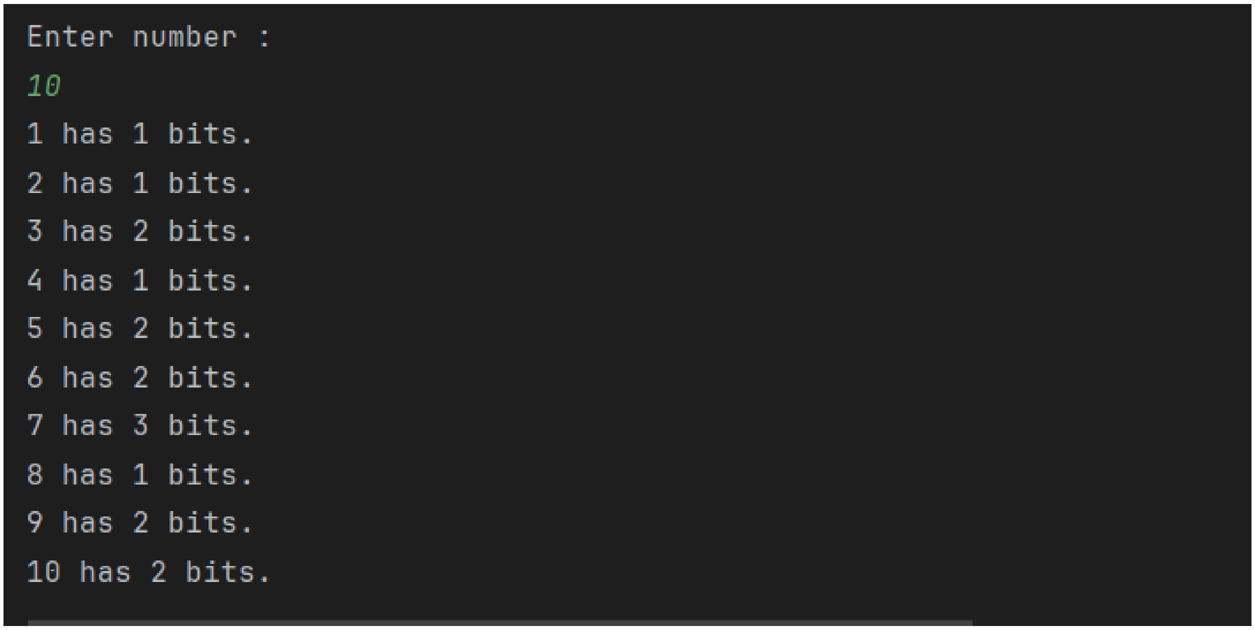
Day 18 Assignment

1. **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.**

```java
package m5_core_java_programming.day_18;

import java.util.Scanner;

/*
   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.
*/
public class Assignment_1 {
   public static int bitCounter(int num) {
       int count = 0;
       while (num > 0) {
           count += ((num & 1) == 1 ? 1 : 0);
           num = num >> 1;
       }
       return count;
   }

   public static void main(String[] args) {
       int num = 0;
       Scanner scan = new Scanner(System.in);
       System.out.println("Enter number : ");
       num = scan.nextInt();
       for (int i = 1; i <= num; i++) {
           int count = bitCounter(i);
           System.out.println(i + " has " + count + " bits.");
       }
   }
}
```

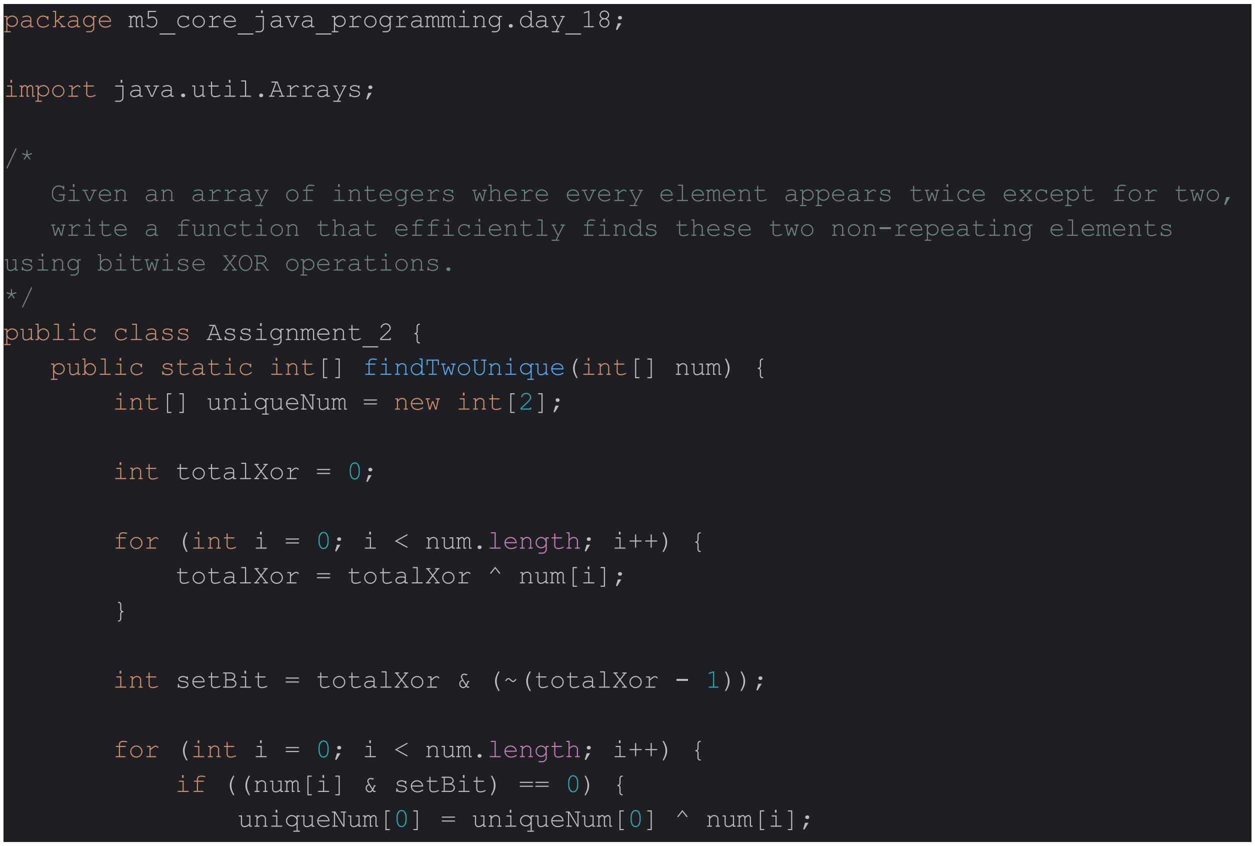
Output



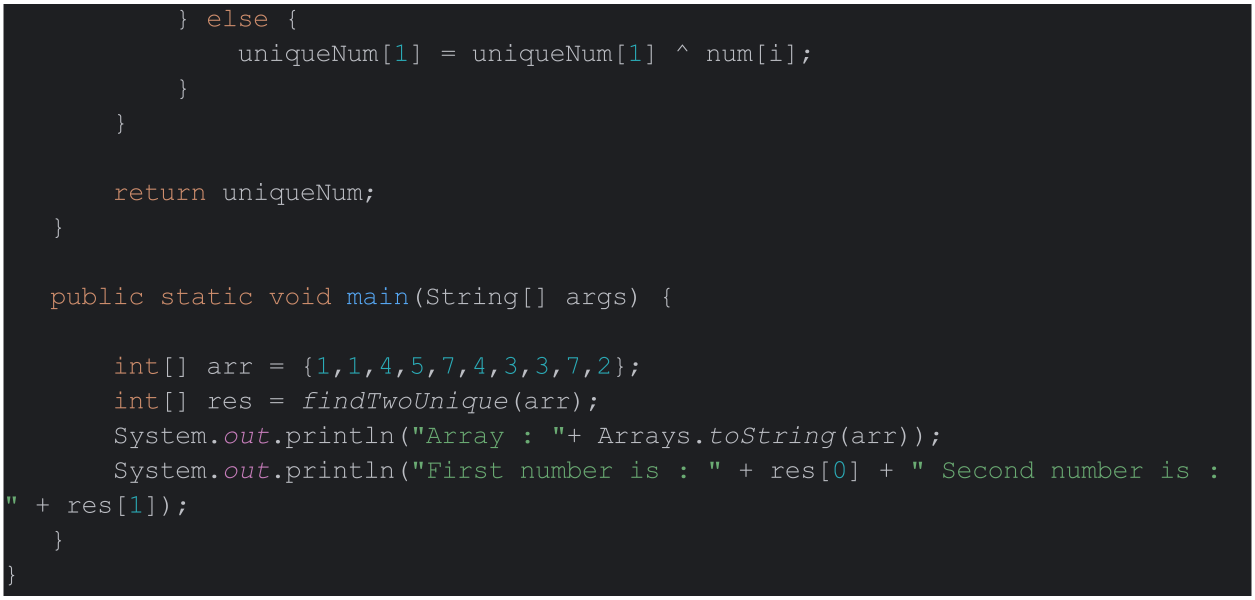
```
Enter number :
10
1 has 1 bits.
2 has 1 bits.
3 has 2 bits.
4 has 1 bits.
5 has 2 bits.
6 has 2 bits.
7 has 3 bits.
8 has 1 bits.
9 has 2 bits.
10 has 2 bits.
```

---

2. 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.

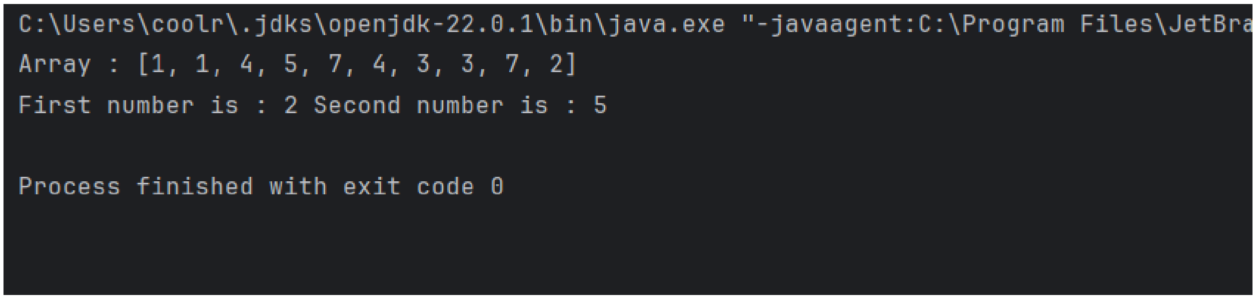


```java
package m5_core_java_programming.day_18;

import java.util.Arrays;

/*
   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.
*/
public class Assignment_2 {
   public static int[] findTwoUnique(int[] num) {
       int[] uniqueNum = new int[2];

       int totalXor = 0;

       for (int i = 0; i < num.length; i++) {
           totalXor = totalXor ^ num[i];
       }

       int setBit = totalXor & (~(totalXor - 1));

       for (int i = 0; i < num.length; i++) {
           if ((num[i] & setBit) == 0) {
               uniqueNum[0] = uniqueNum[0] ^ num[i];
```



```
            } else {
                uniqueNum[1] = uniqueNum[1] ^ num[i];
            }
        }

        return uniqueNum;
    }

    public static void main(String[] args) {

        int[] arr = {1,1,4,5,7,4,3,3,7,2};
        int[] res = findTwoUnique(arr);
        System.out.println("Array : "+ Arrays.toString(arr));
        System.out.println("First number is : " + res[0] + " Second number is :
" + res[1]);
    }
}
```

Output



```
C:\Users\coolr\.jdks\openjdk-22.0.1\bin\java.exe "-javaagent:C:\Program Files\JetBra
Array : [1, 1, 4, 5, 7, 4, 3, 3, 7, 2]
First number is : 2 Second number is : 5

Process finished with exit code 0
```

Tools Used :

IntelliJ IDE
java version "1.8.0_411"
Java(TM) SE Runtime Environment (build 1.8.0_411-b09)
Java HotSpot(TM) Client VM (build 25.411-b09, mixed mode, sharing)