**Trabajo Practico Programación III -**

**“FueraLuces” - Acosta Manuel, Cardozo Nestor**

Implementación del código:

Para este proyecto decidimos hacer un diagrama de 3 clases para dividir las responsabilidades correspondientes.

En primer lugar, hemos creado la clase "Tablero", la misma representa una matriz de booleanos, que recibe una medida para definir su tamaño. Ésta cuenta con el método CambiarEstado(int fila, int col) que a través de los parámetros recibidos chequea, primero, que la posición recibida este dentro de las medidas del tablero, y luego cambia de estado al booleano en cuestión.

En segundo lugar, creamos la clase "Juego" la cual se va a encargar de toda la inteligencia del juego, utilizando al TAD, antes mencionado, "Tablero" como atributo.

Ésta clase cuenta con 3 métodos esenciales para su funcionamiento:

* public void mezclarLuces(int nivel)

Este método da la impresión de mezclar las luces de forma aleatoria, para esto guarda en dos variables auxiliares un numero random para cada una de éstas, que serán pasadas como parámetros al método EncenderLuz(int fila, int col). Esto se realiza las veces que el entero "nivel"recibido como parámetro lo indique. Ya que serán la cantidad de "cruces" de luces encendidas que podrán verse de acuerdo al nivel elegido.

* public void EncenderLuz (int fila, int col)

Este método cambia el estado de la luz en la posición indicada por el parámetro recibido en fila y col a través del método CambiarEstado(int fila, int col) del objeto Tablero, en este caso lo llamamos 5 veces, ya que las reglas del juego indican que al apagar una luz también deben ser afectadas las luces vecinas, es decir las que se encuentran por arriba, abajo y sus laterales.

* Public boolean VerificarGanador()

Este método recorre el tablero verificando que todas las posiciones sean falsas, es decir que las luces están apagadas, si éste es el caso, retorna true y lo usamos como parámetro para terminar el juego.

Luego encontramos el método public int contarToques(int turno) que suma un valor a la variable entera recibida como parámetro, no lo consideramos esencial para el juego ya que la idea era usarlo para contar toques y guardarlos en los records con el nombre del jugador.

Por ultimo, se crea con Windows Builder la clase “Interface”, donde hemos personalizado un espacio grafico donde se ven datos como el nombre del juego, nombre del jugador, turnos, niveles para elegir, créditos y un tablero.

Para poder graficar la información creamos un arreglo de botones que va a representar al tablero y un objeto Juego, que va a estar modificando los estados del tablero original de la siguiente manera.

Private void CrearBotones (int fila, int columna)

Este método instancia el arreglo de botones antes creado, con un tamaño de fila multiplicado por columna y va creando un botón a medida que se va generando dentro de su respectivo panel.

Private void MostrarTablero()

En este método contamos la cantidad de posiciones del tablero y cargamos la imagen encendida si la posición es true y apagada si la posición es false. También se encuentran métodos extra que generan un efecto de apagado y encendido.

public void actionPerformed (ActionEvent arg0)

Dentro de este método creamos un contador auxiliar, y recorremos las posiciones del juego verificando si se apretó un botón, de ser asi, se aumenta el contador de turno, y se llama al metodo EncenderLuz() en esa posición, se actualiza el tablero y verifica si todas las luces fueron apagadas a través del método VerificarGanador()

private void VerificarGanador()

En este método se verifica si el objeto Juego, dio true como resultado al método VerificarGanador de su misma clase, de ser así, remueve gráficamente el tablero para que no se siga usando, actualiza la interfaz y muestra un mensaje de finalización del juego, dando la opción de volver a jugar o cerrar la ventana.

De esta manera llegamos a una implementación en la cual podemos graficar todos los cambios que van ocurriendo en el objeto Tablero según la inteligencia implementada de la clase Juego, sin mezclar las responsabilidades de cada una de las clases.