

```
public class StackOverflow {
```

[illegible]

```
public class StackOverflowSolved {
```

```
public static int recursive(int num) {  
    System.out.println("Number: " + num);
```

```

        num++;
        if(num==10){
            return num;
        }
        else{
            return recursive(num);
        }
    }
}

public static void main(String[] args) {
    StackOverflowSolved.recursive(1);
}
}

```

```

C:\Users\Bhuvanesh\Desktop\Secure coding>java StackOverflowSolved
Number: 1
Number: 2
Number: 3
Number: 4
Number: 5
Number: 6
Number: 7
Number: 8
Number: 9
C:\Users\Bhuvanesh\Desktop\Secure coding>

```

3. Write a program to show the problem of Integer Overflow?

```

public class IntegerOverflow{
    public static void main(String[] args) {
        int a=2147483647;
        int b = a+40;
        System.out.println(b);
    }
}

```

```

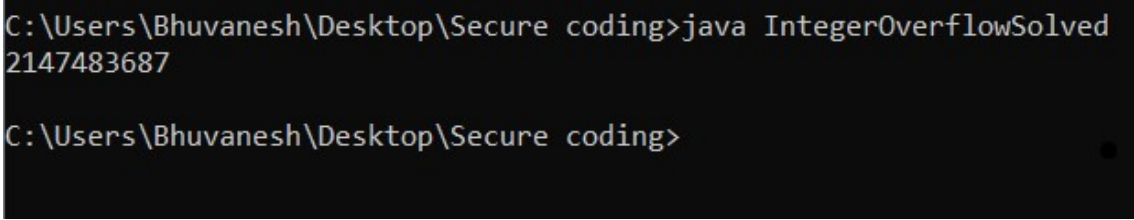
C:\Users\Bhuvanesh\Desktop\Secure coding>java IntegerOverflow
-2147483609
C:\Users\Bhuvanesh\Desktop\Secure coding>

```

Here we are adding 2 positive integers but the output is negative because int cant hold more than 2147483647.

4. Write a program to solve the problem of Integer Overflow?

```
public class IntegerOverflowSolved{  
    public static void main(String[] args) {  
        long a=2147483647;  
        long b = a+40;  
        System.out.println(b);  
    }  
}
```



```
C:\Users\Bhuvanesh\Desktop\Secure coding>java IntegerOverflowSolved  
2147483687  
  
C:\Users\Bhuvanesh\Desktop\Secure coding>
```

Here we are changing int to long so it can hold more than 2147483647 so we are getting positive number.