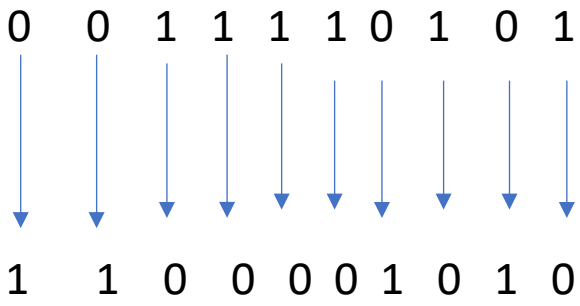
BITWISE COMPLEMENT

The bitwise complement operator is a unary operator .It is denoted by ~.It changes binary digits 1 to 0 and 0 to 1.

Ex:



The bitwise operator should be used carefully. The result of ~ operator on a small number can be a big number if the result is stored in unsigned variable. And the result may be a negative number if the result is stored in the signed variable

EX: input n=2

Binary form of 2=0010

Bitwise complement of 2=~0010

=1101=13(decimal value)

Expected o/p=13

Correct o/p=-3

The compiler returns the 2's complement of the i/p value

The ~2 is -3 instead of 13,but why?

When the numbers are printed in base-10,the result of a NOT operation can be surprising. In particular, positive numbers can become negative and vice versa.

LOGICAL COMPLEMENT

A logical complement operator inverts the value of a Boolean.

Ex: class Demo

```
{
Public static void main(String[] args


s)
{
boolean b=true;
        b=!b;
System.out println(“b=”+b);
}
}
```

o/p: false

REAL NUMBER DATA TYPES

- The upper range of float is 3.4028235E38
- The lower range of float is 1.4E-45.
- The upper range of double is 1.79769313486231 57E308.
- The lower range of double is 4.9E-324.