import javax.swing.\*;

import java.awt.\*;

import java.awt.event.\*;

public class BuscaMinas extends JFrame implements ActionListener {

private int filas = 10;

private int columnas = 10;

private int numMinas = 10;

private JButton[][] botones;

private boolean[][] minas;

private boolean[][] descubiertos;

private boolean juegoTerminado = false;

public BuscaMinas() {

setTitle("Busca Minas");

setSize(500, 500);

setDefaultCloseOperation(JFrame.EXIT\_ON\_CLOSE);

botones = new JButton[filas][columnas];

minas = new boolean[filas][columnas];

descubiertos = new boolean[filas][columnas];

JPanel panelJuego = new JPanel();

panelJuego.setLayout(new GridLayout(filas, columnas));

// Crear botones y añadirlos al panel

for (int i = 0; i < filas; i++) {

for (int j = 0; j < columnas; j++) {

botones[i][j] = new JButton();

botones[i][j].addActionListener(this);

panelJuego.add(botones[i][j]);

}

}

getContentPane().add(panelJuego);

generarMinas();

setVisible(true);

}

private void generarMinas() {

// Colocar minas aleatoriamente

for (int i = 0; i < numMinas; i++) {

int fila = (int) (Math.random() \* filas);

int columna = (int) (Math.random() \* columnas);

if (!minas[fila][columna]) {

minas[fila][columna] = true;

} else {

i--;

}

}

}

private void mostrarMinas() {

for (int i = 0; i < filas; i++) {

for (int j = 0; j < columnas; j++) {

if (minas[i][j]) {

botones[i][j].setText("X");

}

}

}

}

private void descubrirCasilla(int fila, int columna) {

if (fila < 0 || fila >= filas || columna < 0 || columna >= columnas || descubiertos[fila][columna]) {

return;

}

if (minas[fila][columna]) {

juegoTerminado = true;

mostrarMinas();

JOptionPane.showMessageDialog(this, "¡Has perdido!");

return;

}

int minasAlrededor = contarMinasAlrededor(fila, columna);

if (minasAlrededor > 0) {

botones[fila][columna].setText(String.valueOf(minasAlrededor));

descubiertos[fila][columna] = true;

} else {

botones[fila][columna].setText("");

descubiertos[fila][columna] = true;

descubrirCasilla(fila - 1, columna - 1);

descubrirCasilla(fila - 1, columna);

descubrirCasilla(fila - 1, columna + 1);

descubrirCasilla(fila, columna - 1);

descubrirCasilla(fila, columna + 1);

descubrirCasilla(fila + 1, columna - 1);

descubrirCasilla(fila + 1, columna);

descubrirCasilla(fila + 1, columna + 1);

}

if (esVictoria()) {

juegoTerminado = true;

JOptionPane.showMessageDialog(this, "¡Has ganado!");

}

}

private int contarMinasAlrededor(int fila, int columna) {

int count = 0;

for (int i = fila - 1; i <= fila + 1; i++) {

for (int j = columna - 1; j <= columna + 1; j++) {

if (i >= 0 && i < filas && j >= 0 && j < columnas && minas[i][j]) {

count++;

}

}

}

return count;

}

private boolean esVictoria() {

for (int i = 0; i < filas; i++) {

for (int j = 0; j < columnas; j++) {

if (!minas[i][j] && !descubiertos[i][j]) {

return false;

}

}

}

return true;

}

public void actionPerformed(ActionEvent e) {

if (juegoTerminado) {

return;

}

for (int i = 0; i < filas; i++) {

for (int j = 0; j < columnas; j++) {

if (e.getSource() == botones[i][j]) {

descubrirCasilla(i, j);

return;

}

}

}

}

public static void main(String[] args) {

new BuscaMinas();

}

}



