

EXAMEN FINAL

Leé con cuidado el enunciado y por lo menos dos veces para resolver lo pedido. Pensá bien la estrategia de resolución antes de comenzar el desarrollo de lo que te solicitan. El objetivo de este examen es **evaluar la correcta aplicación de los conceptos y técnicas** vistos hasta el momento:

- Correcta definición de clases y asignación adecuada de sus responsabilidades.
- Encapsulamiento, ocultamiento de información y uso de getters y setters sólo cuando corresponda.
- Modularización reutilizable y mantenible con uso de métodos con correcta parametrización.
- Correcta aplicación de miembros de instancia y de clase.
- Correcta aplicación de herencia y polimorfismo.
- Correcta aplicación conceptual de las relaciones entre clases.
- Correcta aplicación de TADs vistas en clase.

Se invalidarán automáticamente los exámenes:

- Donde no estén entregados en el zip los archivos uxf de UMLetino y nsplus de NSPlus.
- Donde los diagramas Nassi-Shneiderman no hayan sido generados desde el aula y en Modo Examen.
- Donde se detecte el uso de IAs para la resolución.
- Que utilicen estructuras de datos no explicadas en la materia.
- Que utilicen las estructuras de control no contempladas por las buenas prácticas enseñadas durante el cuatrimestre.
- Donde se detecte cualquier actividad grupal y/o no lícita en la resolución del ejercicio (serán sancionados todos los participantes).
- No se vea correctamente el rostro del alumno o alumna durante algún momento del examen, el mismo debe estar correctamente iluminado y enfocado durante todo el examen.

Enunciado

La IPSC (International Practical Shooting Confederation) nos encargó desarrollar un sistema para gestionar la puntuación y el ranking de los torneos de Tiro.

En principio se trabajará con un único torneo. En este torneo se utilizarán los tipos de blanco reconocidos por la Confederación.

Cada tipo de blanco tiene sus propias reglas de cálculo de puntos y penalización:

- **Cartón:** Tiene tres áreas de impacto: Alfa (5 puntos), Charlie (3 puntos) y Delta (1 punto). Cada disparo puede estar en cualquiera de estas áreas. La cantidad mínima de disparos por blanco es de 2. Si el blanco recibió dos o más disparos, el puntaje se calcula sumando los dos impactos de mayor valor; si tiene menos de 2, se restarán 10 puntos por cada disparo faltante. En esta etapa del proyecto y a modo de simulador, la clase contará con un método estático y privado llamado **obtenerRegionesImpactadas** que, dada una cantidad total de disparos, genera un array que indica en qué área (Alfa, Charlie o Delta) se produjo cada impacto. Este método no debe implementarse (sí usarse y declararse en el UML).
- **Metal:** Otorga 10 puntos si recibe al menos un disparo, resta 10 puntos si no recibe ninguno.
- **Non-Shoot (blanco trampa, penaliza):** Cuando aparece ese blanco el competidor debe dejarlo pasar sin dispararle. Siempre resta 20 puntos por cada impacto que recibe.

Durante la etapa de inscripción el torneo guarda todos los competidores que se registraron en él (al comenzar el desarrollo del torneo la inscripción ya está cerrada y nadie más se puede registrar). De cada uno guarda el número de inscripción (entero, secuencial, usado como ID del competidor), su nombre completo, el nombre del club al que pertenece y el total acumulado de puntos obtenidos (inicialmente en 0).

Cada torneo consta de distintas etapas que deben ejecutarse en estricto orden de creación. Cada etapa tiene, al momento de calcular los puntajes:

- La secuencia (fija) con todos los blancos de la etapa.
- La cantidad de disparos acumulados por cada competidor en cada uno de sus blancos.

EXAMEN FINAL

El sistema debe generar el puntaje final de cada competidor sumando los puntos obtenidos en cada blanco de cada etapa. De esta manera, el **resultado del torneo** para cada competidor será el total acumulado etapa por etapa, incluyendo las penalizaciones y bonificaciones aplicadas por cada tipo de blanco.

Se pide:

1. **Diseñar el diagrama UML** que modele el problema, incorporando cualquier aspecto necesario que se deduzca de este enunciado.
2. **Desarrollar los siguientes métodos**, asegurando que cualquier método de apoyo derivado también sea implementado según las reglas del torneo:
 - **calcularPuntaje** de cada tipo de **Blanco**: Calcula los puntos obtenidos a partir de la cantidad de disparos recibidos.
 - **calcularPuntaje** de **Etapas**: Calcula los puntos obtenidos en la etapa por un competidor a partir de su ID.
 - **calcularPuntaje** de **Torneo**: Calcula los puntos obtenidos por un competidor en todo el torneo a partir de su ID.
 - **obtenerResultados** de **Torneo**: Retorna una lista ordenada de participantes según su puntuación total, ordenados de mayor a menor puntaje.
 - los métodos de ordenamiento, en caso de haber Listas Ordenadas.