



CE-1103 Algoritmos y Estructuras de Datos I
Área Académica de Ingeniería en Computadores

Primer Proyecto Busca Minas

Realizado por:

Gabriel Bolaños Barboza - 2022327511

Profesor:

Luis Diego Noguera Mena

I Semestre, 2023

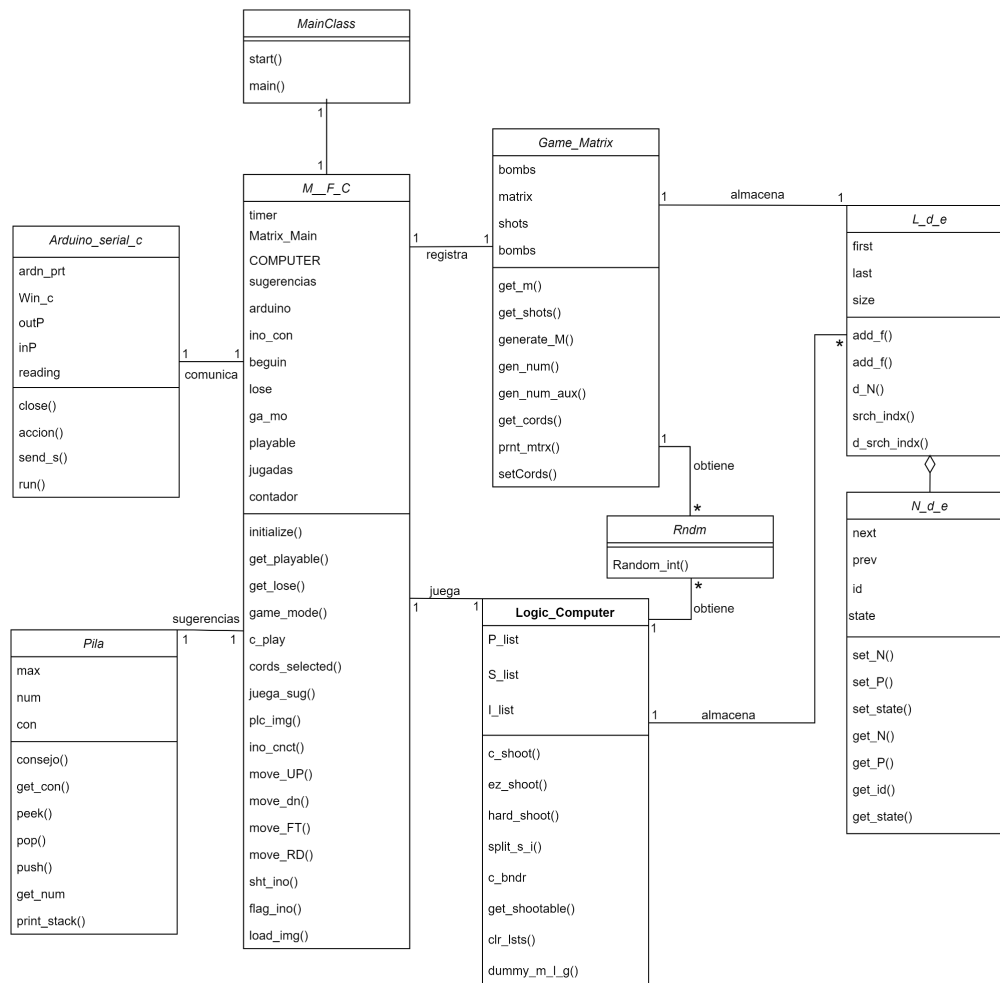
Índice:

Descripción del problema:	3
diagrama de clases:	4
Estructuras de datos empleadas:	4
Lista enlazada:	4
Pila:	5
Matriz:	5
Arreglo:	6
Links importantes:	6

Descripción del problema:

El proyecto consiste en la implementación de un juego llamado Minesweeper, donde el jugador debe elegir celdas en una matriz 8x8 que no contengan minas, según indicadores que describen la cantidad de minas que hay alrededor de una celda específica. El juego debe permitir que el usuario marque los lugares donde ha identificado que hay minas, mostrar la cantidad de minas encontradas y el tiempo transcurrido desde que inició el juego. Además, debe permitir que el usuario juegue contra el ordenador en dos niveles: Dummy Level y Advanced Level, donde el ordenador selecciona las celdas de manera aleatoria o utilizando una lista enlazada de las celdas disponibles. El juego también debe contar con una pila de sugerencias que el sistema agrega después de cada cinco jugadas del usuario. El control del juego se realiza mediante un dispositivo que debe permitir moverse en el tablero de juego, emitir un sonido cuando se seleccione una celda que no tenga mina y un sonido diferente cuando se seleccione una celda que sí tenga mina, y encender un led al dar click derecho sobre una celda para avisar al usuario que ha asignado una celda como posible mina. La implementación debe hacer uso del paradigma orientado a objetos mediante la utilización de estructuras de datos lineales y la programación orientada a objetos en Java.

diagrama de clases:



Estructuras de datos empleadas

Para el desarrollo del proyecto se emplearon estructuras de datos con la finalidad de proporcionar una mejor solución de manera que se adaptara de la manera más adecuada a lo que se necesita. Las estructuras de datos empleadas son:

Lista enlazada:

Para una mayor practicidad la lista empleada es doblemente enlazada y a la vez double ended. Lo que permite eficientar algunos procesos y de esta manera no tener que recorrer la lista más de lo necesario.

La lista en sí está compuesta por dos clases. Una dispuesta para administrar toda la lógica necesaria para añadir y eliminar elementos de la misma entre otros procesos. Por otro lado, la segunda clase necesitada para esta estructura de datos es la clase destinada para el modelado de nodos. por lo que esta clase solo tiene en sí misma no controla ninguna lógica.

En conjunción estas dos clases logran guardar datos a conveniencia y brindan las facilidades que añaden las estructuras de datos modulares. En el caso específico de este proyecto algunos de sus usos más notables es a la hora del turno de la computadora. Esto en ambos modos de juego ya que en el dummy se genera una lista con todas las casillas del tablero y después se seleccione una posición aleatoria de la lista. En el nivel avanzado se utilizan las listas para obtener las casillas que pueden ser jugables. después se dividen en las que son seguras de jugar con bandera o sin mina y otra en las que no se esta seguro que puede haber.

Pila:

La pila es una estructura de datos realmente sencilla. ya que esta almacena datos conforme le llegan y bajo un límite. Esta solo puede acceder al último elemento que recibió por lo que se dice que trabaja los datos de manera LIFO.

En el desarrollo de este proyecto se utiliza una pila para guardar las sugerencias del juego hacia el jugador. Ya que cada cierta cantidad de jugadas se le ofrece una pista y cuando el jugador acepte la sugerencia se le debe brindar las más recientes

Matriz:

Una matriz en general refiere a una manera de acomodar datos en dos dimensiones, es decir una matriz guarda unas proporciones y puede almacenar datos con base en coordenadas. En el desarrollo del proyecto se utilizaron distintas matrices. La principal manera en la se utilizó una matriz es en el tablero. Se utilizó una clase llamada 'Game_Matrix' en esta se guardan los valores de cada casilla y los actualiza a necesidad del programa. como cuando se realiza un tiro y se actualiza el valor de la casilla para que se sepa que ya fue jugada. Otra matriz que se puede encontrar en el programa es una matriz con los botones para un acceso indexado.

Las matrices utilizadas en el desarrollo de este proyecto son con base en arreglos ya que al no ser necesario un tamaño dinámico es más efectivo usar arreglos por el acceso indexado.

Arreglo:

Un arreglo es una estructura de datos no dinámica la cual tiene la ventaja de que por su estructura en memoria tiene acceso indexado eficiente. Esto permite que tenga muchos posibles usos. En este proyecto se implementó de muchas maneras. La manera más notable es que las matrices que se implementaron son matrices hechas con arreglos. Otro uso auxiliar de los arreglos se evidencia en la pila ya que esta funciona con un arreglo principal.

En el desarrollo del proyecto no solo se usaron arreglos de manera auxiliar sino que también se usaron por si mismo un ejemplo de esto se evidencia en el controlador del arduino ya que este utiliza un arreglo para obtener los puertos disponibles.

Links importantes:

github (el repositorio es publico)

https://github.com/GABOBB/PI_DI_I_2023

azure devops

https://dev.azure.com/gabolanos/Busca%20Minas%20PI_DI