



**Universidad Tecnológica Centroamericana**

**UNITEC**

Laboratorio de Programación III

Ana Gómez

**Laboratorio IV**



## Objetivos

- Desarrollar soluciones recursivas a problemas iterativos.
- Practicar el uso de matrices en clases.
- Realizar clases simples.

## Desarrollo de la práctica

Para el desarrollo satisfactorio de este laboratorio, siga las instrucciones que a continuación se le presentan.

### Ejercicio

El día de hoy, crearán lo siguiente:

#### Minesweeper

El Minesweeper es un juego clásico que hace uso de las matrices. Internamente, se maneja un tablero  $m \times n$ , pero para propósitos del laboratorio, se manejarán tableros de  $n \times n$ , con  $n$  siendo un valor **mínimo** de 10. Este tablero podría verse como una matriz numérica (int). Se asignan aleatoriamente en posiciones en el tablero una cantidad  $k$  (donde  $k$  sea menor o igual a  $n$ ) de minas.

El usuario verá un tablero vacío al momento de iniciar el juego (puede ser un tablero de chars). Se le pedirá al usuario que ingrese coordenadas (fila, columna) para jugar.

Si al momento de jugar cae en una mina, es Game Over.

Si al momento de jugar cae en un contador, y alrededor del mismo hay ceros, se deben de mostrar los ceros (y otros contadores) adyacentes de manera recursiva. El recursivo termina cuando se muestra un valor distinto de 0.

*Adicionalmente*, se debe mantener un puntaje para cada partida del juego. El puntaje será igual a la sumatoria del valor revelado \* 100.

#### Administración

La administración del juego consiste en lo siguiente:

Para crear una nueva partida, deben de existir jugadores. Del jugador se conoce su nombre, su usuario, una clave y su puntuación más alta de todos los tiempos.

Si al crear una nueva partida no existe un jugador, entonces se debe dar al usuario la opción de crear uno.

Todos los jugadores se deben almacenar en un vector de tipo jugador. Se deben poder agregar, eliminar y modificar los jugadores del vector.

No es necesario almacenar todas las partidas creadas, pero si es necesario mantener un historial de las 3 puntuaciones más altas hechas por algún jugador.

Finalmente, se debe poder hacer una impresión de dicho historial.

## Ponderación

Ponderación	
<b>Minesweeper</b>	<b>6pts</b>
Método de mostrar los 0s adyacentes de manera recursiva	4pts (si lo hacen iterativo les doy 1/4)
Asignación de minas utilizando random	1pt
Impresión nítida del tablero (no debo de poder ver los números PLS)	1pt
<b>Gameplay, creación de partidas, players (administración y ejecución)</b>	<b>4pts</b>
Agregar, modificar, eliminar	1pt
Impresión de puntuaciones	1pt
Utilizan ncurses (bonito)	2pt

Si su lab está 100% funcional (ya sea recursivo o no el método de los 0s), y hacen que el minesweeper sea de tamaño mxn (ingresado por el usuario), les doy un punto extra.

Penalizaciones por:

1. Quiebres en el programa como floating point exception, segmentation fault, what() bad alloc, entre otros.
2. Programa no amigable.
3. Código mal tabulado.
4. Variables no significativas