

Universidad de Costa Rica
CI-0112 Programación 1
Escuela de Ciencias de la Computación e
Informática
Tarea Programada 1
Arquería
Profesor: Dr. Edgar Casasola
Integrantes:
Sebastian Gonzalez Varela
Esteban Quesada Quesada
Steven Núñez Murillo
Fecha de entrega:
29 de septiembre del 2019

1-) Descripción del problema :En un rango de arquería, se está llevando a cabo una competencia. Esta consiste en partidas de dos arqueros. Cada partida se compone de uno o más sets. Los arqueros se toman turnos haciendo tiros a un blanco fijo. A cada tiro se le asigna un puntaje. Se completa un set cuando ambos jugadores han hecho una cantidad de tiros que se define para toda la partida. Por lo tanto, todos los sets de una misma partida tienen exactamente la misma cantidad de tiros. El jugador que obtiene el mayor puntaje gana el set. Se completa una partida cuando se han jugado todos los sets predeterminados para esa partida. El jugador con la mayor cantidad de triunfos en sets individuales es el ganador. Tendrán que elaborar una aplicación que simule la situación anterior, basado en las varias especificaciones.

2-) Descripción general de la implementación

Main: la clase main es la encargada de llamar al menú y hacer que el programa funcione, solo usa un método void para correr el menú

Menú: El menú se encarga de pedirle al usuario que digite una opción por medio de un método llamado SolicitarOperacionARealizar, si el usuario digita un 0 se cierra el programa, si digita 3 puede cambiar los sets, si digita 4 puede cambiar los tiros, si digita 2 puede ver la puntuación más alta y si digita 1 puede jugar.

Configuración de Partida: Para modificar la cantidad de sets de la partida la clase Configuración tienen un método Configurar Sets y para modificar la cantidad de tiros de la partida la clase Configuración tienen un método Configurar Tiros, ambos métodos piden la cantidad de sets o tiros que quiere el usuario y actualiza la partida.

Juez: Primero el juez es el responsable de primero usar el método tirar moneda para determinar cuál jugador es el primero en tirar, el juez tiene dos métodos de localizar j1 y localizar máquina que se fija en el blanco y según el punto del tiro le asigna un puntaje a cada jugador, el juez tiene métodos de registrar los puntajes de los jugadores y luego compararlos para asignar un ganador en cada set, al final de la partida determina un ganador y muestra las estadísticas de los tiros para cada jugador

Registro de Puntuación: La aplicación debe registrar la mayor puntuación, para eso se crea la clase Registro de Puntuación que tiene un método para cambiar la puntuación en caso de que un jugador supere la puntuación más alta actual, también tiene un método para mostrar la puntuación más alta y otro método que borra la puntuación más alta en caso de que el usuario cambie los sets o los tiros.

Jugadores:

Viento: los tiros de los jugadores deben de ser desviados por el viento, para eso se crea la clase viento que al ser construida se le atribuye un tipo de viento y con cada tiro de jugador varía la intensidad, para eso el viento tiene un método que calcula una variación según su tipo y la dirección se hace al crear el viento, además tiene dos métodos para recibir las coordenadas del tiro del jugador y lo desvía.

Blanco o Diana: Después de que el tiro de un jugador fue desviado por viento tiene que pegar o no en el blanco, la clase blanco tiene 3 métodos de localización en una diana grande, mediana y pequeña según el tiro del jugador

4-) Manual de Usuario:

Si el usuario digita un 0 se cierra el programa

Si digita 3 puede cambiar los sets, el programa le pedirá ingresar un 0 para volver al menú o un 1 para cambiar los sets y luego le pedirá la cantidad de sets

Si el, si digita 4 puede cambiar los tiros, el programa le pedirá ingresar un 0 para volver al menú o un 1 para cambiar los tiros y luego le pedirá la cantidad de tiros

Si digita 2 puede ver la puntuación más alta

Si digita 1 puede jugar, el programa le pedirá las coordenadas a las que quiere tirar .

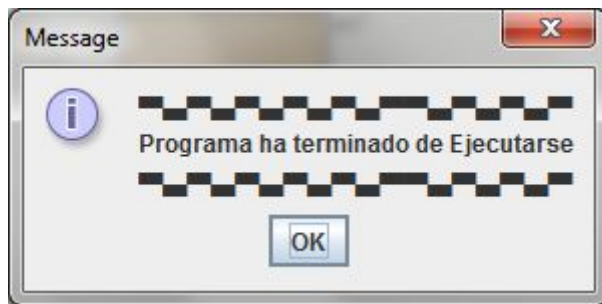
El programa trabaja con una clase main que al correr hace funcionar todo el programa y las demás clases deben estar compiladas (Viento, Registro de Puntuación, Configuración de Partida, Configuración de Partida, Menú).

El programa sirve con modalidad gráfica de JOptionPane, porque se puede correr por medio de un .jar o bien compilando (creando) las clases antes mencionadas y ejecutando el main.

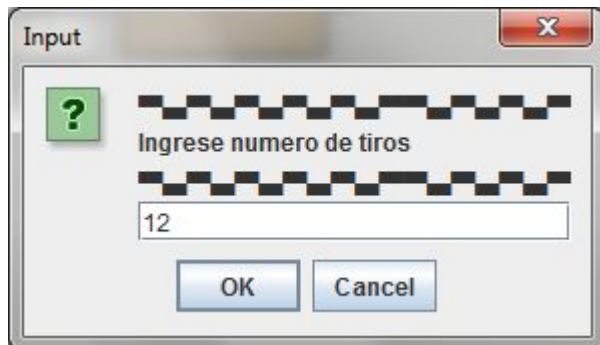
5-) Casos de Prueba:



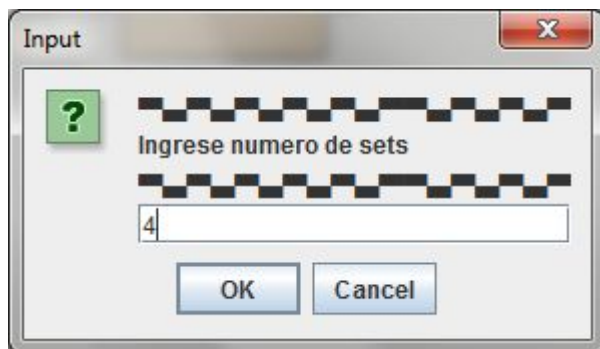
En el menú se pueden poner 4 opciones, cada una hace algo diferente.



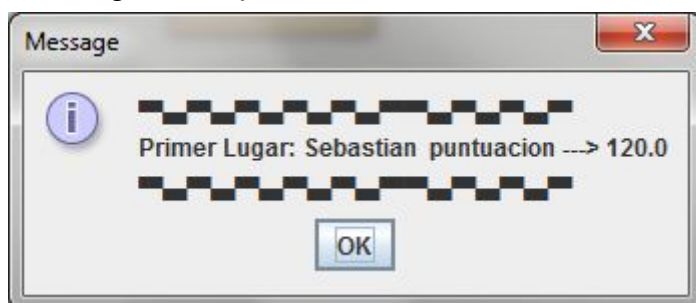
Si se digita 0 el programa terminará de ejecutarse



Si se digita 4 se pueden cambiar los números de tiros por partida



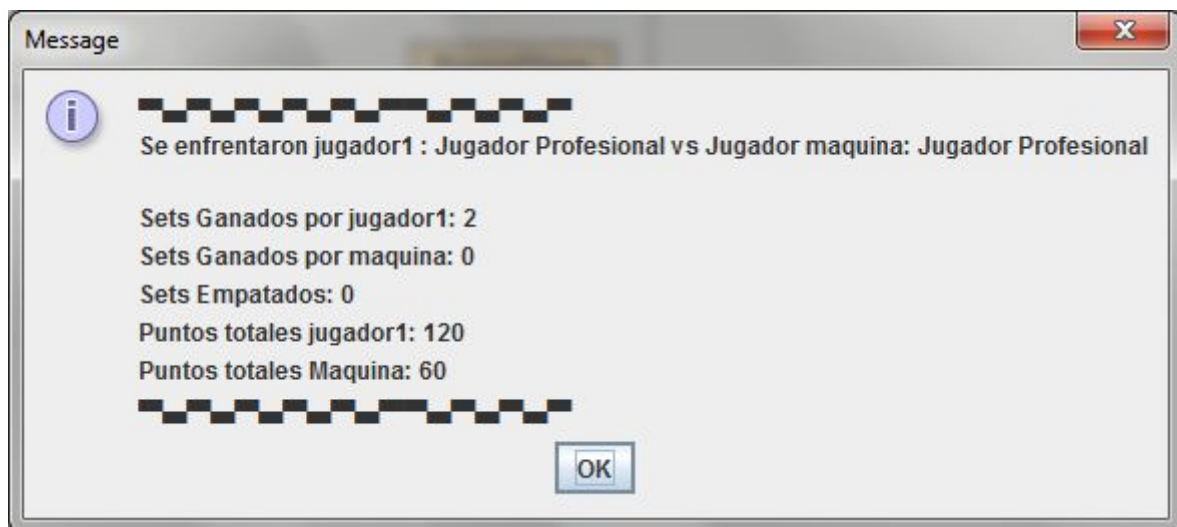
Si se digita 3 se pueden cambiar los números de sets por partida



Si se digita 2 se muestra en pantalla la mayor puntuación



Si se digita 1 inicia la partida y se tira una moneda para ver qué jugador empieza, al jugador usuario se le pide que digite las coordenadas a las que quiere tirar



Al final de la partida se muestran las estadísticas de la partida, en caso de muerte súbita se sigue jugando

6. Lo que no se logró

No se verificó en funcionamiento del .jar aunque el programa corre perfectamente desde el main