CS-114 - Fundamental of Programing

Lab Manual # 08

Course Instructor: Dr. Muhammad Usman Akram

Lab Instructor: Engr. Ayesha Batool

Student Name: _____

Degree/ Syndicate: _____

DATE:

_____

Lab Manual # 08
Repetition structures (II)

Objective:

To understand repetition structure and the types of repetition structure.

Description:

While Loop

A loop is part of a program that repeats. The while loop has two important parts

1. An expression that is tested for a true or false value.
2. A statement or block that is repeated as long as the expression is true.

```
while (expression)
{
    statement;
    statement;
    // Place as many statements here
    // as necessary.
}
```

Do While Loop

The while Loop Is a Pre-test Loop ,which means it tests its expression before each iteration whereas the do-while loop is a post-test loop, which means its expression is tested after each iteration.

```
do
{
  statement;
  statement;
  // Place as many statements here
  // as necessary.
} while (expression);
```

Infinite Loops:

If a loop does not have a way of stopping, it is called an infinite loop. An infinite loop continues to repeat until the program is interrupted. Here is an example of an infinite loop:

```
int number = 0;
while (number < 5)
{
    cout << "Hello\n";
}
```

We can make this loop finite by adding a line as shown below

```
while (number < 5)
{
    cout << "Hello\n";
    number++;
}
```

Examples:

The following example averages a series of three test scores for a student. After the average is displayed, it asks the user if he or she wants to average another set of test scores. The program repeats as long as the user enters Y for yes.

Example	OUTPUT

```
#include <iostream>
using namespace std;

int main()
{
    int score1, score2, score3; // Three scores
    double average;             // Average score
    char again;                 // To hold Y or N input

    do
    {
        // Get three scores.
        cout << "Enter 3 scores and I will average them: ";
        cin >> score1 >> score2 >> score3;

        // Calculate and display the average.
        average = (score1 + score2 + score3) / 3.0;
        cout << "The average is " << average << ".\n";

        // Does the user want to average another set?
        cout << "Do you want to average another set? (Y/N) ";
        cin >> again;
    } while (again == 'Y' || again == 'y');
    return 0;
}
```

```
Enter 3 scores and I will average them: 80 90 70 [Enter]
The average is 80.
Do you want to average another set? (Y/N) y [Enter]
Enter 3 scores and I will average them: 60 75 88 [Enter]
The average is 74.3333.
Do you want to average another set? (Y/N) n [Enter]
```

While	*Do-while*
1. Condition is at top.	1. Condition is at the bottom.
2. No necessity of bracket if there is single statement in body.	2. Brackets are compulsory even if there is a single statement.
3. There is no semicolon at the end of while.	3. The semicolon is compulsory at the end do-while.
4. Computer executes the body if and only if condition is true.	4. Computer executes the body at least once even if condition is false.
5. This should be used when condition is more important.	5. This should be used when the process is important.
6. This loop is also refered as entry controlled loop.	6. This loop is also refered as exit controlled loop.

7.While(n<10)
{
printf("%d\n",n);
}
	7.Do
	{
	Printf("%d\n",n);
	}while(n<=100);

Lab Task:

1. Convert the following while loop to a do-while loop:

```
int x = 1;
 while (x > 0)
 {
cout << "enter a number: ";
 cin >> x;
 }
```

2. Write a do-while loop that asks the user to enter two numbers and operator +, -, / ,*. After performing operation, and the resultant answer should be displayed, and the user should be asked if he or she wishes to perform the operation again. If so, the loop should repeat; otherwise it should terminate.
3. Write a C++ program to Print Table of any Number using while loop.

Output should look like:

$2 \times 1 = 2$

$2 \times 2 = 4$

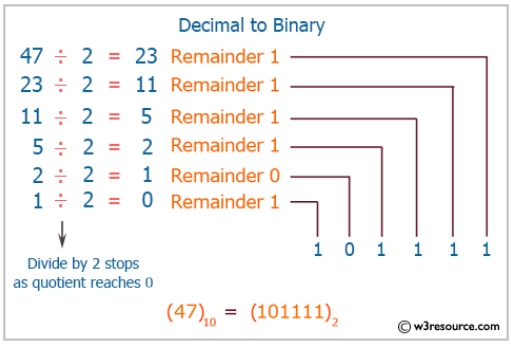
4. Write a C++ program, take two integer i (starting point) and n (ending point) from user and print all even numbers between i to n using while loop.
   For example:
   Let say i=5; and n =20; Output should be (6 8 10 12 14 16 18 20).
5. Write a C++ program to print natural numbers in reverse from n to 1 using both do while loop and while loop.

Home Task:

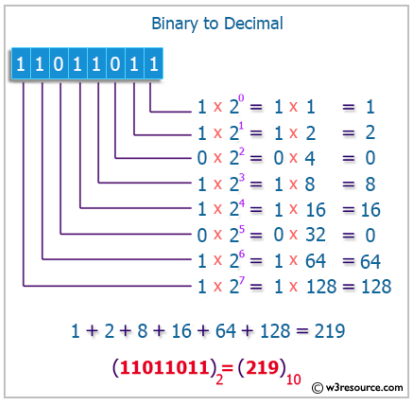
1. Write a program in C++ to find LCM of any two numbers using HCF.
2. Write a program in C++ to find out the sum of an Arithmetic progression series.
3. Write a program in C++ to create a diamond

```
    *
   ***
  *****
 *******
*********
 *******
  *****
   ***
    *
```

4. Write a program in C++ to convert a decimal number to binary number.



5. Write a program in C++ to convert a binary number to decimal number.



Useful links for practice

https://www.w3schools.com/

https://www.codecademy.com/learn/learn-c-plus-plus