Faculty of Information Technology, University of Moratuwa
BSc. (Hons) in Information Technology
BSc. (Hons) in Artificial Intelligence
IN1101 - Programming Fundamentals

Level 1 – Semester 1

Introduction

In this lab session, you'll learn about repetition which is the flow control part of ***while*** and ***do while*** loops properly.

While loop

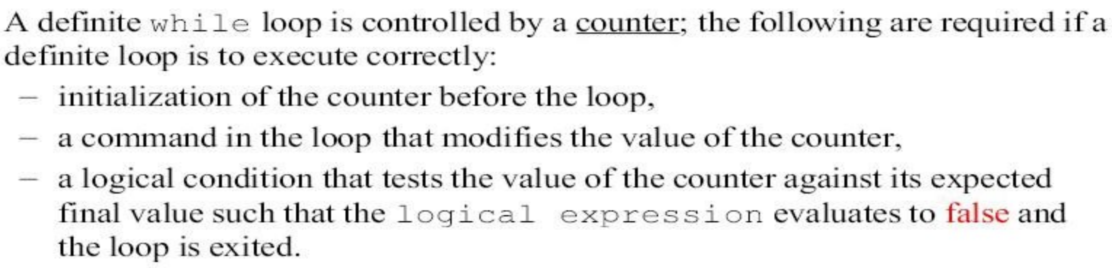
- ✓ While loop will run until the condition becomes false.
- ✓ If the condition becomes false, it will come out of the loop and go to the next line.
- ✓ A while syntax looks like this.

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

Components of a *while* Loop

•Initial value of counter
•Condition (defines the final/end value)
•Body of statements
•Increment/decrement counter

While Repetition Structure



A definite `while` loop is controlled by a counter; the following are required if a definite loop is to execute correctly:

- – initialization of the counter before the loop,
- – a command in the loop that modifies the value of the counter,
- – a logical condition that tests the value of the counter against its expected final value such that the `logical expression` evaluates to false and the loop is exited.

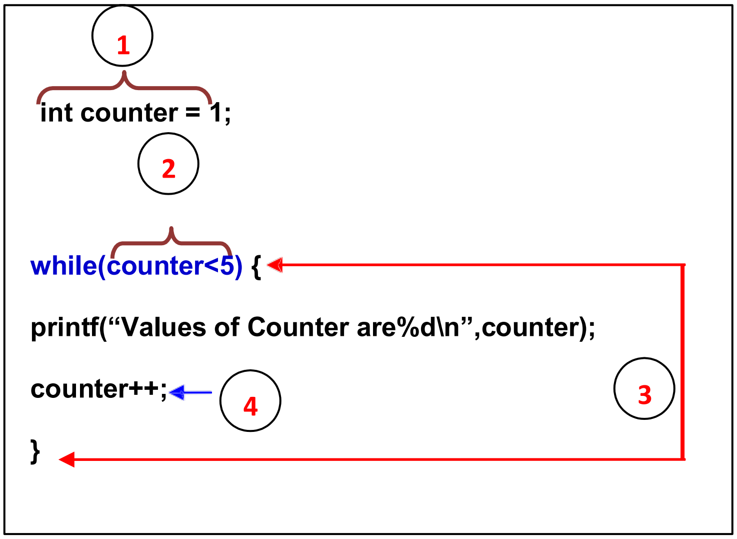
Operation of a ***while*** structure in a program:

```
 int counter = 1;
while(counter < 5){
        printf("%d\n", counter);
        counter++;
}
```

Example of a repetition loop using a ***while*** structure:

```
int i = 1;
while (i < = 10) {
        printf("Iteration number: %d\n", i);
        i++;
}
```

Form of execution of *while* loop



```
int counter = 1;

while(counter<5) {
printf("Values of Counter are%d\n",counter);
counter++;
}
```

Components of a while Loop

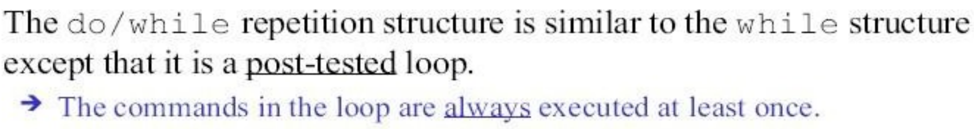
1 Initial value of counter

2 Condition

3 Body of statements

4 Increment/decrement counter

do/while Repetition Structure



The do/while repetition structure is similar to the while structure except that it is a post-tested loop.

➔ The commands in the loop are always executed at least once.

do while syntax

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

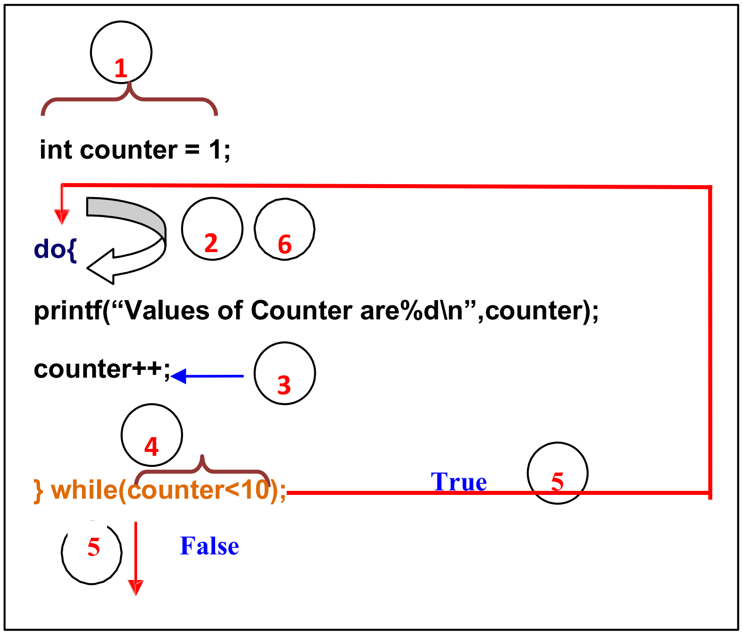
Operation of a ***dowhile*** structure in a program:

```
int counter = 1;
do{
printf("%d\n", counter);
counter++;
}while(counter<5);
```

Example of a definite repetition loop using the do/while structure:

```
int i= 1;
do {
printf("Iteration %d\n", i);
i++;
 } while (i<= 10);
```

Form of execution of *dowhile* loop



```
int counter = 1;
do{
printf("Values of Counter are%d\n",counter);
counter++;
} while(counter<10);
```

Components of a do while Loop

1. Initial value of counter
2. Print the statement
3. Increment/decrement
4. Condition
5. If true execute the do body
If false exit from the do body
6. Similar as step 2

Exercises

1. Using a proper loop statement (while) Try to get the following outputs using C.
   - a. Display the numbers from 1 to 10 line by line
   - b. Display the numbers from 10 to 1 line by line
   - c. Display the sequence 1,3,5,7,9 line by line
   - d. Display the sequence 1,4,7,10,13,16… up to 52 line by line
   - e. Display the sequence 2,3,5,9,17,…up to 33 line by line
   - f. Display all the odd numbers between 1 to 100.
   - g. Display all the even numbers between 1 to 100.
   - h. Display all the numbers that are multiple of 3 between 1 to 100.

2. Print the integers from 1 to 20 using a while loop and the counter variable x. Print only five integers per line.
3. Write a program that calculates and prints the sum of the even integers from 2 to 50.
4. Write a program that calculates and prints the product of the odd integers from 1 to 30.
5. Write a C program to get the sum of odd integers between 1 and 99 using a while loop.
6. Using a proper loop (while) statement. Try to get the following outputs using C.
   - a. Summation of all the numbers from 1 to 10
   - b. Multiplication of all the numbers from 1 to 10
   - c. Number of even numbers from 1 to 100
   - d. Number of odd numbers from 1 to 100

7. Write a C program using a while loop to compute total course fees for students.
   1. Prompt the user to enter their name (string), id no (string) and duration of the study (integer).
   2. Suppose that the course fee is starting from $200 in the first year and increases 5% in the following years.
   3. Calculate the annual fee and total course fees for the whole duration of the study.
   4. Display all information

8. Write a C program for the following output:-

```
Enter First Number, operator, second number: 10/3

Answer = 3.3333333

Do You Want to Try It Again(y/n)? y

Enter First Number, operator, second number: 12/3
Answer = 4
Do You Want to Try It Again(y/n)? n
Goodbye
```

9. One large chemical company pays its salesperson on a commission basis. The salespeople each receive $200 per week plus 9% of their gross sales for that week. For example, a salesperson who sells $5000 worth of chemicals in a week receives $200 plus 9% of his salary as a total of $650.

   Write a C program that uses a while statement to input each salesperson's gross sales for the last week and calculates and displays the salesperson's earnings.
   Note: - Program will quit when the user enters -1 as input.

10. Write a C program that takes an integer from the user and finds its reverse.

11. Write a C program to Check an Armstrong Number.
    A number is said to be Armstrong if it is equivalent to the sum of its own digits when those digits are each multiplied by the power of the number of digits.
    For example:
    153 = 1*1*1 + 5*5*5 + 3*3*3 // 153 is an Armstrong number.

12. Write a C Program to find the Highest Common Factor (H.C.F.) or Greatest Common Divisor (G.C.D.) for given two numbers.
    Hint: The largest integer which is perfectly divisible by two numbers is known as H.C.F or G.C.D of those two numbers.

13. Write a Program to find Lowest Common Multiple (L.C.M.)
    Hint: L.C.M of two integers a and b is the lowest positive integer that is divisible by both a and b

14. Write a C Program to Find the Number of Digits in a Number.

15. Write a C program to calculate the power of a number without using math.h file.

16. Write a C program to check whether a number is a palindrome or not.
If a number that remains the same when it is written backwards is known as a palindrome.
For example, 121, 34543, 343, 131, and 48984 are the palindrome numbers.

17. Write a Program to read a sequence of integer numbers from the keyboard and evaluate the following for all integers that were entered prior to the zero entry. (Use Do-While)

    a. Number of positive integers
    b. Number of negative integers.
    c. Number of integers that are multiples of 3
    d. Summation of all positive and negative integers
    e. Maximum and minimum values

    Your output should be like this:

```
Enter the numbers:
10
89
-8
0

Number of positive integers: =2
Number of negative integers: =1
Sum of positive integers: =99
Sum of negative integers: =-8
Number of multipliers of 3 :=0
Maximum value: =89
Minimum value: =-8
```