

Informe

Se dividió el programa en la mayor cantidad posible de TDAs. Al iniciar se preguntan las dimensiones del tablero, la dificultad de la partida y el numero de jugadores. Con esta información, se programa un tablero, con sus respectivas casillas organizadas en forma matricial y se plantaran minas en él. Con las minas dispuestas, se calculan los puntos de las casillas circundantes. También se utilizo la librería BMP para poder imprimir un tablero mas personalizable. Se dispondrá también a los jugadores en una lista y se iterara a traves de ella con el correr de los turnos (mientras quede alguno vivo). Se implemento para cada jugador una planilla para llevar sus puntos.

En cuanto a la ampliación correspondiente al trabajo practico N° 3, se implemento una estructura semejante a un árbol binario, que almacenara (en forma del tipo de dato Jugada), el estado de la partida en cada turno y se indicara constantemente la jugada actual. Se podra iterar a traves de dicha estructura verticalmente – a través de los turnos – y horizontalmente – ya que si se deshace una jugada, se creara una nueva linea de tiempo donde jugar los proximos turnos-.

Archivos:

- Casilla: Funcionalidades de la unidad fundamental del juego: las casillas del tablero.
- Tablero: Quien organizara a las casillas en forma matricial.
- Terrorista: Encargado de plantar las bombas en el tablero.
- Jugador: Clase que abarca las funcionalidades disponibles para quien juega.
- Planilla: Encargada de los puntos de los jugadores.
- Referee: Organizara a los jugadores.
- Lista: Estructura de datos que se usara para el almacenamiento de jugadores y la rotacion de sus turnos.
- Graficador: Quien traducira el desarrollo del juego en un tablero en BMP.
- EasyBMP: Funcionalidades relacionadas a las imagenes BMP que seran usadas en el juego.
- LineasDeTiempo: Estructura donde se almacenaran los turnos y las “bifuraciones temporales” que ocurran con el deshacer y rehacer de los mismos.
- Jugada: TDA que se almacenara dentro de las LineasDeTiempo. Contiene los datos del estado del juego turno a turno.

Manual de Usuario

Al iniciar, se preguntará la cantidad de filas y columnas del tablero para jugar. Además también será necesario establecer la cantidad de jugadores y la dificultad de la partida.

El juego consiste en despejar todas las casillas de una pantalla que no oculten una mina.

Si un jugador descubre una casilla con una mina queda afuera del juego.

Algunas casillas tienen un número, el cual indica la cantidad de minas que hay en las casillas circundantes. Así, si una casilla tiene el número 3, significa que de las ocho casillas que hay alrededor (si no es en una esquina o borde) hay 3 con minas y 5 sin minas. Si se descubre una casilla sin número indica que ninguna de las casillas vecinas tiene mina y éstas se descubren automáticamente.

El jugador de turno también tiene la opción de sugerir que una casilla es una mina.

El jugador hará su jugada y luego pasará su turno.

El juego tiene un sistema de puntos, donde un jugador suma un punto si supone la existencia de una mina y es correcto, o resta un punto si es incorrecta. Si otro jugador quiere corregir esta suposición y tiene razón, suma 2 puntos y si es equivocada resta 2 puntos. El jugador con más puntos al terminar el juego gana (de los que no están eliminados).

Hay 3 dificultades posibles: fácil, medio, difícil; que determinaran la cantidad de bombas que se planten en el tablero, y los puntos ganados.

Manual del Programador

Sistemas Operativos donde esta probado el correcto funcionamiento:

- Linux Ubuntu – Linux Lubuntu – Windows 10

Entornos de Desarrollo Integrado donde esta probado el correcto funcionamiento:

- Eclipse C/C++ - Codeblocks

Cada función se encuentra comentada en el archivo header (.h) correspondiente.

Cuestionario

1) Este trabajo práctico se programó y testeó en los siguientes sistemas operativos y entornos de desarrollo:

- * Ubuntu - Eclipse
- * Ubuntu - Eclipse
- * Windows - Codeblocks

2) Para sincronizar el código se utilizó la plataforma de desarrollo cooperativo de software Github, donde, previamente divididas las tareas, cada integrante ponía en común su código. Además se utilizó un grupo de WhatsApp para una mejor comunicación y seguimiento del trabajo.

3) Aspectos Positivos de EasyBMP:

- Otorga al usuario una gran libertad, se pueden lograr variadas cosas con su uso.
- No requiere instalación y es una librería liviana. Los programas resultantes no serán pesados.
- Presenta una gran cantidad de funciones y el código es fácil de compilar.

Aspectos Negativos de EasyBMP:

- Es muy complejo y comprenderlo al principio es difícil.
- Se trabaja pixel por pixel, lo cual hace que pintar algo con cierto grado de complejidad sea un trabajo considerable.
- Está únicamente en idioma inglés, tanto en documentación como en el código, y podría presentar alguna dificultad.