**SECURE CODING LAB -10**

**STACK OVERFLOW AND INTEGER OVERFLOW ATTACK**

**NAME : SREENIDHI GANACHARI**

**REGISTRATION NUMBER : 19BCE7230**

**SLOT : L – 39+40**

**STACK OVERFLOW** -

**1. Write a program to show problem of Stack Overflow?**

STACK OVERFLOW - A Stack Overflow is a runtime error in java. It is thrown when the amount of call stack memory allocated by JVM is exceeded. A common case of a Stack Overflow being thrown, is when call stack exceeds due to excessive deep or infinite recursion.

**PROGRAM –**

**public** **class** stack\_overflow {

**static** **int** *x* = 0;

**public** **static** **int** printNumber(**int** s)

{

*x*=*x*+5;

System.***out***.println(*x*);

**return** *x* + *printNumber*(*x* + 5);

}

**public** **static** **void** main(String[] args)

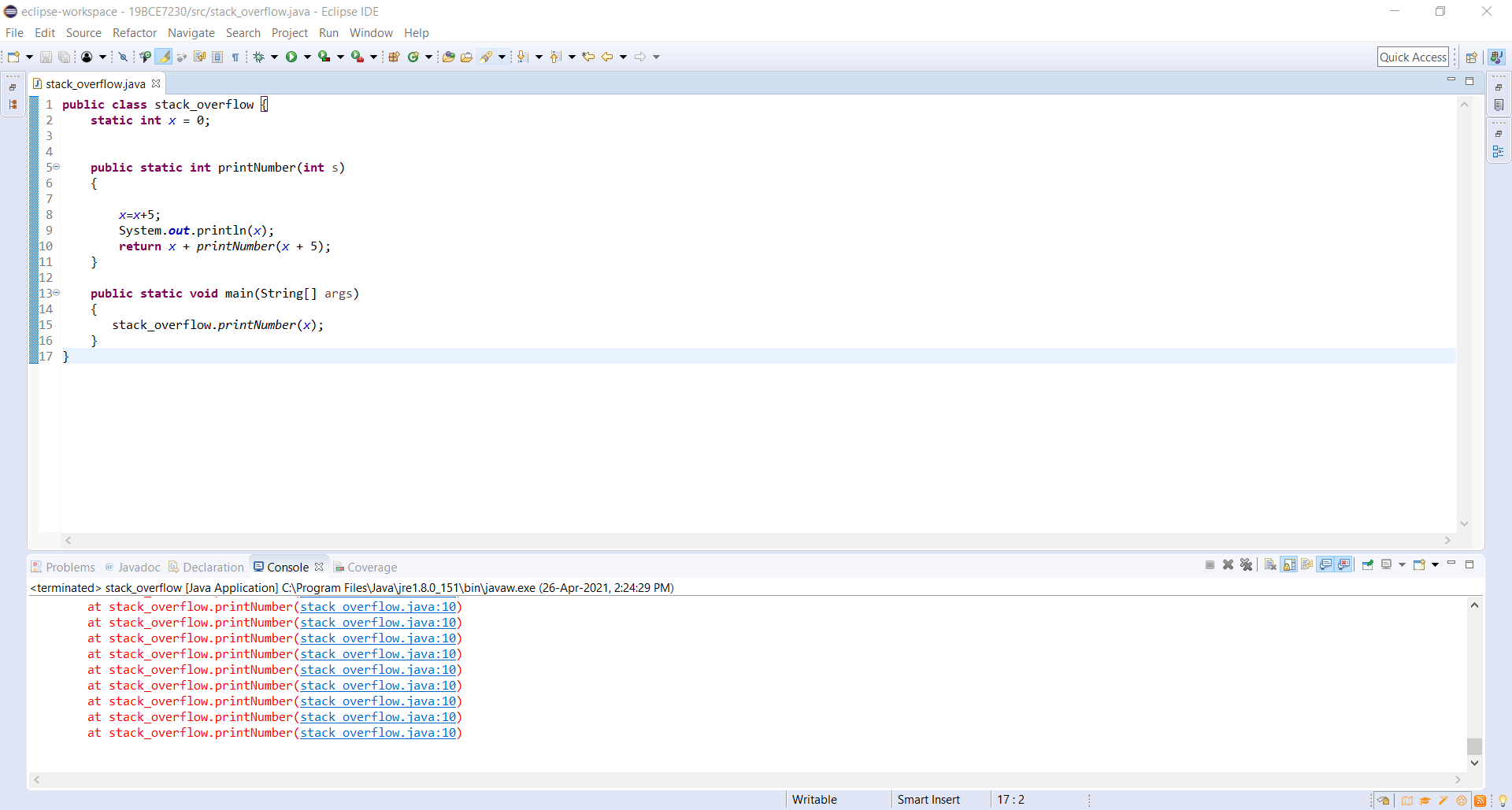
{

stack\_overflow.*printNumber*(*x*);

}

}

**OUTPUT-**



In the above program the recursion runs infinite times and thus it causes stack overflow to occur.

**2. Write a program to show without problem of Stack Overflow?**

To correct the same program above , we do the following changes –

**PROGRAM-**

**public** **class** stack\_overflow {

**static** **int** *x* = 0;

**public** **static** **int** printNumber(**int** s)

{

*x*=*x*+5;

System.***out***.println(*x*);

**if** (*x*==50)

**return** *x*;

**return** *x* + *printNumber*(*x* + 5);

}

**public** **static** **void** main(String[] args)

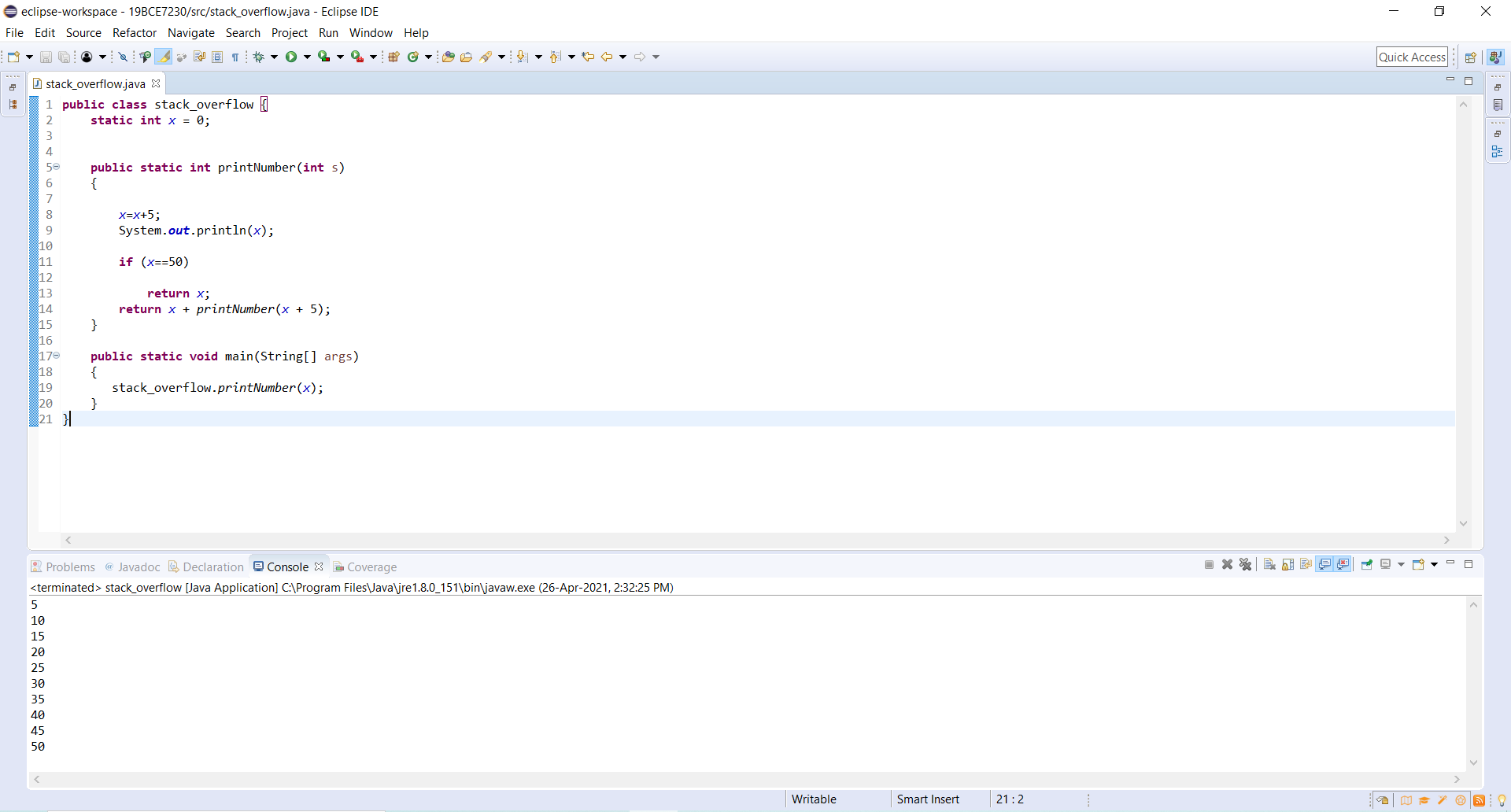
{

stack\_overflow.*printNumber*(*x*);

}

}

**OUTPUT –**



**INTEGER OVERFLOW -**

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

INTEGER OVERFLOW – Integer Overflow means that after incrementing 1 on Integer. MAX\_VALUE (2147483647), the returned value will be -2147483648.

**PROGRAM –**

**public** **class** integer\_overflow {

**public** **static** **void** main(String[] args) {

**int** val1 = 9898989234578904;

**int** val2 = 6789054567777888990;

System.***out***.println("Value1: "+val1);

System.***out***.println("Value2: "+val2);

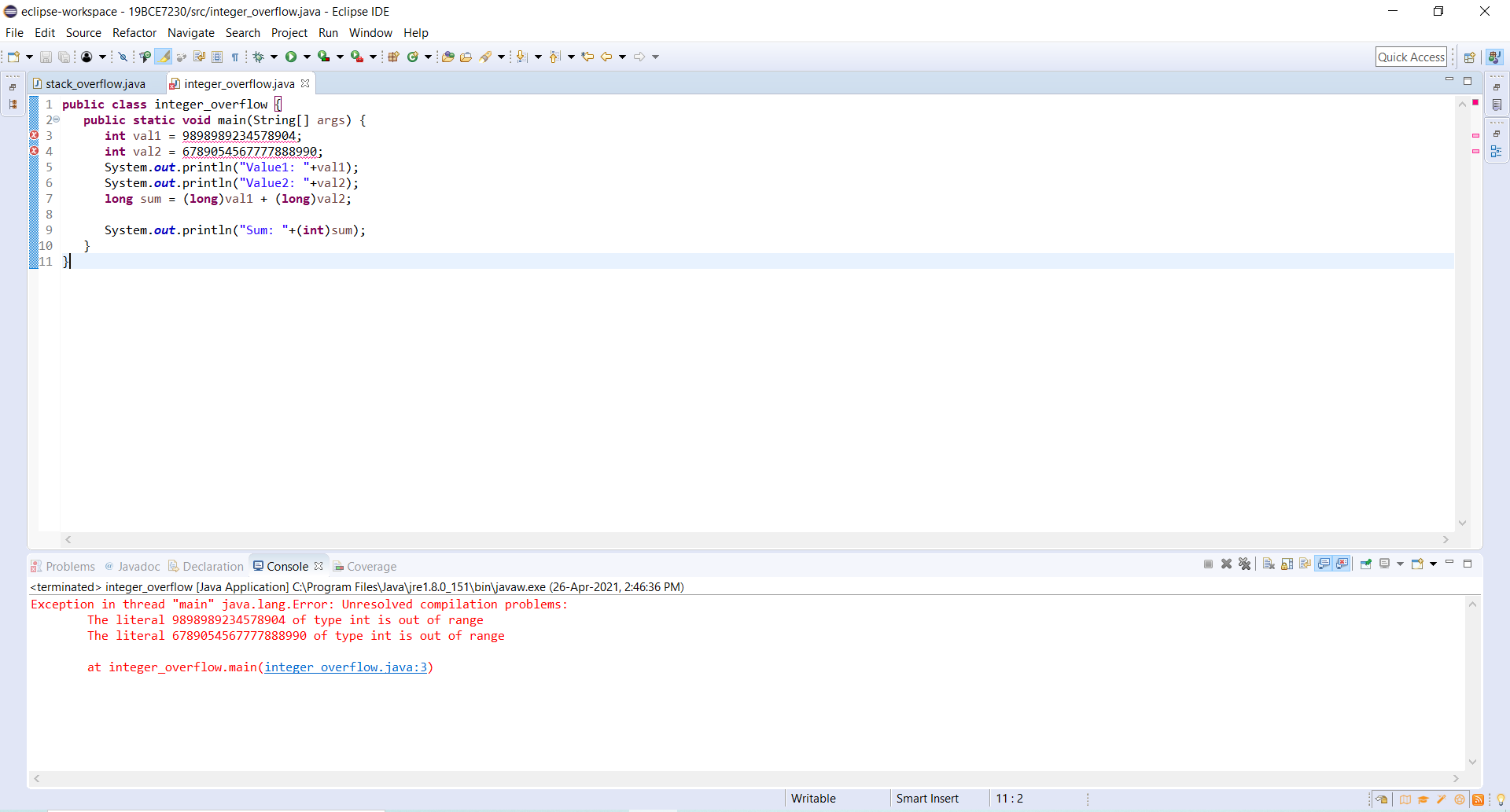
**long** sum = (**long**)val1 + (**long**)val2;

System.***out***.println("Sum: "+(**int**)sum);

}

}

**OUTPUT –**



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

To correct the integer overflow we do the following changes –

**PROGRAM –**

**public** **class** integer\_overflow {

**public** **static** **void** main(String[] args) {

**int** val1 = 9898989;

**int** val2 = 6789054;

System.***out***.println("Value1: "+val1);

System.***out***.println("Value2: "+val2);

**long** sum = (**long**)val1 + (**long**)val2;

**if** (sum > Integer.***MAX\_VALUE***) {

**throw** **new** ArithmeticException("Integer Overflow”);

}

System.***out***.println("Sum: "+(**int**)sum);

}

}

**OUTPUT-**

