JAVA HANDS ON PRACTICE

Day 25
04/08/25

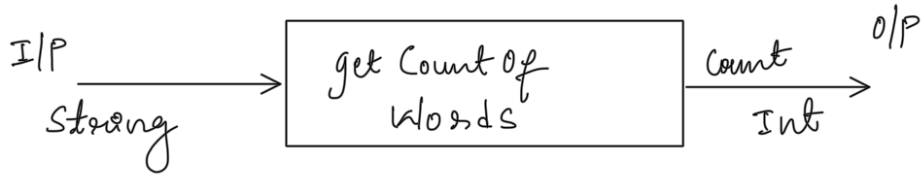
get count of Words In a String

String → "How are you"

word1 word2 word3

H	O	W		a	r	e		y	o	u	\0

Counter → 1 + 1 + 1 = 3




```java
public static int getCountOfWords(String s){

    if(s.length()==0)return 0;

    int count = 0;

    char prev = s.charAt(index:0) ;
    for(int i = 0 ; i<s.length() ; i++){

        char ch = s.charAt(i);

        if(ch == ' ' || ch == '\t' || ch == '\n'){

            if(prev != ' ' && prev != '\t' && prev != '\n' ){
                count++;
            }

        }
        prev = ch ;

    }

    if (prev == ' ' || prev == '\t' || prev == '\n')  return count;

    return count + 1;
}
```

Else return count+1

take one prev characters to trace Weather the prev is space or \t or \n to avoid tailing spaces

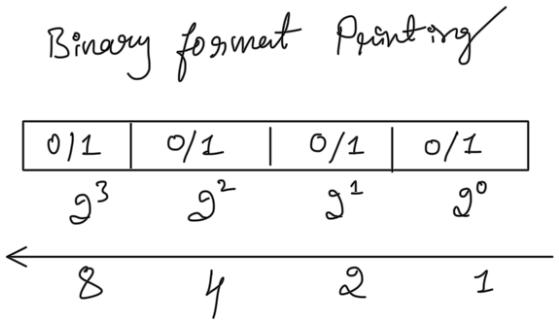
condition to check last characters is " " or \t or \n then last char is tailing Null Space so In that case Just return count

```java
public static void invoke_getCountOfWords() {
    String str = "  Namaskara My LinkedIn Family  ";
    int wordCount = getCountOfWords(str);
    System.out.println("Input string: \"" + str + "\"");
    System.out.println("Word count: " + wordCount);
}
```

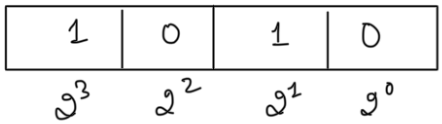
Write a function to print Binary Values of Various input like Integers, also Perform shift operations on them

① Binary format
② left, Right shift

Binary format Printing



Example : 10



1 Active
0 off

$2^3 \times 1 + 2^2 \times 0 + 2^1 \times 1 + 2^0 \times 0$
$8 + 0 + 2 + 0$
$8 + 2$
$= 10$

Binary "&" operation

Multiplication

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

Bit Masking

A	B	A&B
1	0	0
1	1	1
0	0	0
0	1	1

With the help of Bit Masking we can find weather the Bit is Active or Not.

dec = 10 Binary $(1010)_2$

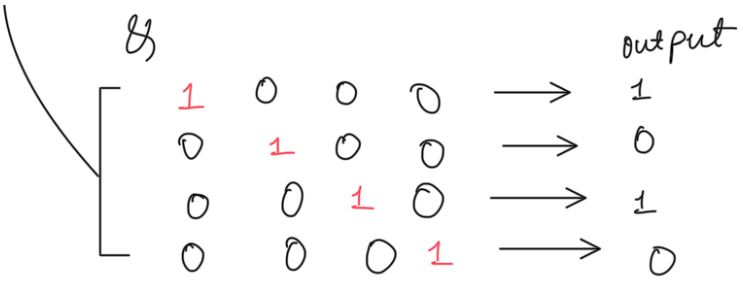
```
  1010
& 1000
------
  1000
------
```

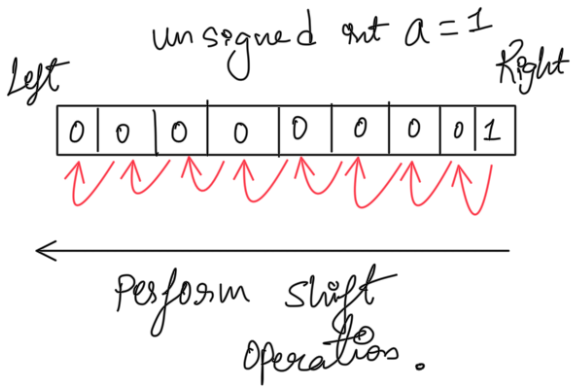
```
  1010
& 0100
------
  0000
------
```

```
  1010
& 0010
------
  0010
------
```

```
  1010
& 0001
------
  0000
------
```

1 0 1 0





Step find Bit size

unsigned int n = 1

n = n << 31

Size of 4 X 8 = 32

32 - 1 = 31

For -ve Integes (2's complement)

1's complement 1 0 1 0

0 1 0 1

Add 1 to 1's complement

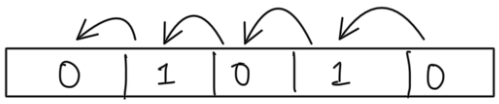
0 1 0 1

Add 1 1

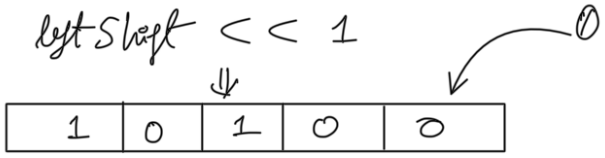
0 1 1 0

0 1 1 0 ⟹ -10

Left shift



left Shift << 1



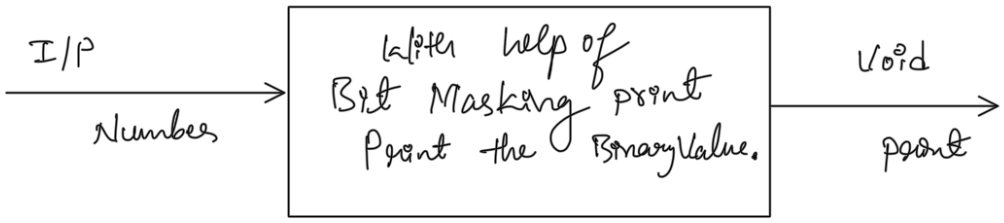
2^4 2^3 2^2 2^1 2^0

$1 \times 2^4 + 0 \times 2^3 + 1 \times 2^2 + 0 \times 2^1 + 0 \times 2^0$

16 + 0 + 4 + 0 + 0 = 20

left shift $\boxed{\text{Number} \times 2^i}$ → where i is Number of Shift

Right shift $\frac{\text{Number}}{2^i}$



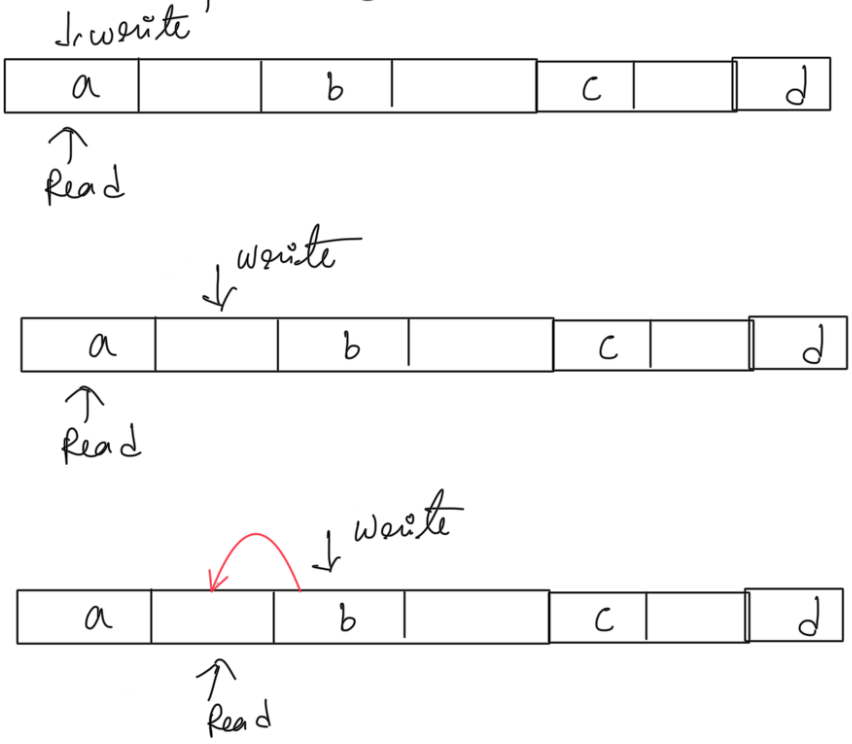

```java
public static void printBinary(int number){

    int mask = 1; // 1 is aleardy present at 1st bit or house
    mask = mask << ((Integer.BYTES * 8) - 2); // leave 1 bit for sign and 1 bit for already standing

    System.out.println("Binary Represntation of "+ number );

    while(mask>0){

        if ((number & mask) == 0) {
            System.out.print(s:"0 ");
        } else {
            System.out.print(s:"1 ");
        }
        mask = mask >> 1 ;
    }
    System.out.println();
    System.out.println();
}

public static void invoke_printBinary() {
    int a = 10;
    int b = -10;
    int c = 0;
    int d = 35;

    printBinary(a);
    printBinary(b);
    printBinary(c);
    printBinary(d);
}
```

```
Word Count: 4
Binary Represntation of 10
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 0

Binary Represntation of -10
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 0
```

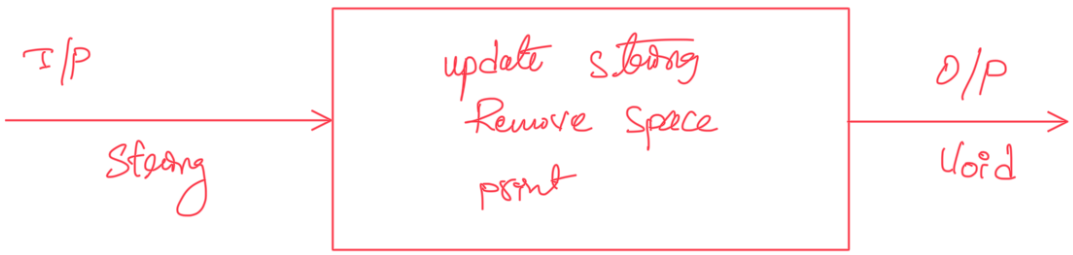
```
Binary Represntation of 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

Binary Represntation of 255
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1
```

Remove Spaces In the String

using Read Write pointes (Two Pointes Approach)



or we can use the New StringBuilder and append Characters



```java
    public static void removespaces(String s){

        StringBuilder sb = new StringBuilder();

        for(int i =0 ; i<s.length();i++){

            char ch = s.charAt(i);

            if(ch == ' ')continue;

            sb.append(ch);
        }

        System.out.println (x:"After Removing spaces ");

        System.out.println(sb.toString());
    }

    public static void invoke_removespaces() {
        String input1 = "My Name Is K Veeresh";
        String input2 = "  Remove all  spaces!  ";

        removespaces(input1);
        removespaces(input2);
    }
```