

Parcial II(Informática II)

Davin José Chavarria Valencia

Universidad de Antioquia

26/10/2023

Contextualización breve del problema.

En un mundo cada vez más influenciado por la tecnología, los juegos de mesa clásicos se enfrentan al desafío de mantener su atractivo entre las nuevas generaciones. Este proyecto se enfoca en la tarea de transformar el juego estratégico de Othello en una versión digital utilizando el lenguaje de programación C++ y el paradigma de Programación Orientada a Objetos. El propósito es crear una adaptación electrónica de Othello que mantenga la integridad de las reglas y la diversión del juego original, permitiendo que sea disfrutado en plataformas en línea y dispositivos móviles. El principal reto reside en desarrollar la lógica subyacente del juego, diseñar una interfaz de usuario (si es necesaria) y proporcionar un código bien estructurado y documentado.

Análisis del problema:

1. Contexto:

- a. El contexto del problema se encuentra en el ámbito de la adaptación de juegos de mesa tradicionales al entorno digital. El juego de Othello, un juego de estrategia para dos jugadores se ha seleccionado como el foco de este proyecto. La adaptación es necesaria para mantener la relevancia y la accesibilidad del juego a las nuevas generaciones que están más familiarizadas con las plataformas digitales.

2. Requisitos Clave:

- a. Implementación de las reglas de Othello: Es fundamental comprender y codificar las reglas del juego Othello, que incluyen la colocación de fichas, la captura y volteo de fichas del oponente, y la determinación de cuándo termina el juego.
- b. Programación Orientada a Objetos: Se debe aplicar el enfoque de Programación Orientada a Objetos (POO) para organizar el código en clases y objetos que representen el tablero, las fichas, los jugadores y las reglas del juego.
- c. Interfaz de Usuario (Opcional): Si se requiere, se debe diseñar una interfaz de usuario que permita a los jugadores interactuar con el juego de manera intuitiva. Se hace desde consola

- d. Documentación: Es necesario proporcionar documentación clara y comentarios en el código para facilitar la comprensión y el mantenimiento del proyecto.

Desafíos y Consideraciones:

1. Diseño del Tablero: La representación del estado del tablero y las fichas es un aspecto crítico del proyecto. Se debe determinar cómo almacenar y actualizar esta información de manera eficiente.
2. Lógica del Juego: La implementación de las reglas del juego, incluida la identificación de movimientos válidos y la gestión de la puntuación, es un desafío técnico importante.
3. Pruebas: La validación del funcionamiento correcto del juego es esencial. Se deben realizar pruebas exhaustivas para garantizar que las reglas se cumplan y que no haya errores.
4. Entregables Esperados:
 - a. Código bien estructurado en C++ que implementa el juego Othello utilizando POO.
 - b. Informe detallado que describa el contexto del proyecto, las reglas del juego, el diseño de las clases y la implementación del código.
 - c. Video demostrativo que muestre el funcionamiento del juego y las características clave.
5. Conclusiones:
 - a. La implementación de Othello en C++ es un proyecto que combina desafíos técnicos y creativos. Requiere una comprensión profunda de las reglas del juego, una planificación cuidadosa de la estructura de clases y una posible interfaz de usuario. La documentación y las pruebas rigurosas son esenciales para garantizar que el juego funcione correctamente. La adaptación exitosa de juegos tradicionales al mundo digital permite preservar su atractivo para nuevas generaciones y proporciona una oportunidad emocionante para aplicar conceptos de programación orientada a objetos.

Parte de la solución:

1. Saber de qué se trata el juego: es un juego para dos jugadores en un tablero de 8x8

Clases para implementar:

1. Tablero: donde se llevará a cabo el juego.
2. Fichas: Irán dentro del tablero y podrán cambiar de estado (Blanco o negro).

Solo hay dos clases porque se trató de simplificar los objetos a usar en dicho juego y lo más relevante es el tablero y las fichas, ya que estos van a estar en constante cambio y debido a que el jugador depende del color de las fichas.

Se definen los atributos de las clases y sus respectivos métodos, para luego proceder a crear una interfaz para el usuario vía consola.

Se podría hacer un menú que le proporcione tres opciones según sea el caso

1. Iniciar una nueva partida: como el nombre lo indica este iniciaría una nueva partida.
 2. Ver juegos anteriores: permite ver las personas que jugaron con anterioridad, fecha, hora, ganador y con cuantas fichas gano
 3. Cerrar el programa: se finaliza la ejecución del programa.
-
1. Para la opción de iniciar partida se podrían crear opciones que permitan ver el tablero cada vez que se haga una jugada, saber cuando finalizar la partida, saber quien es el ganador, con cuantas fichas gano, intercalar entre jugadores para que estos posicionen sus fichas (verificar si se pueden colocar fichas en la posición que quiere el usuario), guardar los datos de la partida.
 2. Para esta opción se crearon funciones que permitieron leer el archivo donde se almacenaba la información y poder imprimirlo en pantalla.
 3. Se cierra el ciclo.