# CHAPTER 5 LOOPING

- We use loop when you want to execute statement several times until a condition is reached.
- Generally, loops consist of two parts:
- one or more control expressions which control the execution of the loop.
- body , which is the statement or a set of statement which is executed over and over

**In C programming, we use 4 types of looping central structures:**

- for loop
- while loop
- do while loop
- Nested loop

UNIVERSITY OF KIGALI

# 5.1. FOR LOOP

The basic format of the for statement/syntax is:

for( start_condition; continue_ condition; re-evaluation )

{

Set of program statement;

}

# 5.1.1 Sample program using a for loop

```
#include <stdio.h>
 main()
 {
int  count;
for(count= 1;count<= 10;count=count+1)
printf("%d\t\t",count);
 }
```

**Sample Program Output** 1 2 3 4 5 6 7 8 9 10

UNIVERSITY Of KIGALI

# 5.1.2 An example of using a for loop to print out characters

```
#include<stdio.h>
main()
{
char letter;
for( letter = 'A'; letter <= 'E'; letter = letter + 1 )
{
printf("%c ", letter);
}
getchar();
}
```

**Sample Program Output**A B C D E

UNIVERSITY Of KIGALI

# 5.1.3 AN EXAMPLE OF USING A FOR LOOP TO DISPLAY ODD NUMBER BETWEEN 1 AND 50

```
 #include<stdio.h>
main()
{
Int i;
for(i=1; i <= 50; i++ )
{
If(i%2!=0)
printf("%d\t", i);
}
getchar();
}
```

# 5.1.4 SUM OF THE FIRST N NUMBERS

```
#include<stdio.h>
main()
{
Int   i,n,sum=0;
printf("ENTER THE LIMIT\n");
 scanf("%d",&n);
 for(i=1;i<=n;i++)
{
 sum=sum+i;
 }
printf("\n THE SUM OF FIRST %d NUMBERS = %d", n,
   sum);
 getchar();
}
```

UNIVERSITY Of KIGALI

# 5.1.5 MULTIPLICATION TABLE

```
#include<stdio.h>
main()
{
 int i,table,counter=1;

 printf("ENTER THE MULTIPLICATION TABLE \n");
 scanf("%d",&table);
 printf("ENTER THE COUNTER VALUE\n");
 scanf("%d",&counter);
 printf("THE REQUIRED TABLE IS SHOWN BELOW:\n");
 for(i=1;i<=counter;i++)
 {
  printf("\n%d * %d = %d",table,i,i*table);
 }
 getchar();
}
```

UNIVERSITY Of KIGALI

# 5.1.6 PROGRAM THAT DISPLAY MULTIPLICATION TABLE FROM 1 TO 10

```
#include<stdio.h>
main()
{
Int  i,j;
for(i=1;i<=10;i++)
{
for(j=1;j<=10;j++)
{
 printf("%d * %d = %d\n",i,j,i*j);
}
printf("\n\n");
printf("multiplication table of %d\n",i);
printf("\n\n");
}
getchar();
}
```

UNIVERSITY Of KIGALI

# 5.1.7 SOMETIME ANY LOOP COULD BE PLACED INSIDE THE OTHER LOOP

```
#include<stdio.h>
main()
{
int i,j;
for(i=1;i<=5;i++)
{
for(j=1;j<=i;j++)
printf("%d\t",j);
printf("\n");
} }
```

**The output is:**

```
1
1   2
1    2   3
1    2   3   4
1    2   3   4   5
```

UNIVERSITY Of KIGALI

```
#include<stdio.h>
main()
{
int i,j;
for(i=1;i<=5;i++)
{
for(j=5;j>=i; j--)
 printf("%d\t", j);
printf("\n");
}
}
```

**The output is:**

- 5 4 3 2 1
- 5 4 3 2
- 5 4 3
- 5 4
- 5

# PRACTICAL ASSIGNMENT I

Make a program which gives the below out put

```
5
4  4
3   3   3
2   2   2   2
1   1   1   1   1
```

UNIVERSITY Of KIGALI

# solution

```
#include<stdio.h>
main()
{
   int i,j;
   for(i=5;i>=1;i--)
{
   for(j=5;j>=i;j--)
   printf("%d",i);
}}
```

# 5.2 WHILE LOOP

- The while provides a mechanism for repeating C statements whilst a condition is true. While loop has one control expression and executes as long as that expression is true.

**Syntax:**

```
while(expression)
{
statements;
}
```

# /* Sample program including while */

```
#include<stdio.h>
main()
{
int  loop=0;
while(loop<=10)
{
Printf ("%d\n",loop);
++loop;
}
}
```

```
#include<stdio.h>
main()
{
int  i=1;
while(i<=10)
{
Printf ("%d\n",i);
++i;
}
}
```

# 5.2.1 Using while for multiplication table from 1 to 10

```
#include<stdio.h>
main()
{
int i,n=5;
i=1;
while(i<=10)
{
printf("%d x %d = %d\n",i,n,i*n);
printf("\n\n");i++;
}
}
```

**NB:**
We can also provide Printf() and scanf () statements here in order to validate any number needed, by replacing n=5.

# 5.2.2 USING WHILE FOR A PROGRAM TO DISPLAY EVEN NUMBER BETWEEN 1 AND 20

```
#include<stdio.h>
main()
{
int i;
 i=1;
while(i<=20)
{
if(i%2==0)
printf("%d\t",i);
i++;
}
}
```

UNIVERSITY Of KIGALI

# 5.2.3 USING WHILE FOR A PROGRAM TO DISPLAY THE FIRST INTEGER NUMBERS

```
#include<stdio.h>
main()
{
int i;
i=1;
while(i<=10)
{
printf("%d\n",i);
i++;
}
}
```

# 5.3 DO WHILE

**When developing programs, it sometimes become desirable to have the test made at the end of the loop rather than at the beginning.**

- Naturally, C language provides a special language construct to handle such a situation.
- This looping statement is known as the do statement.

The syntax is:

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

UNIVERSITY Of KIGALI

# Execution of the do statement proceeds as follows:

- Program statement is executed first.
- Next, the expression inside the parenthesis is evaluated, if the result of evaluating is TRUE, the loop continues and the program statement is once again executed.
- As long as evaluation of expression continues to be true, program statement is repeatedly executed. When evaluation of the expression is FALSE, the loop is terminated and the next statement in the program is executed in the normal sequential manner.
- **The do statement is simply a transposition of the while loop statement with the looping conditions placed at the end of the loop rather than at the beginning.**
- **N.B:** Remember that, unlike the for and while loops, the do statement guarantees that the body of the loop will be **executed at least once**.

# 5.3.1 PROGRAM THAT DISPLAYS ANY SERIES OF NUMBERS USING DO WHILE

```
#include<stdio.h>
main()
{
int num,i=1;
printf("Enter any number you want\n");
scanf("%d",&num);
do
{
printf("%d\n",i);
i++;
}
while(i<=num);
}
```

UNIVERSITY Of KIGALI

# MULTIPLICATION TABLE USING DO_WHILE_LOOP

```
#include<stdio.h>
main()
{
int i=1,table,counter=1;
 printf("ENTER THE MULTIPLICATION TABLE \n");
 scanf("%d",&table);
 printf("ENTER THE COUNTER VALUE\n");
 scanf("%d",&counter);
 printf("THE REQUIRED TABLE IS SHOWN BELOW:\n");
do
{
printf("%d*%d=%d\n",i,table,i*table);i++;
}
while(i<=counter);
}
```

UNIVERSITY Of KIGALI

# 5.4 NESTED LOOP

This nested loop uses inner loop and outer loop.

Syntax:

```
for( initialisation, condition, increment/decrement)
{
for (condition)
{
statement
}
}
```

## 3.4.1 PROGRAM TO CHECK WHETER THE GIVEN NUMBER IS PRIME OR NOT

```
#include<stdio.h>
main()
{
int n,j,ck=0;
printif("enter a number");
scanf("%d",&n);
for(j=2;j<n;j++)
{
If(n%j==0)
{
ck=1;
}
}
If(ck==0)
{
printf("The number is prime\n");
}
 else
 {
printf("The number is not prime\n");
}
```

# 5.4.2 PROGRAM TO CHECK THE PRIME NUMBER BETWEEN 1 & N

```
#include<stdio.h>
main()
{
int n,i,j,ck;
printf("enter a limit number\n");
scanf("%d",&n);
for(i=1;i<=n;i++)
{
ck=0;
for(j=2;j<i;j++)
{
if(i%j==0)
ck=1;
}
if(ck==0)
printf("%d\t",i);
}
}
```

UNIVERSITY Of KIGALI

# Exercise

- Make a program to print out multiplication table of 7 using do while.