

MANJULA NANNURI

## DAY-12-Assignment

### Task 1: Bit Manipulation Basics

Create a function that counts the number of set bits (1s) in the binary representation of an integer. Extend this to count the total number of set bits in all integers from 1 to n.

```
package com.example;

public class BitManipulation {

    public static int countSetBits(int n) {

        int count = 0;

        while (n > 0) {

            count += n & 1;

            n >>= 1;

        }

        return count;

    }

    public static int countTotalSetBits(int n) {

        int totalSetBits = 0;

        for (int i = 1; i <= n; i++) {

            totalSetBits += countSetBits(i);

        }

        return totalSetBits;

    }

    public static void main(String[] args) {

        int num = 10;

        System.out.println("Number of set bits in " + num + ": " + countSetBits(num));

    }

}
```

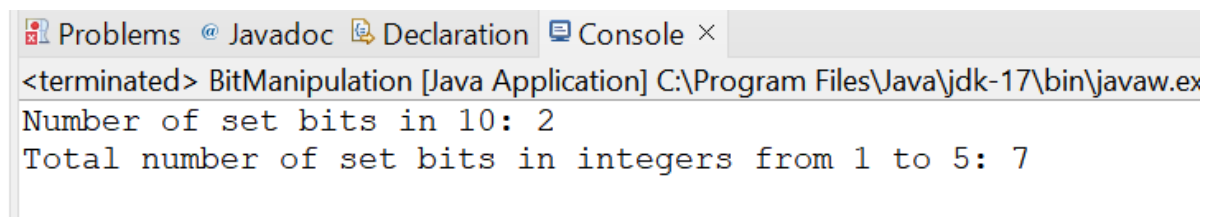
```
int range = 5;
```

```
System.out.println("Total number of set bits in integers from 1 to " + range + ": " +  
countTotalSetBits(range));
```

```
}
```

```
}
```

#### OUTPUT:

A screenshot of a Java IDE's console window. The window has tabs for 'Problems', 'Javadoc', 'Declaration', and 'Console'. The 'Console' tab is active, showing the output of a Java application. The text in the console is: '<terminated> BitManipulation [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe', 'Number of set bits in 10: 2', and 'Total number of set bits in integers from 1 to 5: 7'.

```
<terminated> BitManipulation [Java Application] C:\Program Files\Java\jdk-17\bin\javaw.exe  
Number of set bits in 10: 2  
Total number of set bits in integers from 1 to 5: 7
```

#### Task 2: Unique Elements Identification

Given an array of integers where every element appears twice except for two, write a function that efficiently finds these two non-repeating elements using bitwise XOR operations.

```
package com.example;
```

```
public class UniqueElements {
```

```
public static void findUniqueElements(int[] arr) {
```

```
int xor = 0;
```

```
for (int num : arr) {
```

```
xor ^= num;
```

```
}
```

```
int rightmostSetBit = xor & -xor;
```

```
int x = 0;
```

```
int y = 0;
```

```
for (int num : arr) {
```

```
if ((num & rightmostSetBit) == 0) {
```

```
x ^= num;
```

```
} else {  
  
    y ^= num;  
  
}  
  
System.out.println("The two non-repeating elements are: " + x + " and " + y);  
  
}  
  
public static void main(String[] args) {  
  
    int[] arr = {4, 2, 4, 5, 2, 3, 3, 1};  
  
    findUniqueElements(arr);  
  
}  
  
}
```

#### OUTPUT:

```
<terminated> UniqueElements [Java Application] C:\Program Files\Java\  
The two non-repeating elements are: 1 and 5
```