/\* Bryce Cheng-Campbell, Period 3

\* time taken: 3 hours

\* it wasn't that hard to read the header and know how many colors to read in and all of the

\* different attributes to the picture, but for the longest time I could not get the color

\* array to work correctly because of the padding. One sample picture I used had three

\* bits per pixel but NO padding. so the picture was garbled when i read what I thought was padding,

\* until i finally opened it in the hex editor. Then another

\* picture i had, was using four bits per pixel and had one bit of padding but even though

\* I am reading and throwing that bit away the picture still came out messed up.

\*/

**import** java.awt.Color;

**import** java.awt.Graphics;

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

**import** java.io.DataInputStream;

**import** java.io.FileInputStream;

**import** javax.swing.BorderFactory;

**import** javax.swing.JFrame;

**import** javax.swing.JPanel;

**public** **class** BMPReader {

**static** **int** *BMPSize*, *ArrayOffset*, *DIBSize*, *width*, *height*, *compress*, *arraySize*, *hRes*, *vRes*, *numColor*, *impColor*;

**static** **short** *fileType*, *spec1*, *spec2*, *colorPlanes*, *pixBit*;

**static** String *magic* = "";

**static** Color[][] *colors* = **null**;

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

*loadBMP*("test.bmp");

*printBMPHeader*();

*drawBMP*();

}

**public** **static** **void** loadBMP(String fileName){

FileInputStream fIn = **null**;

DataInputStream dIn = **null**;

**try** {

fIn = **new** FileInputStream(fileName);

dIn = **new** DataInputStream(fIn);

*magic* += (**char**)dIn.readByte();

*magic* += (**char**)dIn.readByte();

**if**(*magic*.equals("BM")){

//BMP header

*BMPSize* = dIn.readInt();

*spec1* = *flip*(dIn.readShort());

*spec2* = *flip*(dIn.readShort());

*ArrayOffset* = *flip*(dIn.readInt());

//DIB header

*DIBSize* = *flip*(dIn.readInt());

*width* = *flip*(dIn.readInt());

*height* = *flip*(dIn.readInt());

*colorPlanes* = *flip*(dIn.readShort());

*pixBit* = *flip*(dIn.readShort());

*compress* = *flip*(dIn.readInt());

*arraySize* = *flip*(dIn.readInt());

*hRes* = *flip*(dIn.readInt());

*vRes* = *flip*(dIn.readInt());

*numColor* = *flip*(dIn.readInt());

*impColor* = *flip*(dIn.readInt());

**int** r = 0, g = 0, b = 0, count = 0;

*colors* = **new** Color[*height*][*width*];

**for**(**int** x = *height*-1; x >= 0; x--){

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

**if**(count == *compress* && dIn.available() > 1 && *compress* > 0){

dIn.readByte();

count = 0;

}

**if**(dIn.available() > 3){

b = dIn.readUnsignedByte();

g = dIn.readUnsignedByte();

r = dIn.readUnsignedByte();

}

*colors*[x][y] = **new** Color(r, g, b);

count++;

}

}

}

**else**{

System.*out*.println("wrong file format");

**return**;

}

dIn.close();

} **catch** (Exception e) {e.printStackTrace();}

}

**public** **static** **int** flip (**int** value){

**int** b1 = (value >> 0) & 255;

**int** b2 = (value >> 8) & 255;

**int** b3 = (value >> 16) & 255;

**int** b4 = (value >> 24) & 255;

**return** b1 << 24 | b2 << 16 | b3 << 8 | b4 << 0;

}

**public** **static** **short** flip (**short** value){

**int** b1 = value & 255;

**int** b2 = (value >> 8) & 255;

**return** (**short**) (b1 << 8 | b2 << 0);

}

**public** **static** **void** drawBMP() {

BMP gui = **new** BMP(*width*, *height*, *colors*);

}

**public** **static** **void** drawBMP(**int** x, **int** y) {

}

**public** **static** **void** printBMPHeader(){

System.*out*.println("Width: " + *width*);

System.*out*.println("Height: " + *height*);

System.*out*.println("Color Planes: " + *colorPlanes*);

System.*out*.println("Bits per pixel: " + *pixBit*);

System.*out*.println("Horizontal Res: " + *hRes*);

System.*out*.println("Vertical Res: " + *vRes*);

System.*out*.println("Amount of bits: " + *arraySize*);

System.*out*.println("Compression: " + *compress*);

/\*for(int x = 0; x < width; x++)

System.out.println(Arrays.toString(colors[x]));\*/

}

}

**class** BMP {

MyDrawingPanel drawingPanel;

**int** width = 0, height = 0;

Color[][] colors = **null**;

BMP(**int** w, **int** h, Color[][] c) {

width = w;

height = h;

colors = c;

JFrame window = **new** JFrame("Picture");

window.setBounds(0, 0, width+20, height+40);

window.setResizable(**true**);

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

drawingPanel = **new** MyDrawingPanel();

drawingPanel.setBounds(0, 0, width, height);

drawingPanel.setBorder(BorderFactory.*createEtchedBorder*());

JPanel mainPanel = **new** JPanel();

mainPanel.setLayout(**null**);

mainPanel.add(drawingPanel);

window.getContentPane().add(mainPanel);

window.setVisible(**true**);

}

@SuppressWarnings("serial")

**private** **class** MyDrawingPanel **extends** JPanel{

MouseEvent mouse = **null**;

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

**for**(**int** x = width-1; x >= 0; x--){

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

g.setColor(colors[y][x]);

g.drawLine(x, y, 1, 1);

}

}

}

}

}