Fundamentals of Object Oriented Programming

CSN- 103

Dr. R. Balasubramanian

Associate Professor

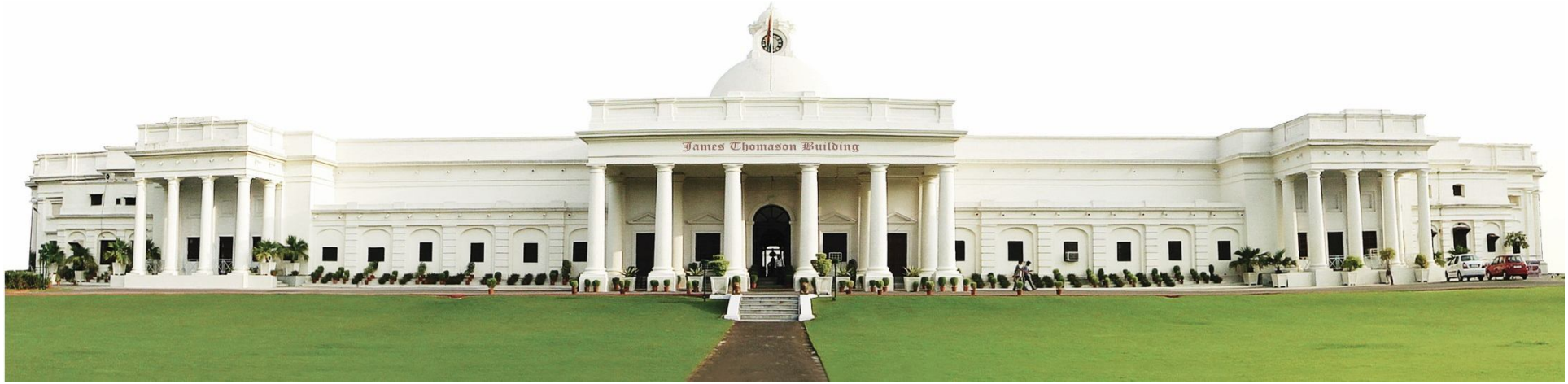
Department of Computer Science and Engineering

Indian Institute of Technology Roorkee

Roorkee 247 667

balarfcs@iitr.ac.in

https://sites.google.com/site/balaiitr/



Logical (Conditional) Operators

```
1  public class Test {
2
3     public static void main(String args[]) {
4        boolean a = true;
5        boolean b = false;
6
7        System.out.println("a && b = " + (a&&b));
8
9        System.out.println("a || b = " + (a||b) );
10
11       System.out.println("!(a && b) = " + !(a && b));
12    }
13 }
```

false

true

true

false

Terminal

```
sh-4.3$ javac Test.java
sh-4.3$ java Test
a && b = false
a || b = true
!(a && b) = true
sh-4.3$
```

!false → true

!true → false

Truth Table

P	Q	P && Q	P || Q	!P
T	T	T	T	F
T	F	F	T	F
F	T	F	T	T
F	F	F	F	T

Short circuit

Assignment Operators

```
Sum=a+b;
```

Shorthand Operators

Operator	Example	Equivalent
+=	i += 8	i = i + 8
-=	i -= 8.0	i = i - 8.0
*=	i *= 8	i = i * 8
/=	i /= 8	i = i / 8
%=	i %= 8	i = i % 8

Increment and Decrement Operators

Operator	Name	Example expression	Meaning
++	Postfix increment	x++	add 1 to x and return the old value
++	Prefix increment	++x	add 1 to x and return the new value
--	Postfix decrement	x--	take 1 from x and return the old value
--	Prefix decrement	--x	take 1 from x and return the new value

```java
class PostIncrement {
  public static void main(String args[]) {
    int num1 = 1;
    int num2 = 1;

    num1++;
    num2++;

    System.out.println("num1 = " + num1);
    System.out.println("num2 = " + num2);
  }
}
```

Terminal

```
sh-4.3$ javac PostIncrement.java
sh-4.3$ java PostIncrement
num1 = 2
num2 = 2
sh-4.3$
```

```java
class PreIncrement {
  public static void main(String args[]) {
    int num1 = 1;
    int num2 = 1;

    --num1;
    --num2;

    System.out.println("num1 = " + num1);
    System.out.println("num2 = " + num2);
  }
}
```

Terminal

```
sh-4.3$ javac PreIncrement.java
sh-4.3$ java PreIncrement
num1 = 0
num2 = 0
sh-4.3$
```

```java
class IncDec {
  public static void main(String args[]) {
    int a = 44;
    int b = 45;
    int c;
    int d;
    c = ++b;   // increment & then assign
    d = a++;   // assign and then increment
    c++;
    System.out.println("a = " + a);
    System.out.println("b = " + b);
    System.out.println("c = " + c);
    System.out.println("d = " + d);
  }
}
```

c → 47 ✓

✓ b → 46, c → 46

✓ a → 45, d → 44

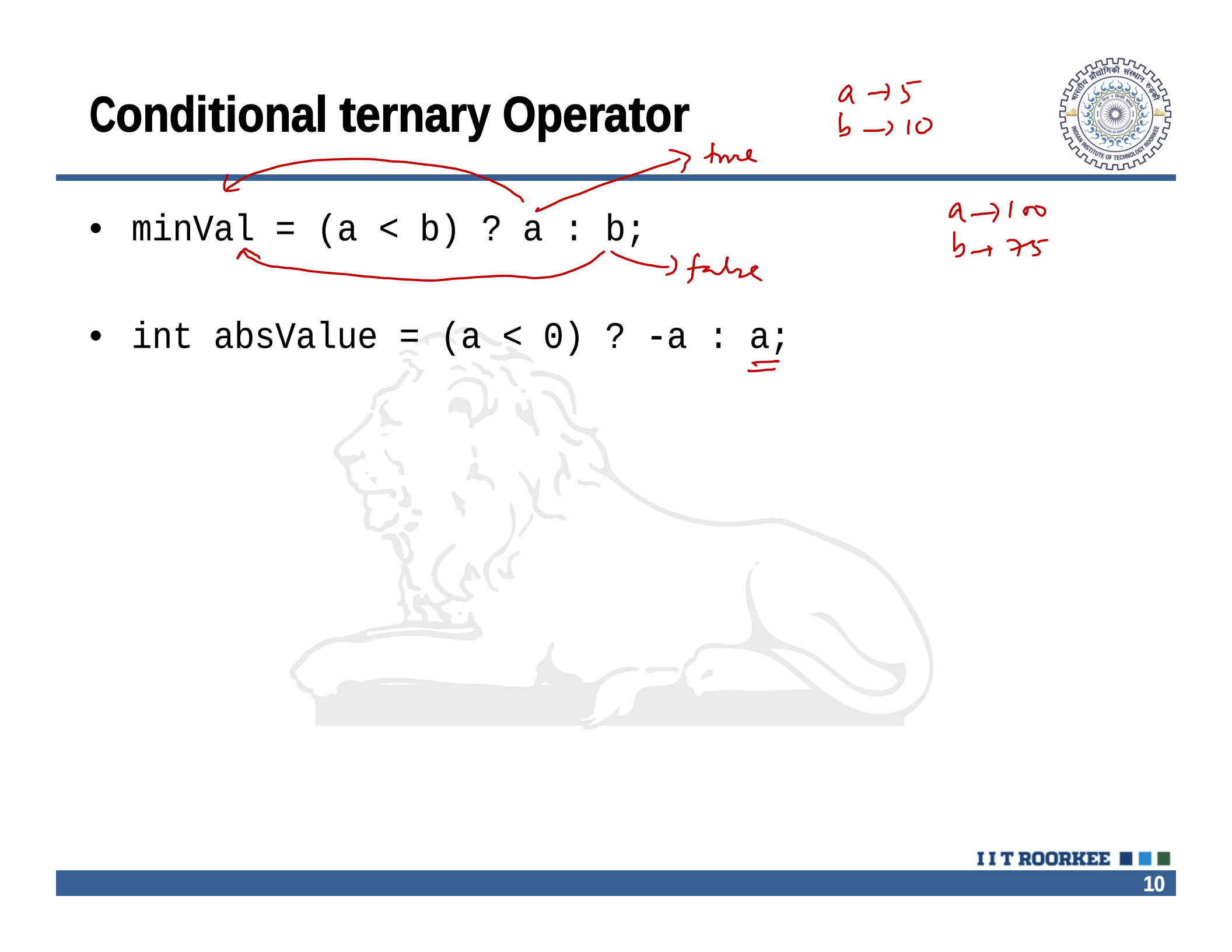
Terminal

```
sh-4.3$ javac IncDec.java
sh-4.3$ java IncDec
a = 45
b = 46
c = 47
d = 44
sh-4.3$
```

```java
class IncDec {
  public static void main(String args[]) {
    int a = 44;
    int b = 45;
    int c;
    int d;
    c = --b;
    d = a--;
    c++;
    System.out.println("a = " + a);
    System.out.println("b = " + b);
    System.out.println("c = " + c);
    System.out.println("d = " + d);
  }
}
```

```
sh-4.3$ javac IncDec.java
sh-4.3$ java IncDec
a = 43
b = 44
c = 45
d = 44
sh-4.3$
```

Conditional ternary Operator



- `minVal = (a < b) ? a : b;`
- `int absValue = (a < 0) ? -a : a;`

```
1  public class Test {
2
3    public static void main(String args[]) {
4       int a , b;
5        a = 10;
6        b = (a == 1) ? 20: 30;
7        System.out.println( "Value of b is : " +  b );
8
9        b = (a == 10) ? 20: 30;
10       System.out.println( "Value of b is : " + b );
11   }
12 }
```

```
Terminal
sh-4.3$ javac Test.java
sh-4.3$ java Test
Value of b is : 30
Value of b is : 20
sh-4.3$
```

Bitwise Operators

Operator	Description
&	Binary AND Operator copies a bit to the result if it exists in both operands.
|	Binary OR Operator copies a bit if it exists in either operand.
^	Binary XOR Operator copies the bit if it is set in one operand but not both.
~	Binary Ones Complement Operator is unary and has the effect of 'flipping' bits.
<<	Binary Left Shift Operator. The left operands value is moved left by the number of bits specified by the right operand.
>>	Binary Right Shift Operator. The left operands value is moved right by the number of bits specified by the right operand.
>>>	Shift right zero fill operator. The left operands value is moved right by the number of bits specified by the right operand and shifted values are filled up with zeros.

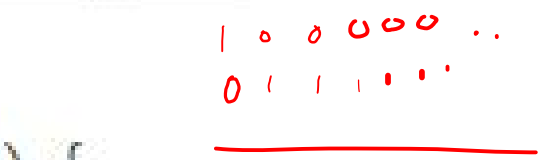
A	B	XOR
0	0	0
0	1	1
1	0	1
1	1	0

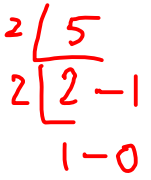
Bitwise Operator

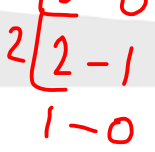
if (a & (a-1) == 0)

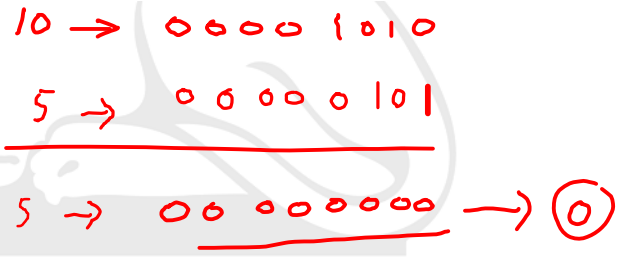
```
1  public class Test {
2
3     public static void main(String args[]) {
4        int a = 10;
5        int b = 5;
6        System.out.println("a & b = " + (a & b) );
7        System.out.println("a | b = " + (a | b) );
8     }
9  }
```

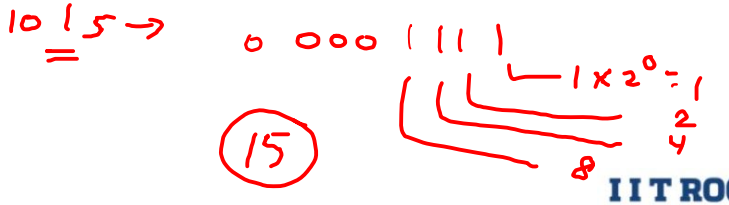
```
Terminal
sh-4.3$ javac Test.java
sh-4.3$ java Test
a & b = 0
a | b = 15
sh-4.3$
```









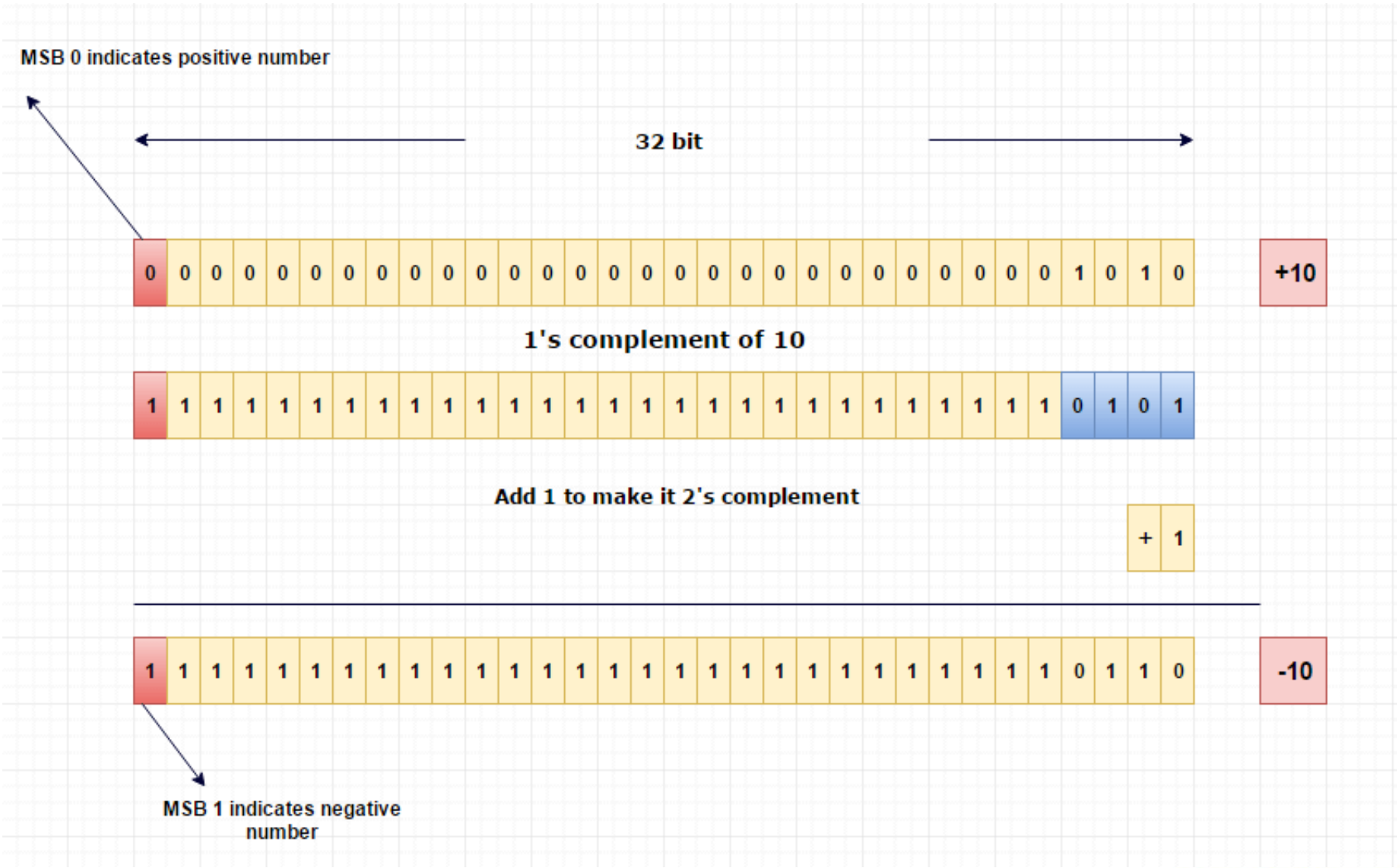


```java
public class Test {

   public static void main(String args[]) {
      int a = 60;     /* 60 = 0011 1100 */
      int b = 13;     /* 13 = 0000 1101 */
      int c = 0;

      c = a ^ b;
      System.out.println("a ^ b = " + c );

      c = ~a;
      System.out.println("~a = " + c );

      c = a << 2;
      System.out.println("a << 2 = " + c );

      c = a >> 2;
      System.out.println("a >> 2  = " + c );

      c = a >>> 2;
      System.out.println("a >>> 2 = " + c );

   }
}
```

Terminal

```
sh-4.3$ javac Test.java
sh-4.3$ java Test
a ^ b = 49
~a = -61
a << 2 = 240
a >> 2  = 15
a >>> 2 = 15
sh-4.3$
```

Binary representation of Negative number



Decimal to Binary illustration

```java
public class Dec2Bin{

    public static void main(String []args){
        byte a=123;
        byte b=5;
        byte c= (byte) (a+b);
        System.out.println(c);
    }
}
```

```
sh-4.4$ javac Dec2Bin.java
sh-4.4$ java Dec2Bin
-128
sh-4.4$
```

https://www.ideone.com/gYreaO

```java
public class Dec2Bin2{

    public static void main(String []args){
        byte a=127;
        byte b=127;
        byte c= (byte) (a+b);
        System.out.println(c);
    }
}
```

```
sh-4.4$ javac Dec2Bin2.java
sh-4.4$ java Dec2Bin2
-2
sh-4.4$
```

https://www.ideone.com/yYq9ZT

```
  1 1 1 1
  1 0 1 0 1
  0 1 1 1 1
-------------
  1 0 0 1 0 0
```

```
2 | 3      2 | 2
  1 - 1      1 - 0
```

① a → 00 11 11 00

000.... 0 00111100

24 times

1's complem 111 ..... 1 11000011

24 times

✓

a → 1100 0011

① 195 − 256 − 61

0 : 1100 0011

1's [1] : 0011 1100

1

2's [1] : 0011 1101 ②

−61

② Find the binary of −30

Step 1 : Find the binary of 30

30_2 → 0 : 0001 1110

1's comp → 1 : 1110 0001

1

2's comp → [1] : 1 1110 0010

Binary of −30 is 111111....[1] : 11100010

....11100010

③ Given $y < 0$ & its binary equivalent is

11010011

Find y.

$211 - 256 = -45$

-45

given $1\ 1010011$

$1^{s}c\ 0\ 0101100$

$+\ 1$

$2^{s}c\ 0\ 0101101$

-45