UNSCHOOL TASK-1

Bitwise Operators:

In Computer Programming a Bitwise operation operates on one or more bit patterns at the level of their individual bits.

There are 6 Bitwise Operators they are

1.AND:The AND operator compares two bits and generates results as 1 if two bits are 1 ,otherwise 0

2.OR:The OR operator compares two bits and generates results as 1 if any one of the bit is 1 or both are 1 ,otherwise 0 i.e two bits are 0

3.NOT:The NOT operator is complement of bits .

4.XOR:The Exclusive-OR operator compares two operands and generates result as 1 if bits are complementary,otherwise 0.

5.SHIFT RIGHT(>>):.The SHIFT RIGHT operator moves bits to right,discards the far right bit,and assigns leftmost bit a value to 0.Each move to the right effectively and divides first operand in half.

6.SHIFT LEFT(<<):The SHIFT LEFT operator moves bits to left,discards the far left bit,and assigns rightmost bit a value to 0.Each move to the left effectively and multiplies first operand by 2.

Examples:

AND:

```
  1010
  1100
 -------
  1000
 -------
```

OR:

```
  1010
  1100
 --------
  1110
 --------
```

NOT:

```
   1001
 -------
  0110
 -------
```

EXCLUSIVE-OR:

```
  0101
  0110
 ------
  0011
 ------
```

SHIFT LEFT:

int a=2<<1;

Let's take the binary representation of 2 assuming int is 1 byte for simplicity.

Position	7	6	5	4	3	2	1	0
Bits	0	0	0	0	0	0	1	0

Now shifting the bits towards left for 1

Now the result in decimal is 4
If you left shift like 2<<2, then it will give the result as 8. Therefore left shifting 1 time, is equal to multiplying the value by 2.

SHIFT RIGHT:

int a=8>>1;
Let's take the binary representation of 8 assuming int is 1 byte for simplicity.

Position	7	6	5	4	3	2	1	0
Bits	0	0	0	0	1	0	0	0

Now shifting the bits towards right for 1 time, will give the following result

Position	7	6	5	4	3	2	1	0
Bits	0	0	0	0	0	1	0	0

Now the result in decimal is 4. Right shifting 1 time, is equivalent to dividing the value by 2.