**import** java.awt.Dimension;

**import** java.awt.Graphics;

**import** java.awt.event.MouseEvent;

**import** java.awt.event.MouseListener;

**import** java.awt.event.MouseMotionListener;

**import** java.awt.image.BufferStrategy;

**import** java.io.BufferedReader;

**import** java.io.BufferedWriter;

**import** java.io.FileReader;

**import** java.io.FileWriter;

**import** java.io.IOException;

**import** javax.swing.JFrame;

**import** javax.swing.JFrame;

**public** **class** Main **extends** JFrame **implements** Runnable, MouseListener, MouseMotionListener {

/\*\*

\*

\*/

**private** **static** **final** **long** ***serialVersionUID*** = 1L;

**private** **static** **final** Dimension ***WindowSize*** = **new** Dimension(800,800);

**private** BufferStrategy strategy;

**private** **boolean**[][][] State = **new** **boolean**[40][40][2];

**private** **boolean** playing = **false**;

**private** Graphics offscreenbuffer;

**int** gamestatefrontbuffer;

**private** **boolean** initialised = **false**;

**public** Main() {

**this**.setTitle("Game of life");

setDefaultCloseOperation(JFrame.***EXIT\_ON\_CLOSE***);

Dimension screensize = java.awt.Toolkit.*getDefaultToolkit*().getScreenSize();

**int** x = screensize.width/ - ***WindowSize***.width/2 ;

**int** y = screensize.height/ - ***WindowSize***.height/2 ;

setBounds(x,y,***WindowSize***.width,***WindowSize***.height);

setVisible(**true**);

createBufferStrategy(2);

strategy = getBufferStrategy();

offscreenbuffer = strategy.getDrawGraphics();

addMouseListener(**this**);

addMouseMotionListener(**this**);

**for**(**int** X=0;X<40;X++) {

**for**(**int** Y=0;Y<40;Y++) {

State[X][Y][0]=State[X][Y][1]=**false**;

}

}

Thread t = **new** Thread(**this**);

t.start();

initialised = **true**;

}

**public** **void** run() {

// **TODO** Auto-generated method stub

**while**(1 ==1) {

**try** {

Thread.*sleep*(100);

}**catch**(InterruptedException e) {}

game();

**this**.repaint();

}

}

**public** **void** game() {

**int** front = gamestatefrontbuffer;

**int** back = (front+1)%2;

**for** (**int** x=0;x<40;x++) {

**for** (**int** y=0;y<40;y++) {

**int** count=0;

**for** (**int** xx=-1;xx<=1;xx++) {

**for** (**int** yy=-1;yy<=1;yy++) {

**if** (xx!=0 || yy!=0) {

**int** xxx = x+xx;

**if**(xxx<0)

xxx=39;

**else** **if**(xxx>39)

xxx=0;

**int** yyy=y+yy;

**if**(yyy<0)

yyy=39;

**else** **if**(yyy>39)

yyy=0;

**if**(State[x][y][front])

count++;

}

}

}

**if**(State[x][y][front]) {

**if**(count<2)

State[x][y][back] = **false**;

**else** **if**(count<4)

State[x][y][front] = **true**;

**else**

State[x][y][back] = **false**;

}

**else** {

**if**(count == 3)

State[x][y][back] = **true**;

**else**

State[x][y][back] = **false**;

}

}

}

gamestatefrontbuffer = back;

}

**public** **void** mouseDragged(MouseEvent e) {

**if**(playing == **true**) {

**int** X = e.getX()/20;

**int** Y = e.getY()/20;

State[X][Y][gamestatefrontbuffer] =

!State[X][Y][gamestatefrontbuffer];

**this**.repaint();

}

}

@Override

**public** **void** mouseMoved(MouseEvent e) {

**if**(playing == **true**) {

**int** X = e.getX()/20;

**int** Y = e.getY()/20;

State[X][Y][gamestatefrontbuffer] =

!State[X][Y][gamestatefrontbuffer];

**this**.repaint();

}

}

**public** **void** mouseClicked(MouseEvent e) {

**int** X = e.getX()/20;

**int** Y = e.getY()/20;

**if**(e.getX()>=50 && e.getX()<=140 && e.getY()>=50 && e.getY()<= 80)

{

playing = **true**;

}

**if**(e.getX()>= 168 && e.getX()<=218 && e.getY()>=50 && e.getY()<= 80)

{

playing = **true**;

randomise();

**this**.repaint();

}

**if**(playing == **true**) {

**if**(X>=420 && X<=465 && Y>=50 && Y<= 80)

{

writefile(**null**);

}

**else** **if**(X>=490 && X<=535 && Y>=50 && Y<= 80)

{

readfile();

}

State[X][Y][gamestatefrontbuffer] =

!State[X][Y][gamestatefrontbuffer];

}

}

**public** **void** randomise() {

**for**(**int** X=0;X<40;X++) {

randomise();

}

}

@Override

**public** **void** mouseEntered(MouseEvent arg0) {

// **TODO** Auto-generated method stub

}

@Override

**public** **void** mouseExited(MouseEvent arg0) {

// **TODO** Auto-generated method stub

}

@Override

**public** **void** mousePressed(MouseEvent e) {}

// **TODO** Auto-generated method stub

@Override

**public** **void** mouseReleased(MouseEvent arg0) {

// **TODO** Auto-generated method stub

}

**public** **void** paint(Graphics g) {

**if**(!initialised)

**return**;

g = offscreenbuffer;

g.setColor(Color.***BLACK***);

g.fillRect(0, 0, ***WindowSize***.width, ***WindowSize***.height);

g.setColor(Color.***WHITE***);

**for**(**int** X = 0;X <40;X++) {

**for**(**int** Y = 0; Y < 40;Y++) {

**if**(State[X][Y][gamestatefrontbuffer])

{

g.fillRect(X\*20, Y\*20, 20, 20);

}

}

}

g.setColor(Color.***GREEN***);

g.fillRect(50, 50, 90, 20);

g.fillRect(160, 50, 90, 20);

g.fillRect(420, 50, 45, 20);

g.fillRect(490, 50, 45, 20);

g.setColor(Color.***BLACK***);

g.drawString("START GAME", 55, 65);

g.drawString("RANDOMIZE", 168, 65);

g.drawString("SAVE", 426, 65);

g.drawString("LOAD", 496, 65);

strategy.show();

}

**public** **static** **void** main(String[] args) {

Main w = **new** Main();

}

**public** **void** writefile(String s)

{

String filename = "C:\\Users\\lois\\Desktop\\lifegame.txt" ;

**try** {

BufferedWriter writer = **new** BufferedWriter(**new** FileWriter(filename));

writer.write(s);

writer.close();

}

**catch** (IOException e) { }

}

**public** **void** readfile() {

String line=**null**;

String filename = "C:\\Users\\lois\\Desktop\\lifegame.txt";

**try** {

BufferedReader reader = **new** BufferedReader(**new** FileReader(filename));

**do** {

**try** {

line = reader.readLine();

} **catch** (IOException e) { }

}

**while** (line != **null**);

reader.close();

} **catch** (IOException e) { }

}

}