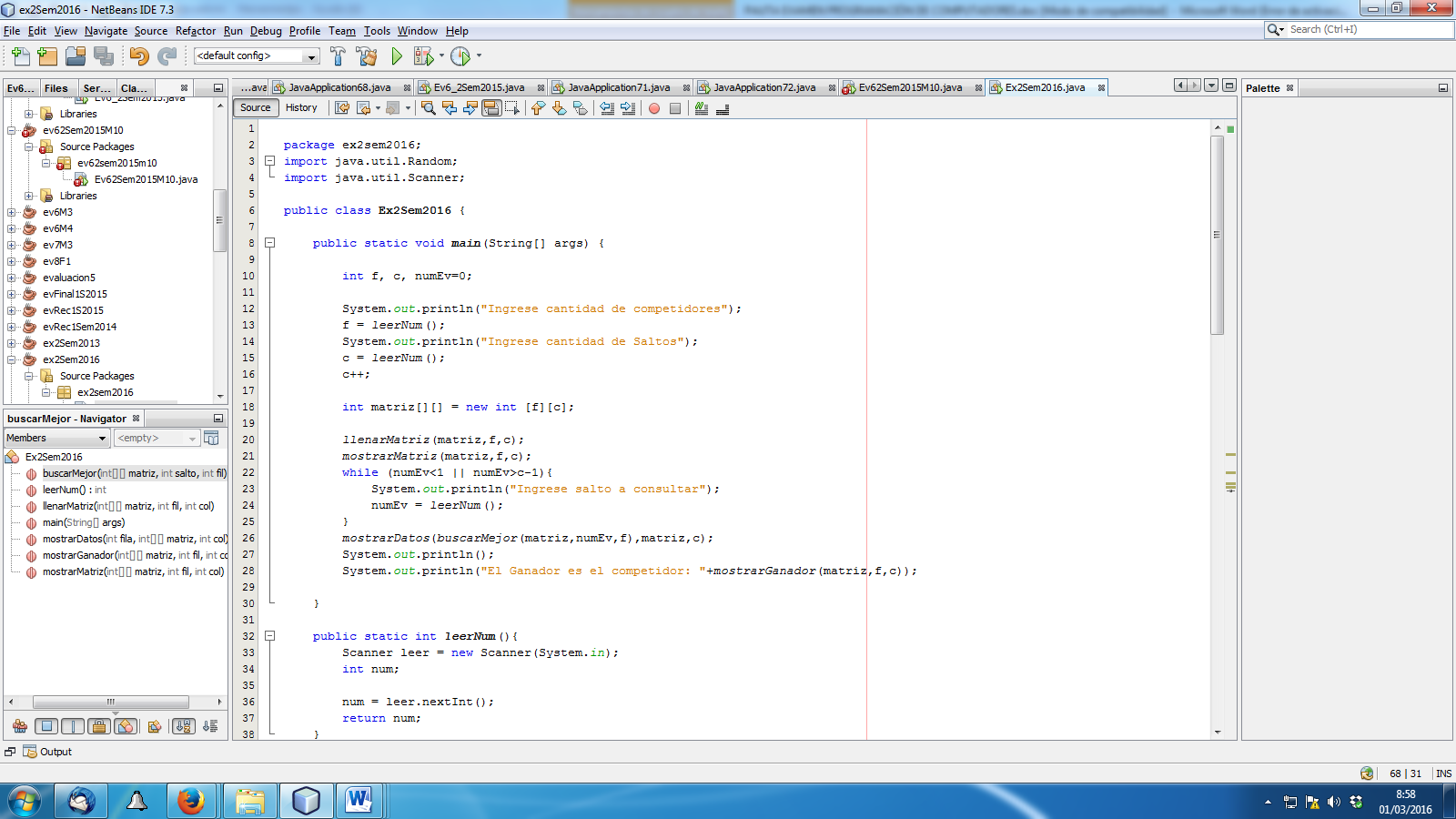
***EJERCICIO FUNCIONES Nº 1***

Suponga que desea construir un programa que lleve el registro de una competición de natación. Suponga que tanto la cantidad de participantes como el número de saltos a realizar por cada participante es Ingresados por teclado al principio de la competencia. Por cada salto, el participante, recibe un puntaje que puede estar entre 0 y 10. El ganador será quien sume más puntos en todos sus saltos. Para llevar el registro use una tabla, donde la primera columna corresponde al número del competidor (correlativo desde 1 hasta N) y el resto de las columnas corresponde a los saltos.

Para facilitar su trabajo, en la Figura 1 le indicamos el “main” del programa, por lo que solo debe construir las funciones faltantes. NO debe modificar la clase principal a menos que utilice datos globales.

***Fig. 1: Código de la función principal del programa.***



El programa debe entregar como respuesta:

1. El competidor que realizó el mejor puntaje de un salto consultado por el usuario.
2. El Ganador de la competencia.
3. Los puntos de todos los saltos para cada jugador (Tabla completa)

Funciones:

1. leerNum(): Función que lee desde teclado un número entero.
2. llenarMatriz(): Función que llena la matriz con el número del competidor y los puntajes de sus saltos. Puede utilizar números aleatorios para el ingreso de los datos.
3. mostrarMatriz(): Función que muestra la tabla de registro de puntajes.
4. buscarMejor(): Función que busca el mejor puntaje de una salto indicado por el usuario y entrega el competidor que lo realizó.
5. mostrarDatos(): Función que muestra los datos de un determinado competidor.
6. mostrarGanador(): Función que muestra al ganador de la competición

*Importante:*

1. *Asuma el enunciado como cierto, no realice mejoras al programa principal (a menos que trabaje con datos globales) ni modificaciones en las especificaciones de las funciones.*

***RESPUESTA***

package ex2sem2016;

import java.util.Random;

import java.util.Scanner;

public class Ex2Sem2016 {

public static void main(String[] args) {

int f, c, numEv;

System.out.println("Ingrese cantidad de competidores");

f = leerNum();

System.out.println("Ingrese cantidad de Saltos");

c = leerNum();

c++;

int matriz[][] = new int [f][c];

llenarMatriz(matriz,f,c);

mostrarMatriz(matriz,f,c);

do{

System.out.println("Ingrese evento");

numEv = leerNum();

}while(numEv<1 || numEv>c-1);

mostrarDatos(buscarMejor(matriz,numEv,f),matriz,c);

System.out.println();

System.out.println("El Ganador es el competidor: "+mostrarGanador(matriz,f,c));

}

public static int leerNum(){

Scanner leer = new Scanner(System.in);

int num;

num = leer.nextInt();

return num;

}

public static void llenarMatriz(int [][] matriz, int fil, int col){

int i,j;

Random azar = new Random();

for(i=0;i<fil;i++){

matriz[i][0] = i + 1;

for(j=1;j<col;j++){

matriz[i][j] = azar.nextInt(11);

}

}

}

public static void mostrarMatriz(int [][] matriz, int fil, int col){

int i,j;

for(i=0;i<fil;i++){

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

System.out.print(matriz[i][j]+"\t");

System.out.println();

}

}

public static int buscarMejor(int [][] matriz,int evento, int fil){

int i, mayorPtje=0, mejorPart=0;

for(i=0;i<fil;i++){

if(matriz[i][evento]>=mayorPtje){

mayorPtje = matriz[i][evento];

mejorPart = i;

}

}

return mejorPart;

}

public static void mostrarDatos(int fila,int [][] matriz,int col){

int j;

for(j=0;j<col;j++){

System.out.print(matriz[fila][j]+"\t");

}

}

public static int mostrarGanador(int [][] matriz, int fil, int col){

int i,j, sumaPts, mayorPtje=0, ganador=0;

for(i=0;i<fil;i++){

sumaPts = 0;

for(j=1;j<col;j++){

sumaPts += matriz[i][j];

}

if (mayorPtje<sumaPts){

mayorPtje = sumaPts;

ganador = i;

}

}

return ganador;

}

}