**Name : Jacob Nedumattathil Johar**

**ID: 102204195**

**Workshop 1**

**Date: 26 September 2020**

***Task 1***

Design a class named MaxLocation for locating a maximal value and its location in a twodimensional array.

***Logic***

* Declared a class named MaxLocation with members maxVal(double) ,row, column of type int.
* A function maxLoc with parameters a 2D array

=> Declare members m,n,i,j of type int , 1D array locIndex of size 2 of type int

=> Assign the value in index (0,0) to maxVal and assign the length of the array

*maxVal* = value[0][0];

m = value.length;

n = value[0].length;

=> with the help of nested loop unitl i<m and j<n , compute the operation value[i][j] > maxVal

=> Assign the largest value to variable maximal and index to the array declared

**for**(i = 0; i < m; i++)

{

**for**(j = 0;j < n; j++)

{

**if**(value[i][j] > *maxVal*)

{

*maxVal* = value[i][j];

locIndex[0] = i;

locIndex[1]= j;

}

}

}

=> Return the index value

* Main Function

=> Get the input from the user using Scanner library

=> Get the value for Number of rows and columns

Scanner input = new Scanner([system.in](http://system.in));

System.out.println("Enter the number of rows and columns of the array");

row = input.nextInt();

column = input.nextInt();

=> Based on the number of rows and columns entered, input the number of elements

System.out.println("Enter the elements of the array");

for(i = 0; i < row; i++)

{

for(j = 0;j < column; j++)

{

arr[i][j] = input.nextDouble();

}

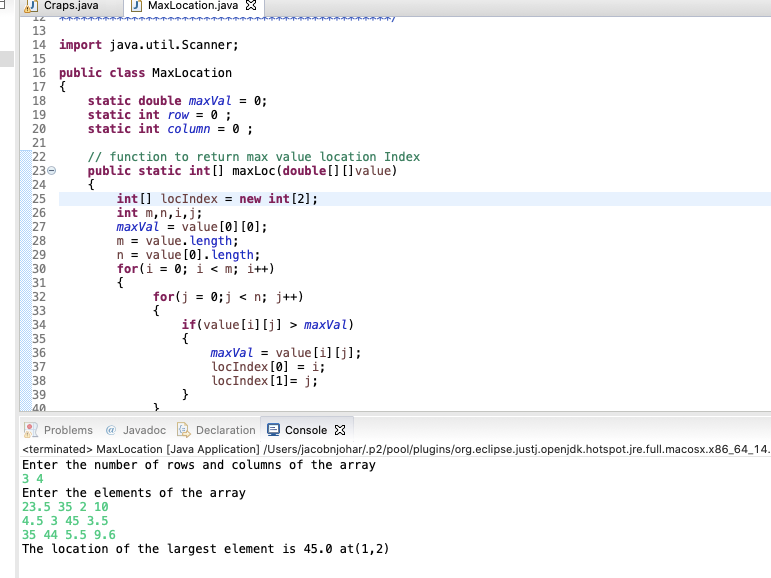
}

=> Call the function maxLoc with the parameter as array

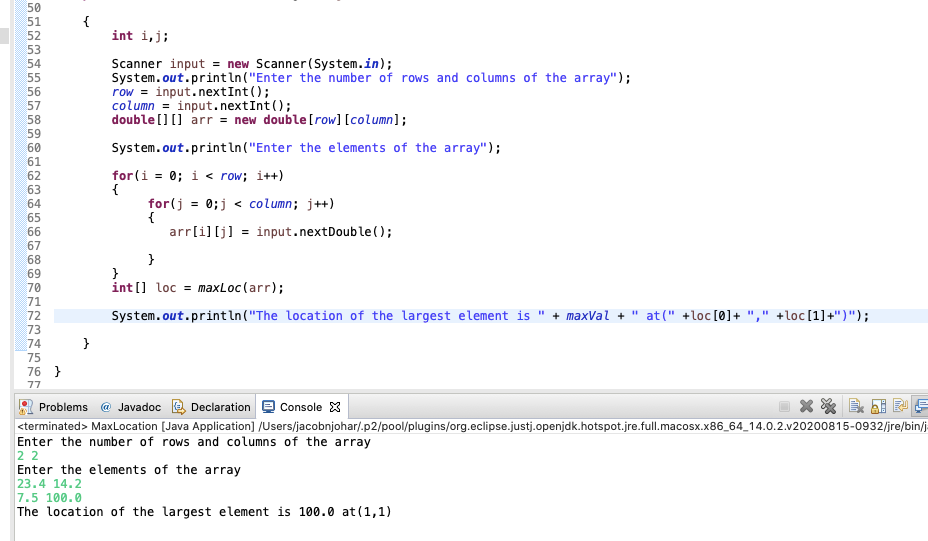
**int**[] loc = *maxLoc*(arr);

***Output***

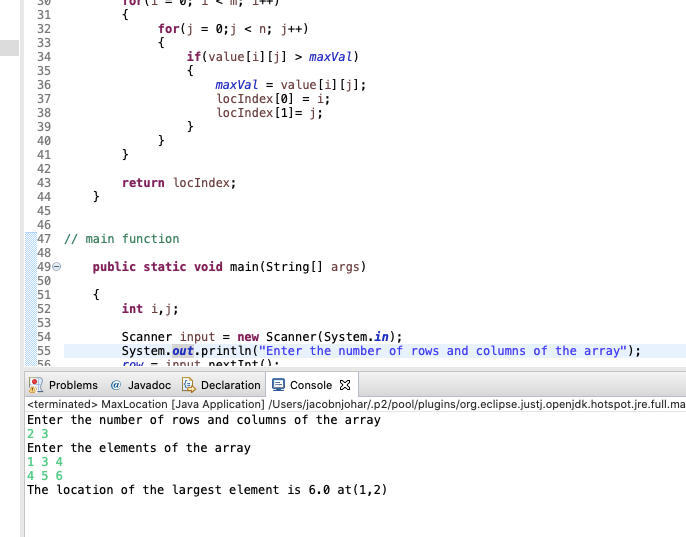
***Test Case 1***

******

***Test Case 2***

******

***Test Case 3***

******

***Task 2***

This task is required to create Craps, which is a popular dice game played in casinos.

***Logic***

* Declare a class named Craps with members

dice1,dice2,total,sum of type int

* A function rolling used for generating random values for dice.

=> The random values between 1 to 6 are generated using Math.random() library

//generating random numbers for dice 1

dice1 = (int) ((Math.random() \* 6) + 1);

//generating random numbers for dice 2

dice2 = (int) ((Math.random() \* 6) + 1);

* A function add to find the consolidated sum of two dice and returns the sum

**int** result;

result = *dice1* + *dice2*;

**return** result;

* Main Function

=> Inside an outer do while loop we are handling each conditions

=> We will invoke the function rolling( generate random values) and add (get the sum of values generated.

=> If the sum is either7 or 11 , the player wins

if (sum == 7 || sum == 11)

{

System.out.println("Congratulations You won");

}

=> if the sum is either 2, 3 or 12 , the player loose the game

else if (sum == 2 || sum == 3 || sum == 12)

{

System.out.println("Craps, Better Luck Next Time, You lose");

}

=> if the sum is either 4,5,6,8,9,10

=> the player will have the chance to throw dice until sum becomes to 7

=> if the player get the latest sum and the previous sum same, he owns

else if (sum == 4 || sum == 5 ||sum == 6 || sum == 8 ||sum == 9 || sum == 10)

{

System.out.println("You rolled " + dice1 + "+" +dice2+ "=" + sum);

System.out.println("Point is (established) set to " + sum );

do

{

rolling();

total = add();

System.out.println("You rolled " + dice1 + "+" +dice2+ "=" + total);

if(total == sum)

{

System.out.println("Congratulations, You Win");

break;

}

else if(total == 7)

{

System.out.println("Craps, Better Luck Next Time, You Lose");

break;

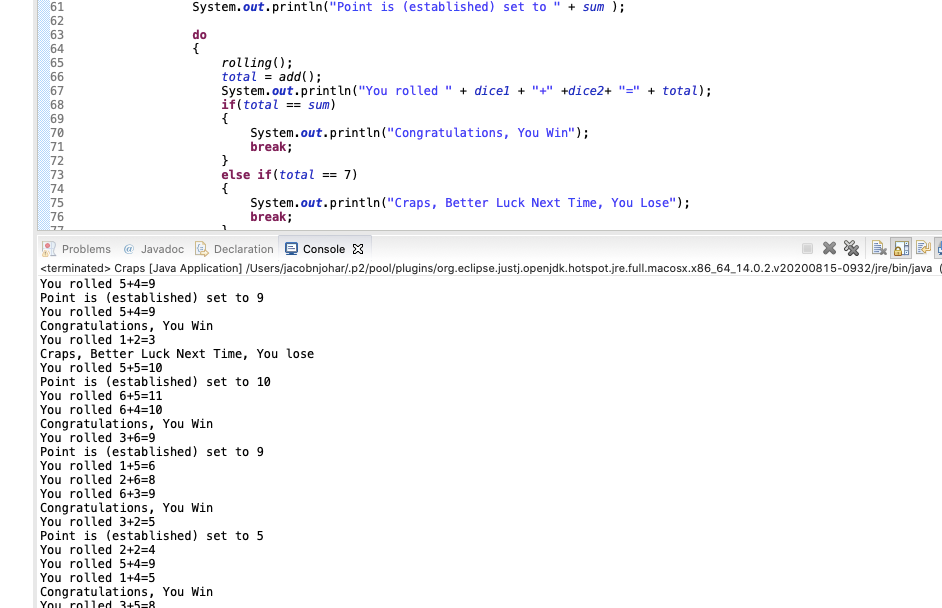
}

}while(sum != 7);

}

***Test Case1***

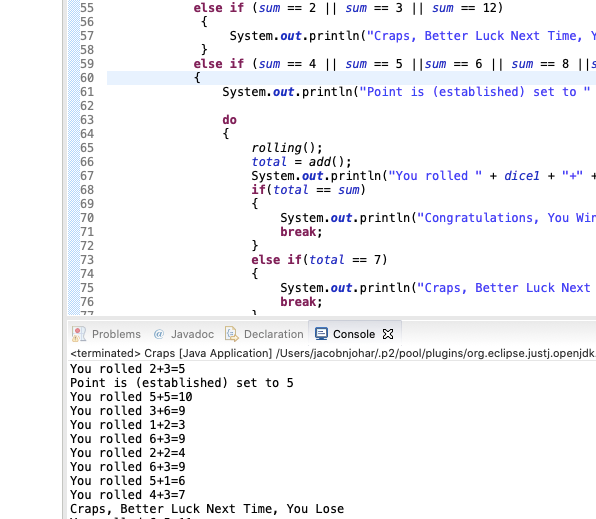
***Condition :***

Initially got sum as 9, then the player got chance to roll again and got the sum 9 ie latest sum equal to initial sum. Therefore, the player wins******

***Test Case 2***

***Condition:***

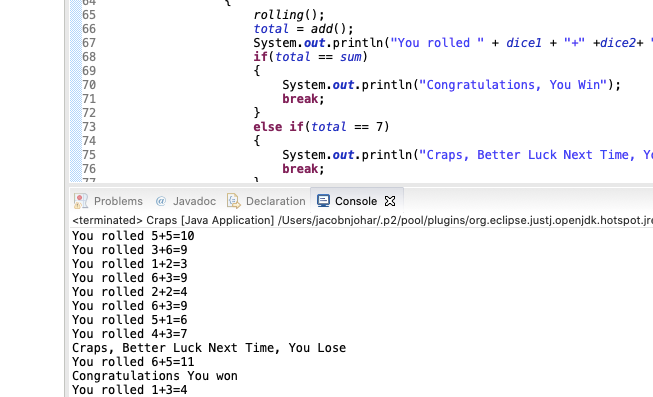
Initially got sum as 5, then the player got chance to roll again and got the sum 7 .Therefore, the player looses the game

******

***Test Case 3***

***Condition:***

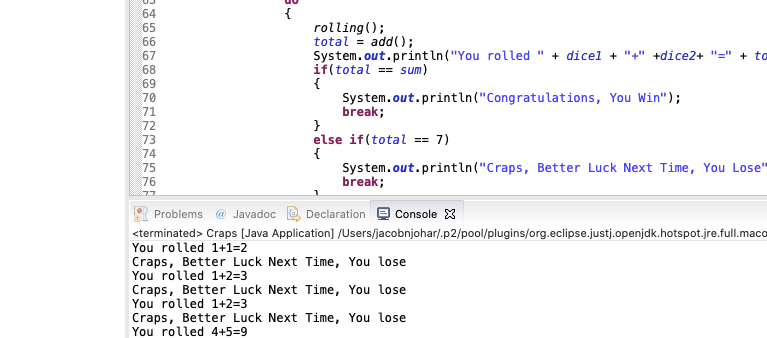
User get the sum 11. Therefore the player wins the game

******

***Test Case 4***

***Condition:***

User get the sum either 2 or 3. Therefore the player loose the game

******