1.

1.BITWISE OPERATORS

DESCRIPTION:

Bitwise operators are used to perform manipulation of individual bits of a number. They can be used with any of the integral types (char, short, int, etc).

Syntax:

Bitwise AND - &

Bitwise OR- |

Left shift -<<

Right shift ->>

Bitwise NOT -~

Bitwise XOR - ^

EXAMPLE:

& Binary AND Operator (A & B) = 12, i.e., 0000 1100

| Binary OR Operator (A | B) = 61, i.e., 0011 1101

^ Binary XOR Operator (A ^ B) = 49, i.e., 0011 0001

~ Binary One's Complement Operator (~A) = ~(60), i.e,. 1100 0011

<< Binary Left Shift Operator A << 2 = 240 i.e., 1111 0000

>> Binary Right Shift Operator A >> 2 = 15 i.e., 0000 1111

2.TERNARY OPERATOR:

DESCRIPTION:

The ternary operator is an operator that takes three arguments. The first argument is a comparison argument, the second is the result upon a true comparison, and the third is the result upon a false comparison.

SYNTAX:

variable=expression 1?expression 2:expression 3

EXAMPLE:

```
c=a>b? printf("a is greater"):printf("b is greater");
```

2.

```
#include<stdio.h>
int main()
{
int num1=4,num2=5;
printf("sum of 4 and 5=%d\n",4+5);
printf("Difference of 4 and 5=%d\n",4-5);
printf("Multiplication of 4 and 5=%d\n",4*5);
printf("division of 4 and 5=%d\n",4/5);
printf("modulo of 4 and 5=%d\n",4%5);
return 0;
}
```