EXPERIMENT 7

NAME: Rahil Sharma
PRN: 18070123062
BATCH: EA-3
SUBJECT: ESRTOS

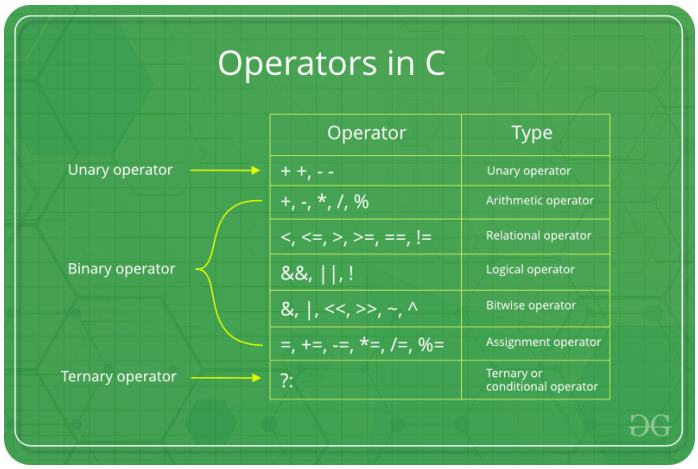
AIM: To use Loops and Conditional Statements in Linux

THEORY:

- Operators in C:

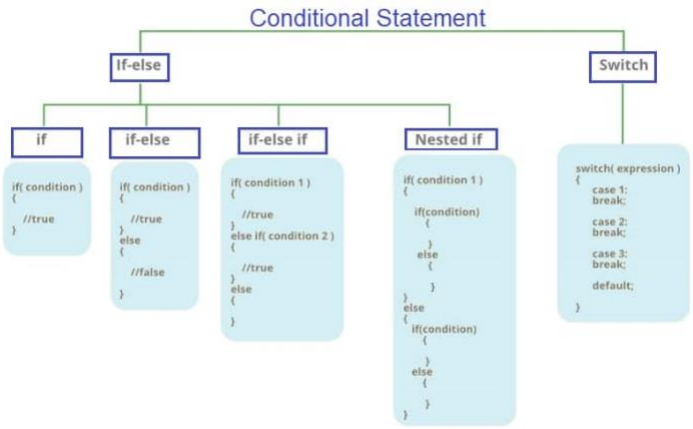
  An operator is a symbol that operates on a value or a variable. For example: + is an operator to perform addition.

  C has a wide range of operators to perform various operations.



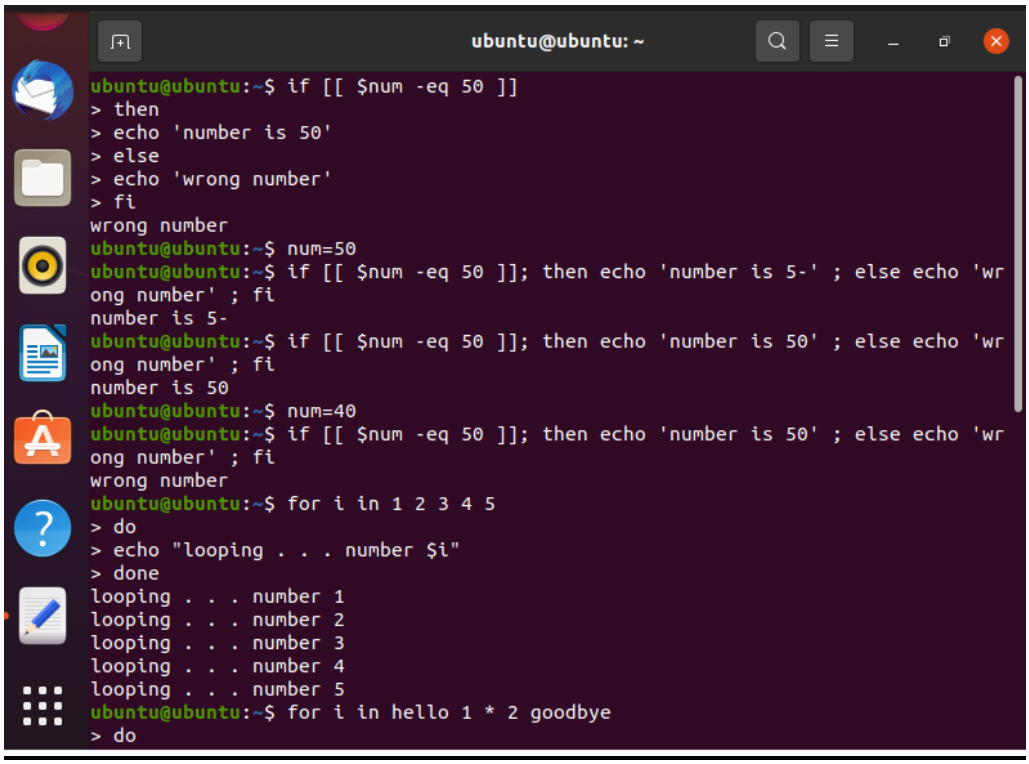
- <u>Conditional Statements in C:</u>

Conditional Statements in C programming are used to make decisions based on the conditions. Conditional statements execute sequentially when there is no condition around the statements. It is also called as branching as a program decides which statement to execute based on the result of the evaluated condition.



<u>CODES & OUPUTS OF THE PROGRAM:</u>

1. Task 1: Using if else loop on the Linux Terminal

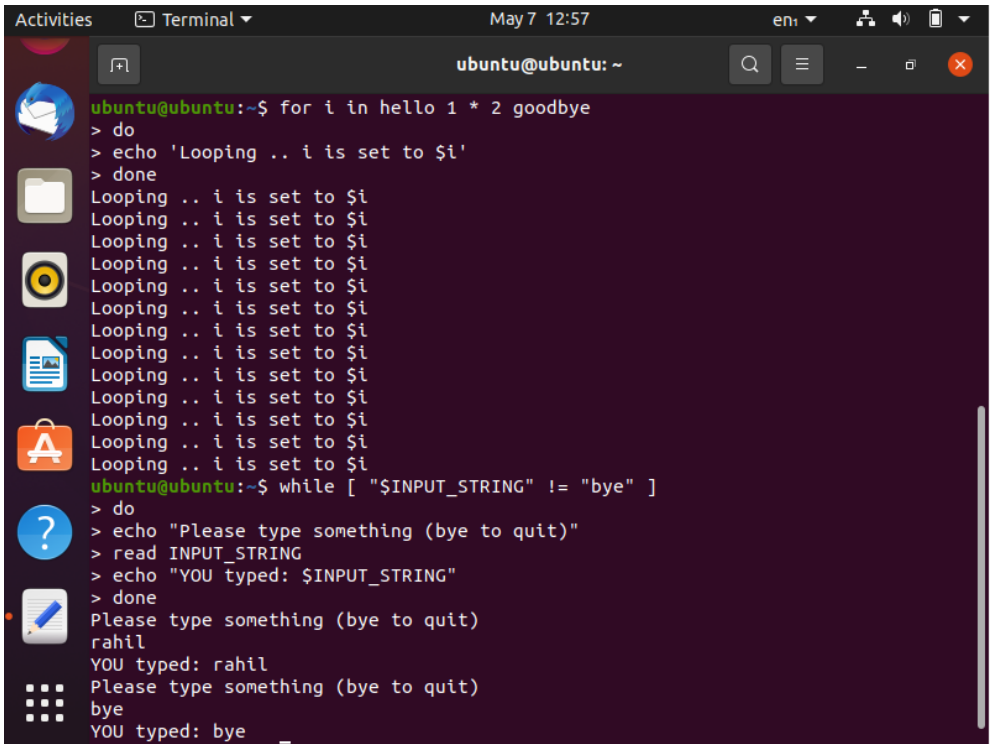


2. For Loop in Linux Terminal

```
ubuntu@ubuntu: ~
ubuntu@ubuntu:~$ if [[ $num -eq 50 ]]
> then
> echo 'number is 50'
> else
> echo 'wrong number'
> fi
wrong number
ubuntu@ubuntu:~$ num=50
ubuntu@ubuntu:~$ if [[ $num -eq 50 ]]; then echo 'number is 5-' ; else echo 'wr
ong number' ; fi
number is 5-
ubuntu@ubuntu:~$ if [[ $num -eq 50 ]]; then echo 'number is 50' ; else echo 'wr
ong number' ; fi
number is 50
ubuntu@ubuntu:~$ num=40
ubuntu@ubuntu:~$ if [[ $num -eq 50 ]]; then echo 'number is 50' ; else echo 'wr
ong number' ; fi
wrong number
ubuntu@ubuntu:~$ for i in 1 2 3 4 5
> do
> echo "looping . . . number $i"
> done
looping . . . number 1
looping . . . number 2
looping . . . number 3
looping . . . number 4
looping . . . number 5
ubuntu@ubuntu:~$ for i in hello 1 * 2 goodbye
> do
```

```
Activities  Terminal  May 7 12:57  en
ubuntu@ubuntu: ~
ubuntu@ubuntu:~$ for i in hello 1 * 2 goodbye
> do
> echo 'Looping .. i is set to $i'
> done
Looping .. i is set to $i
Looping .. i is set to $i
Looping .. i is set to $i
Looping .. i is set to $i
Looping .. i is set to $i
Looping .. i is set to $i
Looping .. i is set to $i
Looping .. i is set to $i
Looping .. i is set to $i
Looping .. i is set to $i
Looping .. i is set to $i
Looping .. i is set to $i
Looping .. i is set to $i
ubuntu@ubuntu:~$ while [ "$INPUT_STRING" != "bye" ]
> do
> echo "Please type something (bye to quit)"
> read INPUT_STRING
> echo "YOU typed: $INPUT_STRING"
> done
Please type something (bye to quit)
rahil
YOU typed: rahil
Please type something (bye to quit)
bye
YOU typed: bye
```

3. While Loop in Linux Terminal

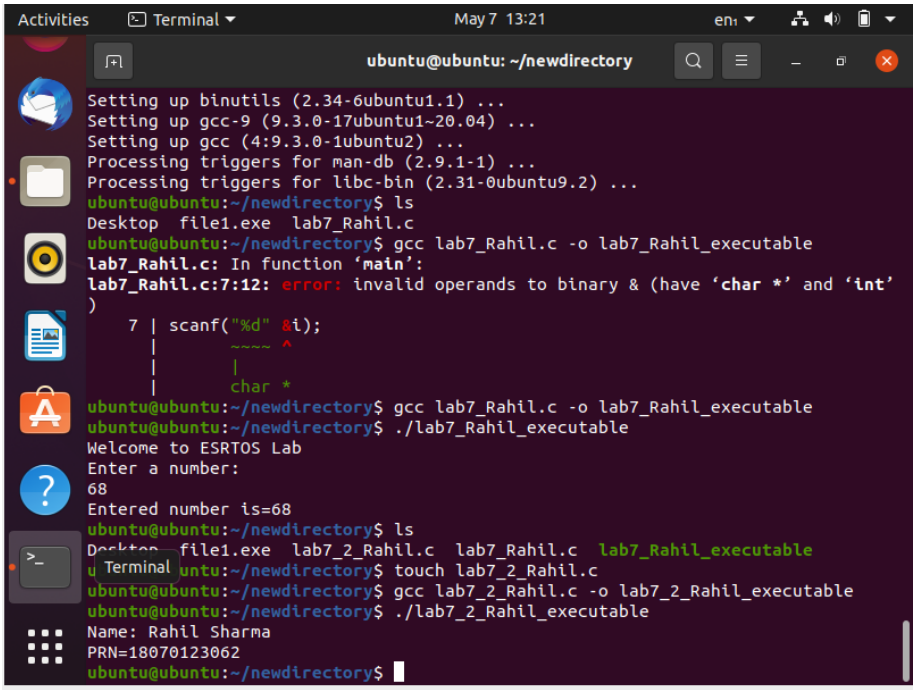


4. Displaying the C program Output in Linux Terminal

Code for C Program:

a)
```
#include <stdio.h>
int main()
{
int i;
printf("Welcome to ESRTOS LAB \n");
printf("Enter a number: \n");
scanf("%d", &i);
printf("Entered number is= %d \n",i);
return 0
}
```

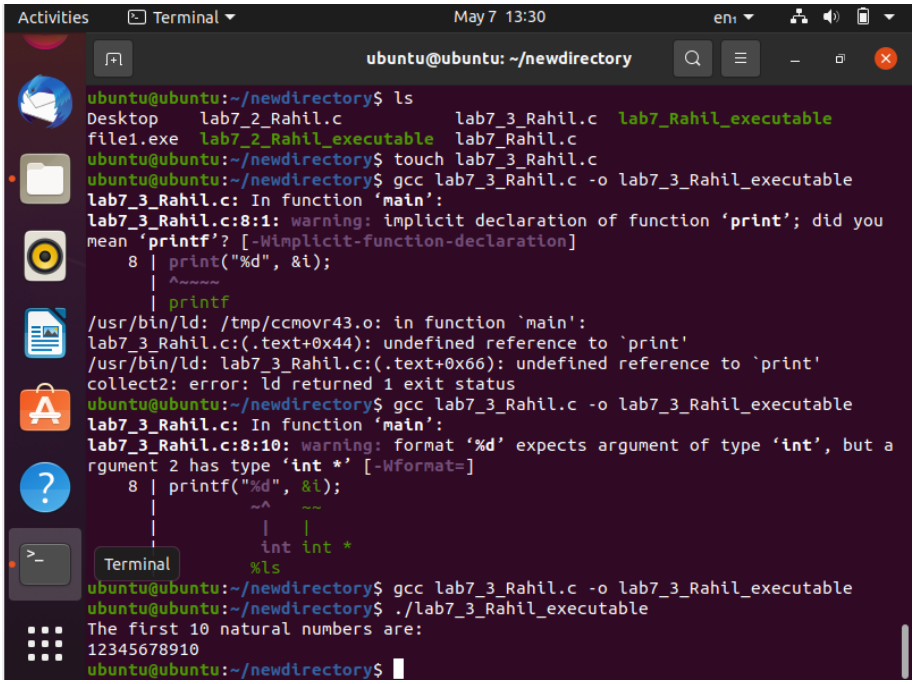
b)
```
#include <stdio.h>
Void main()
{
Printf("Name: Rahil Sharma");
Printf("PRN: 18070123062");
}
```



5. For Loop in C program to be printed in Linux Terminal:

C Program to print first 10 Natural Numbers:

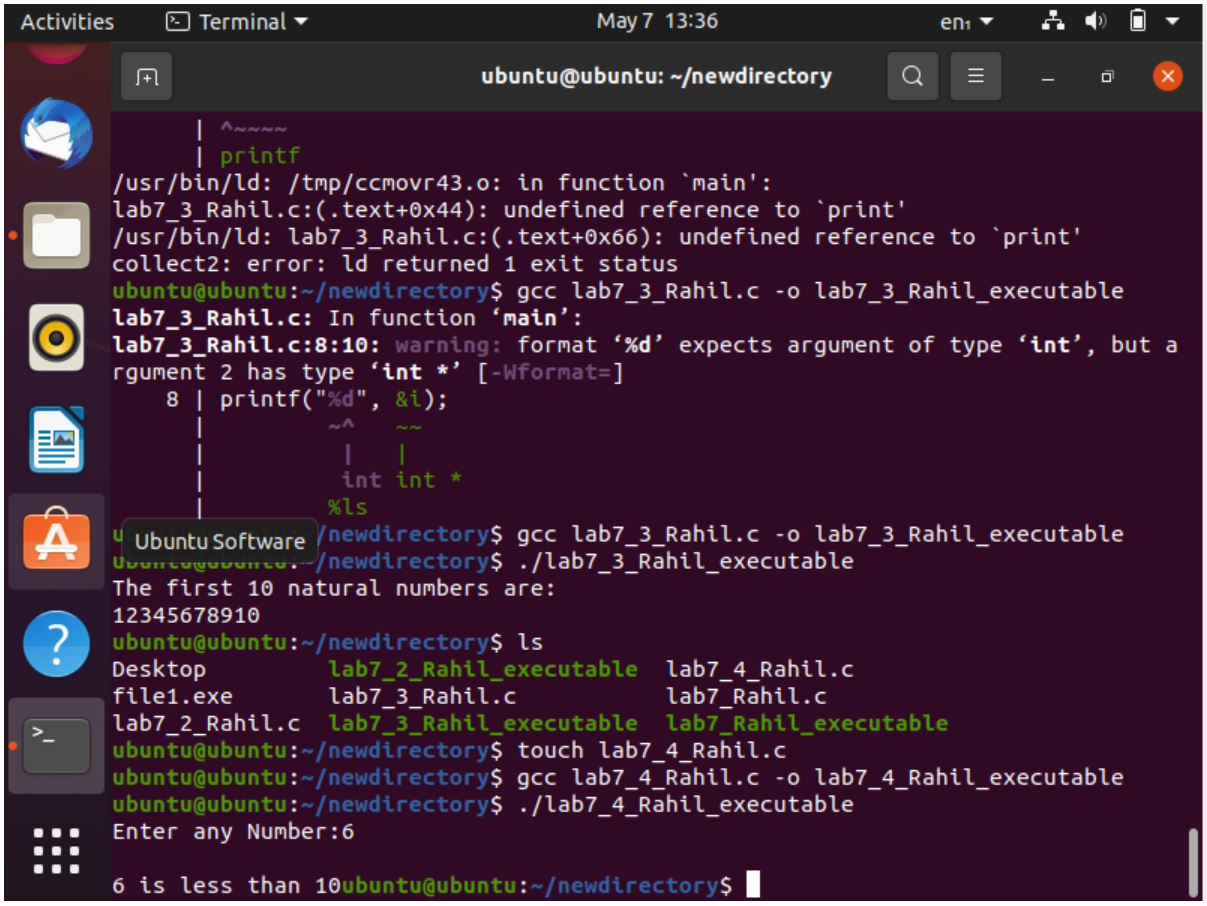
a)
```
#include <stdio.h>
void main()
{
int i;
printf("The first 10 natural numbers are:\n");
for (i=1;i<=10;i++)
{
        printf("%d ",i);
}
printf("\n");
}
```



6. Conditional Program in C:

C Program for displaying output:

```
#include<stdio.h>
int main()
{
    int num;
    printf("Enter Any Number : ");
    scanf("%d",&num);
    if(num<10)
        printf("\n%d is Less Than 10",num);
    return 0;
}
```



Conclusion: From this experiment we have understood how to work on LINUX terminal and C programs and siplay outputs for Conditional Statement programs.