

# RECOMENDACIONES Y PAUTAS PARA EL TP

# ┌ TDA Heap ─┐

Será necesario implementar el TDA Heap para el TP2

Importante: no es necesario que sea completo.

- No se requieren las funciones de ordenar por heapsort, reorganizar arreglo, etc.
- Sólo se requiere que se puedan almacenar elementos y que al quitar la raíz siempre se quite el elemento mínimo.

# TDA Heap

¿Cómo sabemos si lo hicimos bien?

Confiamos que aprendieron la importancia de realizar pruebas de su código y que sabrán cómo implementar un TDA que requieren para un proyecto sin necesidad de las pruebas de la cátedra.

Recomendamos que se tomen un tiempo en implementar correctamente la estructura por separado. Así se ahorran dolores de cabeza innecesarios.

Además... no se preocupen mucho, es realmente cortito.



# El Gimnasio



Habrán entrenadores y un líder de gimnasio

Ustedes tendrán que decidir cuál es la mejor manera de organizar el gimnasio para que las batallas se realicen como se especificó en el enunciado.

Recuerden que una vez que un entrenador fue derrotado este ya no deberá ser enfrentado nuevamente.

Y que una vez que se derrotó al líder del gimnasio se habrá ganado la medalla.

# └ Los entrenadores ─

Los entrenadores serán aquellos a quienes debamos combatir antes de luchar contra el líder del gimnasio.

Están ahí para hacer presencia, suelen ser más débiles que el jugador y necesariamente más débiles que el líder del gimnasio.

Un entrenador tendrá hasta 6 Pokémon para combatir.



# └ Los entrenadores ─

Nuevamente quedará en ustedes saber cómo organizar la estructura del entrenador y sus Pokémon.

Recordar que cuando un entrenador te derrota, la próxima vez que te enfrentes utilizará los mismos Pokémon que en la primera batalla (a.k.a: no los pierde).

La estructura de datos que utilicen para los Pokémon debe ser flexible para satisfacer esta característica.



# └ Los Pokémon ─

Los Pokémon tendrán una serie de atributos:

- Velocidad
- Ataque
- Defensa

Y cada vez que ganen una batalla conseguirán una mejora a todas sus características.

Esta mejora tendrá un límite de 63 puntos.



# El personaje Principal



El jugador tendrá dos conjuntos de Pokémon y deben elegir cuál es la mejor manera de organizarlos

- Los Pokémon de combate
- Los Pokémon obtenidos

Recuerden que:

1. Los Pokémon de combate **deben** estar en el conjunto de Pokémon obtenidos.
2. Los Pokémon de combate siguen el orden que de como fueron seleccionados (No puedo comenzar con el cuarto Pokémon y pasar al segundo)



# El personaje Principal

Ejemplo de los conjuntos basado en el juego real

Se llama “party” al conjunto de Pokémon que combatirán y “caja” al conjunto de Pokémon obtenidos.



Party



Caja

# Algunas aclaraciones...

Los TDA's implementados pueden alterarse a gusto.

Pero... no se va a aceptar que se acoplen a los TDA's características del tp2, por ejemplo que las estructuras almacenen `gimnasio_t*` en vez de `void*` o salidas por pantalla dentro del TDA.

Se tendrán en cuenta las estructuras elegidas para almacenar los datos...

En un problema no es lo mismo utilizar una pila, una cola, una lista o un abb... deben utilizar cada estructura a conciencia y debe tener sentido. Será evaluado.

Deben además justificar estas decisiones en el readme.

¿Preguntas?