    do
    {

        cout << "Enter first number, operator, and second number : ";
        cin >> number1 >> Operation >> number2;
        switch (Operation)
        {

            case('+'):
                cout << "Answer = " << number1 + number2 << endl;
                break;

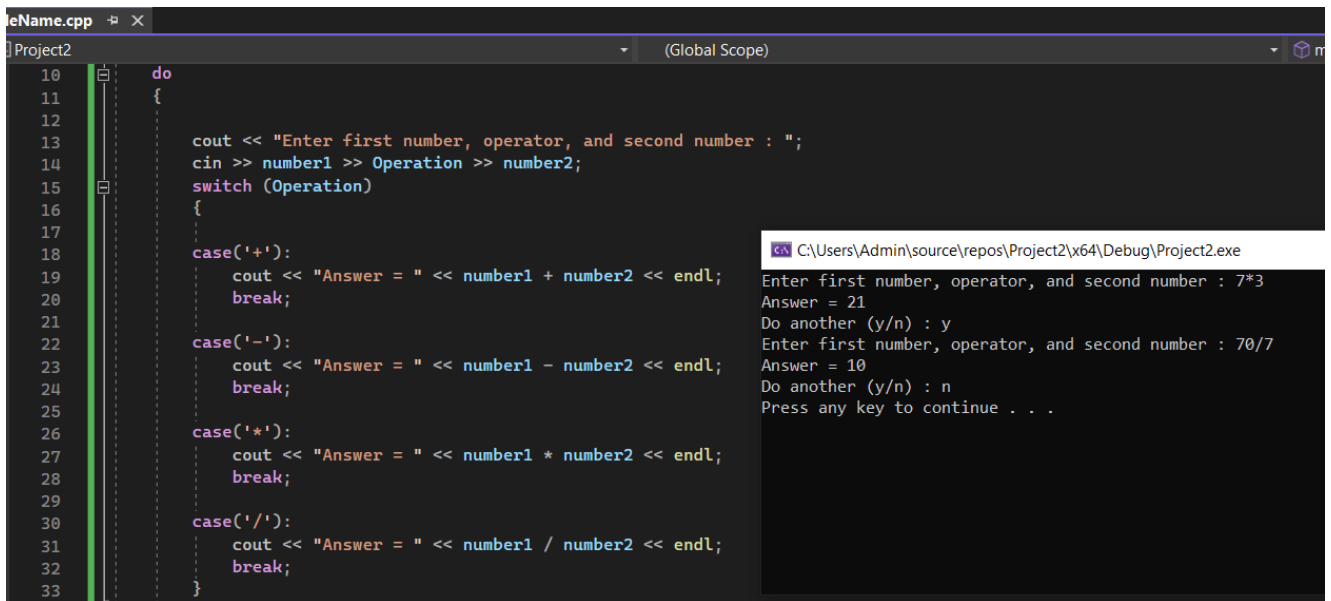
            case('-'):
                cout << "Answer = " << number1 - number2 << endl;
                break;

            case('*'):
                cout << "Answer = " << number1 * number2 << endl;
                break;

            case('/'):
                cout << "Answer = " << number1 / number2 << endl;
                break;
        }
        cout << "Do another (y/n) : ";
        cin >> Choice;

    } while (Choice == 'y');
    system("pause");
    return 0;
}
```

Screenshot:



The screenshot shows a C++ IDE with a file named 'Project2'. The code is a calculator program that takes two numbers and an operator as input and performs the corresponding arithmetic operation. The code is as follows:

```
10 do
11 {
12
13     cout << "Enter first number, operator, and second number : ";
14     cin >> number1 >> Operation >> number2;
15     switch (Operation)
16     {
17
18         case('+'):
19             cout << "Answer = " << number1 + number2 << endl;
20             break;
21
22         case('-'):
23             cout << "Answer = " << number1 - number2 << endl;
24             break;
25
26         case('*'):
27             cout << "Answer = " << number1 * number2 << endl;
28             break;
29
30         case('/'):
31             cout << "Answer = " << number1 / number2 << endl;
32             break;
33     }
```

The output window shows the following execution:

```
C:\Users\Admin\source\repos\Project2\x64\Debug\Project2.exe
Enter first number, operator, and second number : 7*3
Answer = 21
Do another (y/n) : y
Enter first number, operator, and second number : 70/7
Answer = 10
Do another (y/n) : n
Press any key to continue . . .
```

EXTRA TASK

Task No 1:

It is necessary for the program to display the following sequence of numbers:

7 14 21 28 35 42 49 56 63 70 77 84 91 98.

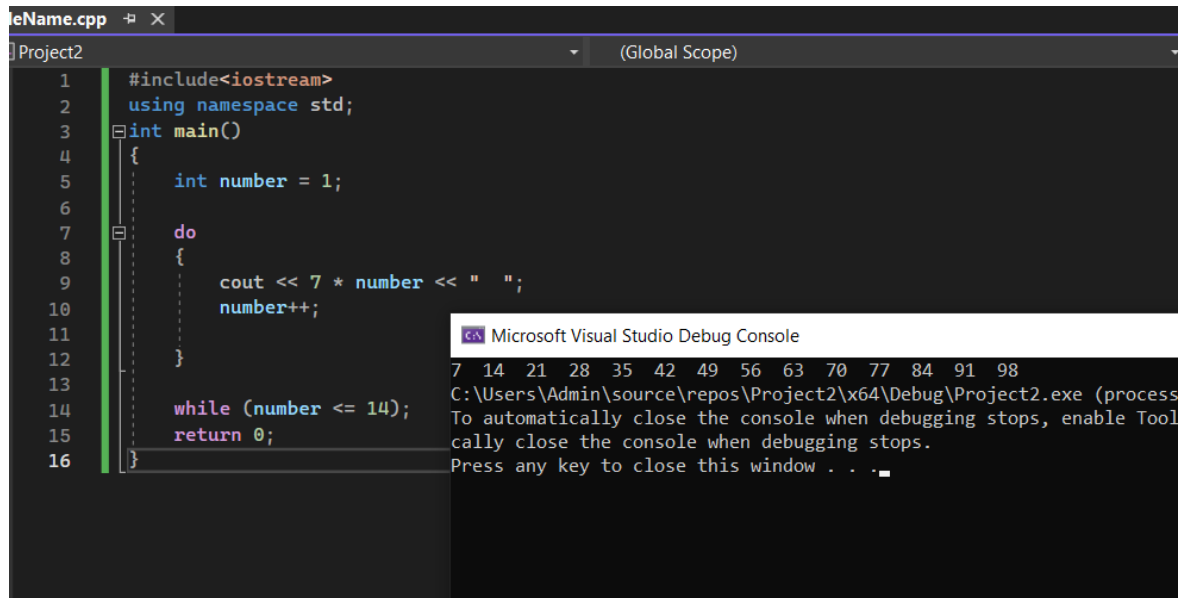
Code:

```
#include<iostream>
using namespace std;
int main()
{
    int number = 1;

    do
    {
        cout << 7 * number << " ";
        number++;
    }

    while (number < 15);
    return 0;
}
```

Screenshot:



The screenshot shows a Visual Studio IDE with a C++ file named 'eName.cpp'. The code is as follows:

```
1 #include<iostream>
2 using namespace std;
3 int main()
4 {
5     int number = 1;
6
7     do
8     {
9         cout << 7 * number << " ";
10        number++;
11    }
12
13    while (number <= 14);
14    return 0;
15 }
```

The debug console shows the output of the program:

```
Microsoft Visual Studio Debug Console
7 14 21 28 35 42 49 56 63 70 77 84 91 98
C:\Users\Admin\source\repos\Project2\x64\Debug\Project2.exe (process
To automatically close the console when debugging stops, enable Tool
cally close the console when debugging stops.
Press any key to close this window . . .
```

Task No 2:

It is necessary to display the following sequence of numbers

1 2 4 8 16 32 64 128 256 512.

Code:

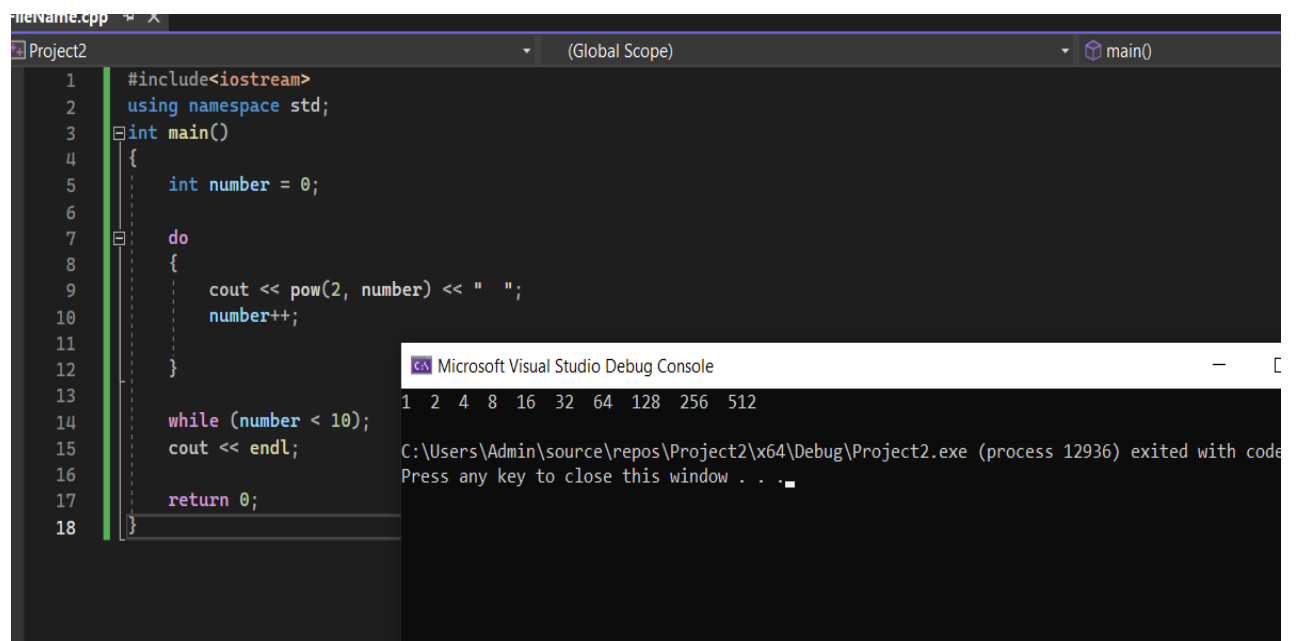
```
#include<iostream>
using namespace std;
int main()
{
    int number = 0;

    do
    {
        cout << pow(2, number) << " ";
        number++;
    }

    while (number < 10);
    cout << endl;

    return 0;
}
```

Screenshot:



The screenshot shows a C++ program in Visual Studio. The code is as follows:

```
1 #include<iostream>
2 using namespace std;
3 int main()
4 {
5     int number = 0;
6
7     do
8     {
9         cout << pow(2, number) << " ";
10        number++;
11    }
12
13    while (number < 10);
14    cout << endl;
15
16    return 0;
17 }
```

The Microsoft Visual Studio Debug Console shows the output of the program:

```
1 2 4 8 16 32 64 128 256 512
C:\Users\Admin\source\repos\Project2\x64\Debug\Project2.exe (process 12936) exited with code
Press any key to close this window . . .
```

Conclusion:

Understanding the basic concept of do-while loop. In do-while loop they will executed once if condition will false if condition is true they will run according to condition. The body of the loop is executed and then the condition is tested.