Modelar juego,Hay personajes y esmeraldas. Los personajes tienen poder y cuando compiten gana el que mas poder tenga.los personajes pueden obtener esmeraldas que esten disponibles pueden usar una esmeralda en su poder para duplicar su poder en forma permanente, y pueden competir con otros personajes para quitarles una esmeralda (solo se producen competencias cuando ambos personajes tienen esmeraldas; el personaje con mas poder se queda con la esmeralda).

El tipo Esmeralda es un entero que representa un identificador unico para cada esmeralda en el juego, y el TAD Personaje, tanto Esmeralda como Personaje son comparables por igualdad.

Interfaz de Personaje:
•Poder :: Personaje → Int
dado un personaje describe su poder (O(1))
•aumentarPoderDe :: Personaje → Int → Personaje
dado 1 personaje y un numero ,describe al personaje que resulta de incrementar el poder del mismo en el numero dado (O(1))

TAD EmGame donde P es la cantidad de personajes en el juego y E la cantidad de esmeraldas.

•IniciarJuego :: Set Personaje → Set Esmeralda → EmGame
dado un conjunto de personajes y un conjunto de esmeraldas (suponiendo ambos no vacios) describe un juego inicial con esas esmeraldas y esos personajes sin esmeraldas. Falla si alguno de los conjuntos esta vacio (O(P log P + E log E))

•elMasPoderoso :: EmGame → Personaje
dado un juego describe el mas poderoso (O(1))

•esmeraldasDe :: EmGame → Personaje → [Esmeralda]
dado 1 juego y un personaje,describe la lista de esmeraldas de ese personaje.
Falla si el personaje no esta en el juego (O(log P))

•obtenerEsmeralda :: EmGame → Personaje → Esmeralda → EmGame
dado un juego,personaje y esmeralda dentro de ese juego,describe el resultado de asignar la esmeralda al personaje suponiendono esta asignada. Falla si el personaje o la esmeralda no esta en el juego (O (log P + log
E))

•ganarEsmeralda :: EmGame → Personaje → Personaje → Esmeralda
dado 1 juego,2 personajes y 1 esmeralda dentro de ese juego,describe el resultado de realizar la competencia entre ambos personajes,suponiendo que esa esmeralda la tiene alguno de los personajes,y se la queda el mas poderoso. Falla si los personajes o la esmeralda no son parte del juego, y si la esmeralda no la tiene uno de los 2 personajes (O(log P + E))

•usarEsmeralda :: EmGame → Personaje → Esmeralda → EmGame
dado un juego,personaje,esmeralda (dentro de ese juego)describe el juego que resulta de la ultilizacion de su esmeralda por el personaje,suponiendo que el personaje tiene la esmeralda. Falla si el personaje o la esmeralda no son parte del juego, y si la esmeralda no la tiene el personaje (O(log P + E))

1) data Comando = IniciarJuego (Set Personaje) (Set Esmeralda)
| ObtenerEsmeralda Personaje Esmeralda
| CompetirPor Personaje Personaje Esmeralda
| UsarEsmeralda Personaje Esmeralda

partida :: [Comando] → Personaje
Dada una lista de comandos valida, describe el ganador del juego que resulta luego de ejecutar la lista. Falla si la lista de comandos no es valida, establecer eficiencia y justificar. Una lista de comandos es valida si su primer elemento es IniciarJuego y los diferentes comandos cumplen las precondiciones de los comandos que representan al momento de ser ejecutados.

REQUISITO: La solucion debe realizarse por recursion estructural, para lo cual dicha recursion debe hacerse en una funcion auxiliar,sobre el reverse de la lista dada.
SUGERENCIA: para fallar, procesar la lista de comandos y dejar que las operaciones del TAD fallen si no se cumplen sus precondiciones.

2) TAD EmGame, Escribir INV.REP.
Data EmGame = AG (Map Personaje [Esmeralda])
(Map Esmeralda (Maybe Personaje))
(MaxHeap Personaje)

El primer Map tiene todos los personajes del juego con las esmeraldas de cada uno, el segundo,todas las esmeraldas del juego y si son poseídas, quien las posee y el tercero todos los personajes del juego ordenados por mayor poder.

3) Implementar funciones respetando propósito, y que las fallas que tenga un mensaje de error, cumpliendo con eficiencia y justificando.