

Trabajo Práctico 1: BATALLA DIGITAL V1.0

75.41 - Algoritmos y Programación II

Cuestionario:

- 1- ¿Qué es un debug?
- 2- ¿Qué es un “breakpoint”?
- 3- ¿Qué es “Step into”, “Step over” y “Step out”?

Respuesta 1:

El término debug, debugging o depuración viene de la palabra bug en inglés que se traduce como un error en el código que está haciendo que este funcione de forma adecuada. De ahí, el término debugar es la acción de eliminar estos mismos errores en el código para que este funcione como corresponde. Las páginas de desarrollo web, cuentan con sus propios debuggers, son fundamentales para códigos que cuenten con miles y miles de líneas de código y ahorran una enorme cantidad de tiempo a que un programador tenga que buscar manualmente cada error en el código. Otra herramienta usada para debugar son los logs de errores, que son listas que dicen que fue lo que pasó al momento de la ejecución de un código, línea por línea, lo que ayuda mucho a saber donde empezar para debugar un programa. En conclusión los debuggers son herramientas fundamentales para los programadores que trabajan en grandes proyectos, estas herramientas ahorran el tiempo de búsqueda de errores y agilizan el tiempo de codeo.

Respuesta 2:

Los breakpoints o puntos de interrupción son puntos de un código o programa donde se decide detener la ejecución por algún motivo tal como investigar el valor de las variables o ver que ocurre en ese momento en la ejecución del código.

Respuesta 3:

Step into, step over y step out son funciones normales de un debug, cuyo objetivo es la ejecución paso a paso de un código o programa.

-*Step into*: esta función ejecuta la siguiente línea de código de donde se encuentre el programa. Si esta línea es una llamada a función, ingresa en esta.

-*Step over*: esta función ejecuta la siguiente línea de código, pero, si esta es una llamada a función, la omite y no ingresa en ella.

-*Step out*: esta función , si me encuentro dentro de una función, sale de ella.

Manual de usuario

Batalla digital es un juego del estilo juego de mesa, se juega de a 2 jugadores, cada uno contiene 4 soldados y se ubican en lados opuestos de un tablero de 20 filas y 20 columnas de casilleros. En estos casilleros va a ser donde los soldados están parados. Los soldados del jugador 1 se representan en el tablero con un "1" y los soldados del jugador 2 respectivamente con un "2".

El juego es por turnos, en cada turno, el jugador coloca una mina y un movimiento de un soldado. En cada turno el tablero se actualiza y se crea un archivo "tablero.txt" donde se guarda el estado actual del tablero.

Colocar minas:

En cada turno un jugador coloca una mina en el tablero (en la posición que desee), si la mina que coloca coincide con la posición de un jugador, la mina mata al jugador y la casilla pasa a estar inactiva. Luego de colocar la mina, si un jugador de cualquier equipo pisa alguna de ellas, muere automáticamente y la casilla queda inactiva.

Movimiento:

Luego de colocar la mina, el usuario puede elegir uno de sus soldados en el tablero y moverlo en 8 direcciones: Arriba, abajo, izquierda, derecha, diagonal arriba derecha, diagonal abajo derecha, diagonal abajo izquierda y diagonal arriba izquierda. Si al moverse, en el casillero hacia donde se movió el soldado, hay un soldado del jugador contrario, ambos soldados son eliminados, y el casillero queda inactivo, al igual que si pisa una mina.

¿Qué es un casillero inactivo?:

Un casillero inhabilitado queda inaccesible para que un soldado se mueva hacia él, en caso de que un soldado intente moverse a un casillero inactivo, este jugador perderá su turno de moverse. El casillero inactivo se representa en el tablero con una "X".

¿Cuándo termina el juego?:

El ganador del juego será el que elimine los 4 soldados del jugador contrario, mediante la colocación de minas. En caso de que ambos se queden sin soldados en el mismo turno, se producirá un empate.

MANUAL DEL PROGRAMADOR

El archivo principal funciones.cpp, consta de una estructura simple, basándose casi completamente en las funciones, clases y constantes de la biblioteca funciones.h.

Clases:

Las 2 clases que se utilizan son Soldado y Mina.

La clase Soldado es una abstracción que modela las características y comportamientos de un soldado en el juego batalla digital. Un objeto de la clase Soldado tiene los siguientes atributos:

Posición: las coordenadas x e y que indican su ubicación en el tablero.

Equipo: un carácter que representa a qué equipo pertenece el soldado.

ID: un número entero que identifica de manera única al soldado.

Estado de vida: un valor booleano que indica si el soldado está vivo o muerto.

Además, la clase Soldado define los siguientes métodos:

Constructor: inicializa los atributos del soldado con los valores proporcionados.

Métodos de establecimiento: permiten cambiar el valor de la posición, el equipo y el ID del soldado.

Método de movimiento: mueve al soldado a una nueva posición en el tablero, siempre y cuando sea un movimiento válido.

Métodos de obtención: devuelven los valores de la posición, el equipo y el ID del soldado.

Métodos de control de vida: permiten matar al soldado o verificar si está vivo o muerto.

En resumen, la clase Soldado modela las propiedades y comportamientos de un soldado en el juego, permitiendo moverlo en el tablero, cambiar su equipo y estado de vida, y obtener información sobre él.

La clase Mina representa una mina en el juego. La clase tiene un constructor que inicializa la mina como activa.

El método "posicionar Mina" se utiliza para posicionar la mina en el tablero y verificar si hay soldados en la casilla seleccionada. Si la casilla está ocupada por un soldado del equipo contrario, la mina explota y mata al soldado, devolviendo el equipo del soldado que fue asesinado. Si la casilla está ocupada por un soldado del mismo equipo, la mina no se activa. Si la casilla está vacía, se coloca la mina en la casilla.

Los métodos "activar" y "desactivar" se utilizan para cambiar el estado de la mina entre activa e inactiva, respectivamente. El método "estaActiva" devuelve el estado de la mina (activa o inactiva). Los métodos "posicionX" y "posicionY" devuelven la posición de la mina en el tablero.

El método "equipoMina" devuelve el equipo de la mina, que se establece cuando se coloca la mina en el tablero.

Funciones:

El código es un juego en el que dos jugadores compiten por colocar minas en un tablero y mover a sus soldados para evitarlas. El juego comienza con cuatro soldados para cada jugador. Cada jugador tiene la oportunidad de colocar una mina en cada ronda y luego decidir si mover o no a uno de sus soldados restantes.

La función principal del juego, llamada juegoPrincipal, toma como argumentos un puntero doble a un tablero (representado como una matriz de caracteres de 20x20 que será representado en la consola), un puntero al primer jugador y un puntero al segundo jugador (cada uno representado por un arreglo de objetos de tipo Soldado). La función crea un arreglo dinámico de objetos Mina y realiza un bucle mientras ambos jugadores tengan soldados con vida.

En cada ronda, el jugador actual es determinado por si el número de ronda es par o impar. El jugador actual es entonces solicitado a colocar una mina en una casilla del tablero mediante el método de dicha clase llamado posicionarMina, con la posibilidad de matar a un soldado enemigo si la mina es colocada en su casilla. Después de colocar la mina, el tablero es mostrado en la consola mediante la función mostrarTablero y exportado en un archivo txt a través de la función exportarTablero. El jugador puede elegir mover uno de sus soldados restantes, lo que le permite desplazarse a una casilla en el tablero. Si el soldado se mueve a una casilla con una mina, muere y se elimina del juego. Si el soldado se mueve a una casilla con un soldado enemigo, ambos soldados mueren y se eliminan del juego.

El proceso de colocación de minas, movimiento de soldados y actualización del tablero se repite hasta que un jugador se queda sin soldados vivos.