Basic Programming Lab

— — 0x06

Operator Precedence

Operator	Meaning of operator	Associativity
() [] -> .	Functional call Array element reference Indirect member selection Direct member selection	Left to right
! ~ + - ++ -- & * sizeof (type)	Logical negation Bitwise(1 's) complement Unary plus Unary minus Increment Decrement Dereference (Address) Pointer reference Returns the size of an object Typecast (conversion)	Right to left
* / %	Multiply Divide Remainder	Left to right
+ -	Binary plus(Addition) Binary minus(subtraction)	Left to right
<< >>	Left shift Right shift	Left to right
< <= > >=	Less than Less than or equal Greater than Greater than or equal	Left to right

Operator Precedence

Operator	Meaning of operator	Associativity
= !=	Equal to Not equal to	Left to right
&	Bitwise AND	Left to right
^	Bitwise exclusive OR	Left to right
|	Bitwise OR	Left to right
&&	Logical AND	Left to right
||	Logical OR	Left to right
?:	Conditional Operator	Right to left
= *= /= %= += -= &= ^= |= <<= >>=	Simple assignment Assign product Assign quotient Assign remainder Assign sum Assign difference Assign bitwise AND Assign bitwise XOR Assign bitwise OR Assign left shift Assign right shift	Right to left
,	Separator of expressions	Left to right

break & continue & goto

// Program to check working of break, continue & goto

```
#include <stdio.h>
int main()
{
        int i;

        for (i = 1; i < 20; ++i)
        {
                printf("\n%d ", i);
                if ( i % 3 == 0)
                {
                        printf("\n%d is divisible by 3", i);
                        continue;
                }
                if ( i % 7 == 0)
                {
                        printf("\n%d is divisible by 7", i);
                        break;
                        printf("\nNot visible to console");
                }
        }
        printf("\nOut of loop");
        goto label1;
        if ( 7 % 2 == 0)
        {
label1: printf("\nHere");
        }

        return 0;
}
```

Assignment

//0x06

//Use scanf for input in Every Program

1. Write a program to print **factorial** of a number.

2. Write a program to print **HCF** of two numbers.

3. Write a program to find the sum of series

1 + (1+2) + (1+2+3) + (1+2+3+4) **...**

4. Write a program to **count the no of digits** from the input no.

5. Write a program to check whether a no is **palindrome or not.**

Points to Remember

1. Filetype: .c

2. Naming Convention for Directory: Assignment_X
   where X = Lab No
   example: Assignment_1

3. Naming Convention for File: RollNo_Q_Y.c
   where Y = Question No in that Assignment
   example: 123XXX4567_Q_1.c

Commands:

	Command	Example
Create Directory	mkdir <directory_name>	mkdir test_directory
Create File	vi <filename>	vi test.c
Compile a C Program	gcc <filename>	gcc test.c
Run a C Program	./a.out	

4. Write your details in every program

```
/*
 ------------------------------------------------
|Author : Your_Name                              |
|Roll No: Your_Roll_No                           |
|Department: Your_Department                     |
 ------------------------------------------------
*/
```