

SIMULADOR DE CASINO

El problema a resolver es simular una partida de ruleta en un casino con varios jugadores y un croupier. Los jugadores hacen apuestas y el croupier gira la ruleta y proporciona el resultado. Los jugadores pueden tener diferentes estrategias de apuesta y pueden seguir apostando hasta que pierden todo su dinero, o la banca se quede sin dinero también.

La solución estructural se basa en la creación de hilos para cada jugador y un hilo para el croupier. Los jugadores hacen apuestas al mismo tiempo y el croupier gira la ruleta y proporciona el resultado a la clase Banca, que lleva el registro del dinero total del casino. El croupier también despierta a los jugadores que esperan con el objeto Lock para que puedan hacer sus apuestas.

El programa se divide en 4 clases:

1-Clase Principal:

Esta clase es la encargada de iniciar el programa y crear los hilos para cada jugador y el croupier. Contiene el método main donde se llaman a los demás métodos necesarios para iniciar la partida de ruleta. También contiene los métodos crearJugadores y crearCroupier, que se encargan de crear y iniciar los hilos correspondientes. El método Menu se utiliza para obtener la estrategia de apuesta de los jugadores y el método EstrategiaRandom se utiliza para seleccionar una estrategia aleatoria para ciertos jugadores.

2-Clase Jugador:

Esta clase representa a cada uno de los jugadores y se encarga de realizar las apuestas. Contiene un constructor que recibe el objeto Lock y la estrategia de apuesta del jugador. También contiene el método run, que se encarga de hacer las apuestas del jugador hasta que pierda todo su dinero o hasta que la banca no pueda repartir más dinero. Utiliza el método sleep para simular una espera entre apuestas y el método synchronized para sincronizar el acceso a los hilos y evitar problemas de concurrencia.

3-Clase Croupier:

Esta clase representa al croupier y se encarga de girar la ruleta y proporcionar el resultado a la clase Banca. Contiene un constructor que recibe el objeto Lock y el método run, que se encarga de girar la ruleta y despertar a los jugadores que esperan con el objeto Lock e iniciar una nueva ronda de apuestas. Utiliza el método sleep para simular un tiempo de espera entre giros de la ruleta y el método synchronized para sincronizar el acceso a los hilos y evitar problemas de concurrencia despertándolos todos a la vez.

4-Clase Banca:

Esta clase se encarga de llevar el registro del dinero total del casino y proporcionar el resultado de cada giro de la ruleta a los jugadores. Contiene los métodos getDineroTotal y setDineroTotal para obtener y modificar el dinero total, y el método setReiniciarBanca para reiniciar el dinero total al inicio de la partida. También contiene los métodos getResultado y setResultado para obtener y modificar el resultado de cada giro de la ruleta. Utiliza el método synchronized para sincronizar el acceso a los hilos y evitar problemas de concurrencia.