

CLASE: Competencia

Esta clase representa al evento en el que los competidores se ponen a prueba bajo la disciplina del lanzamiento de cáber. Sus atributos son una lista con los competidores inscriptos, y los podios que resultan al final del evento.

Métodos:

`_cargarCompetidores(String Path): void`

Este método recibe un path de un archivo. Se encarga de crear los objetos de Competidor y sus lanzamientos con la información que se encuentra en el archivo

`_escribirResultados(String Path): void`

Este método recibe un path de un archivo. Se encarga de escribir el archivo con el resultado de los podios que se dieron al final del evento.

`_calcularPodioConsistencia(): void`

Este método se encarga de calcular a los ganadores de la disciplina en consistencia, con un máximo de tres ganadores.

`_calcularPodioDistancia(): void`

Este método se encarga de calcular a los ganadores de la disciplina en distancia, con un máximo de tres ganadores.

CLASE: Competidor

Esta clase representa a los participantes del evento, los cuales compiten por los premios, y tienen un identificador y tres lanzamientos

Métodos:

`_calcularDistanciaTotal(): void`

Este método se encarga de sumar la distancia entre los lanzamientos válidos del competidor.

`_comprobarConsistencia(): void`

Este método se encarga de verificar que todos los lanzamientos del competidor sean válidos.

CLASE: Lanzamiento

Esta clase representa a los lanzamientos de cáber que realizan los participantes del evento. Tiene un largo y un ángulo

Métodos:

`_calcularDistanciaReal(): void`

Este método se encarga de calcular la distancia del lanzamiento, siendo esta el largo total o parcial del lanzamiento, dependiendo del ángulo.

`_esLanzamientoVálido(): boolean`

Este método se encarga de verificar que el ángulo del lanzamiento se encuentre entre los valores válidos que se tienen predefinidos.

CLASE: ManejoArchivos

Esta clase se encarga de realizar las acciones necesarias con los archivos, ya sea leer o escribir.

Métodos:

`_leerArchivo(String Path): void`

Este método se encarga de leer los datos que tiene el archivo que se pasa por parámetro.

`_escribirArchivo(String Path): void`

Este método se encarga de escribir los datos que sean necesarios en el archivo que se pasa por parámetro.