Lab 07
Repetition Control Structures

(While and Do while Loop)

While

The While loop repeatedly executes a piece of code until a condition becomes false. It is also called condition controlled loop.

Syntax:

```
while (condition){

// statement(s)

}
```

Example:

```
int i=0;

while(i<5){

    cout << i<< endl;

    i++;


}
```

Do while

Unlike for and while loops, which test the loop condition at the top of the loop, the do...while loop checks its condition at the bottom of the loop. A do...while loop is similar to a while loop, except that a do...while loop is guaranteed to execute at least one time.

Syntax

```
do {
    statement(s);
}
while( condition );
```

Example

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

int main () {
   // Local variable declaration:
   int a = 10;

   // do loop execution
   do {
      cout << "value of a: " << a << endl;
      a = a + 1;
   } while( a < 20 );

   return 0;
}
```

Lab Tasks

Problem 01

Write a C++ program to take integers as input from user until user enters zero. Take product of all those numbers and calculate average. **Note:** Since we don't know how many different numbers a user would want to input, so we will use while loop to take product until the user enters digit zero. Zero will not be added to the calculation.

Problem 02

In mathematics, the factorial of a positive integer n, denoted by n!, is the product of all positive integers less than or equal to n. For Example 5! = 5 * 4 * 3 * 2 *1. Write a program to compute factorial to an integer entered by user. If the number is less than zero display "invalid Input". **Note : 0! = 1** .

Problem 03

Write a program to take integers as input from user until the sum of all those numbers becomes equal to or greater than 100. When the sum becomes equal to or greater than 100, display how many numbers user entered to reach or cross 100. **Note: Use do while loop**

Example

Enter an integer: 10

Enter an integer: 20

Problem 04
Write an infinite loop.
An infinite loop never ends. Condition is always true.

Problem 05
A child visits nearby tuck shop with 100 coins in pocket. The shop has eatables with different prices. Child randomly chooses an eatable and the sale price for that is added to the total sales. Consequently upon each random choice the child's coins get consumed (decreased). The child can only choose from eatables having price below or equal to the coins left in his pocket. The child can buy eatables until the total sales price equals to 100. The child should get a receipt for how many total products did he buy and what was the price for each.

Write a program to simulate the above mentioned scenario using do while loop. Your program should randomly choose a number between 1 and the total number of coins left. The process must end when all 100 coins get consumed. For random number generation find below sample code:

```
#include<cstdlib>

#include<ctime>

srand(time(0));

int n=(rand()%100)+1
```