

Introducción

buenas, en esta presentación explicaremos brevemente de que consiste el juego.
como sabemos la Conquista, trata de posicionar varios jugadores dentro de un tablero y realizar enfrentamientos entre cada uno de ellos, donde gana el que logre conquistar todas las casillas.

el desafío planteado por la catedra, era realizar como se solucionaba el duelo.

Nosotros elegimos el método de la ruleta rusa, para aquellos que no conoces, consta de una escopeta, donde se le cargara 6 balas de manera aleatoria y solo una tendrá la pólvora, el resto serán cartuchos vacíos. Esta arma cargada se le dará al jugador, seleccionado aleatoriamente, y el tendrá dos opciones, dispararse o disparar al contrincante.
sí elige dispararse y sale la bala - pierde
sí por el contrario no sale la bala - mantiene su turno y elige de nuevo
ahora, si elige disparar al contrincante y sale la bala gana el turno,
sí por el contrario no sale la bala, pierde el turno.
cada uno contara con 3 vidas y el duelo gana aquel que pierda las 3 vidas.

para implementar este juego, hemos diseñado tres clases principales, (Jugador, Tabla, Juego), cada una encargada de una parte específica de la lógica del juego.

Donde Jugador -> tendrá los atributos de nombre, sigue y posiciones.

Tabla -> tendrá los atributos de celdas y tamaño

Juego -> tendrá toda la lógica para iniciar y finalizar el juego

ahora vamos a nombrar algunos métodos importantes de cada clase,

que hacen al juego.

dentro de Tabla vamos a encontrar:

asignarJugador -> este método recibe un jugador y en base a sus posiciones lo coloca dentro de la tabla. Si las posiciones están vacías, significa que el jugador es la primera vez que juega y lo posiciona de manera contigua, pero si las posiciones no están vacías, significa que el jugador gana un duelo y actualiza en esa posición el jugador dentro de la tabla y por último actualiza las posiciones del jugador.

conquistarCeldas -> este método recibe dos jugadores, el ganador y perdedor, lo único que hace es pasar las posiciones del perdedor al ganador, eliminar las posiciones que controla el perdedor y actualizar la variable sigue a false para saber que el perdedor no está más en juego, y por último actualiza la tabla en esa posición con el jugador.

hayGanador -> este método recupera el primer jugador dentro de la tabla y recorre la misma, comparando los nombres, si son distintos quiere decir que hay más de un jugador dentro del tablero y como resultado devuelve false, pero si son iguales, devuelve un mensaje de ganador y true.

obtenerCeldaConquistar -> este método recibe un jugador y devuelve los jugadores adyacentes, recorre las posiciones del jugador y con las variables top, right, bottom y left, controla los jugadores que se encuentran en esas posiciones, si la posición es mayor que 0 y menor igual al final y no se encuentra entre las posiciones del jugador, los guarda con un formato (jugador, posición) y devuelve un array de eso.

luego tenemos la clase jugador

donde solo vamos a tener un constructor, que le da valores iniciales a las variables.

los setter para darles valores manuales a las variables

getter para mostrar la información

un método addPosiciones, para asignar posiciones contiguas y no borrar todas

y un método infoJugador que muestra de manera más amigable la información del jugador.