**LAB # 7**

**OBJECT**

***STUDY OF FOR AND NESTED FOR LOOPS***

**THEORY**

## Loop

There are many situations where a certain operation is repeated number of time, this repetition is termed “loop” in programming languages.

There are three types of Loops:

* for Loop
* while Loop
* do - while Loop

Nesting may extend these loops.

**For Loop**

There are different loop statements in C language and “For” is one of them. These statements also allow one or more statements to be repeated. The “For” loop is considered most flexible loop because it allows number of variations. In its most common form, the “For” loop is used to repeat a statement or a block of statements a specified number of items.

General syntax:

**for (initialize ;condition ;increment)**

**{**

**Do this;**

**}**

### Example- 1

#include <stdio.h>

void main (void)

{

int count;

for ( count = 0 ; count<10 ; count++ )

printf( “count = %d\n”, count);

}

### Output:-

The result of the program will be as:

Count = 0

Count = 1

Count = 2

Count = 3

Count = 4

Count = 5

Count = 6

Count = 7

Count = 8

Count = 9

**Nested loops**

When the body of one loop contains another, the second loop is said to be the nested loop inside the first. Any of the c language’s loops may be nested with in any other loop.

**Looping and Indentation**

When you are using loops in a program then you have to take special care for indentation and formatting of the program. This will help you debugging the program during any modification and error checking. It will be easy to other programmer as well to understand the code of the program easily and quickly. One should indent the coding of the program in the following way:

void main (void)

{

initialization;

statements;

.

.

for (initialization; conditional test ; increment)

{

for (initialization; conditional test ; increment)

statements

}

}

The body of the outer ‘for’ loop is indented and the body of the inner loop is further indented. If more nested loops are present more indentation are required to make a program simpler and easier to understand.

**TASKS TO BE PERFORMED**

1. Write a program to generate a series of first 50 even numbers.

# PROGRAM

#include <stdio.h>

int main(){

int a;

for(a=0;a<50;a+=2){

printf("\n%d",a);

}

getch();

}

1. Write a program that generates a table of any number.

# PROGRAM

#include <stdio.h>

int main(){

int table, ans, x;

printf("Enter any number ");

scanf("%d",&table);

for(x=1;x<=10;x++){

ans=table\*x;

printf("\n\t%d x %d = %d",table,x,ans);

}

getch();

}

1. Write a program to calculate the factorial of any number.

# PROGRAM

#include <stdio.h>

int main(){

int c, n, fact=1;

printf("Enter any number to find factorial ");

scanf("%d",&n);

for(c=1;c<=n;c++){

fact=fact\*c;

printf("\nFactorial of number %d = %d",n,fact);

}

getch();

}