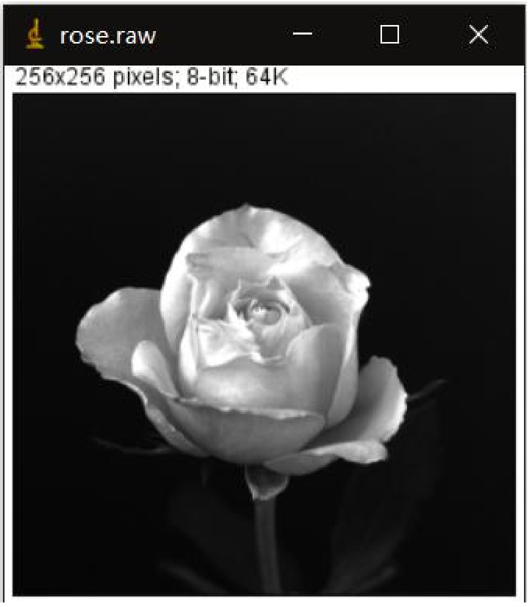
Course: CS 825
Name: Chao Zhang
Student#: 200383834
Programming language: JAVA

Programming 1

The image“rose.raw” (256 x 256)

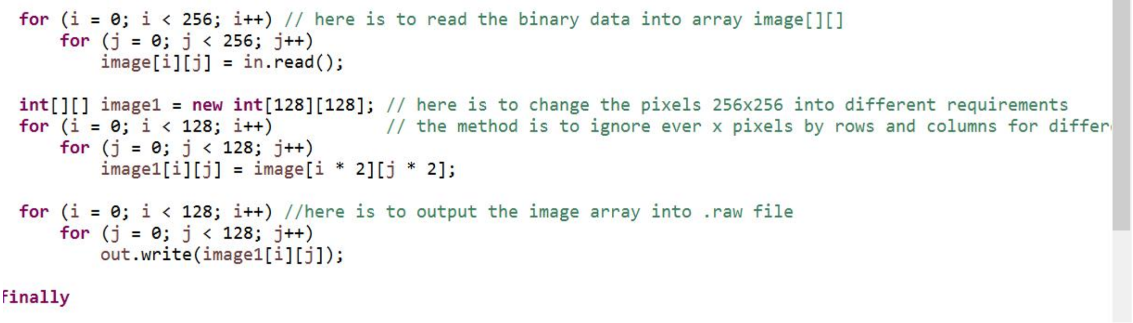


```
//source code
import java.io.*;

public class programming1 {
    public static void main(String args[]) throws IOException {
        FileInputStream in = null;
        FileOutputStream out = null;
        try {
            in = new FileInputStream("D:/rose.raw");    // file path to read
            out = new FileOutputStream("D:/rose+.raw"); // file path to write
            int i = 0, j = 0;
            int[][] image = new int[256][256];

            for (i = 0; i < 256; i++) // here is to read the binary data into array
image[][]
                for (j = 0; j < 256; j++)
                    image[i][j] = in.read();

            int[][] image1 = new int[256][256]; // here is to change the pixels
256x256 into different requirements
            for (i = 0; i < 256; i++)                // the method is to ignore ever x
pixels by rows and columns for different requirements
                for (j = 0; j < 256; j++)
                    image1[i][j] = image[i][j];

            for (i = 0; i < 128; i++) //here is to output the image array into .raw file
                for (j = 0; j < 128; j++)
                    out.write(image1[i][j]);

        } finally

        {
            if (in != null) {
                in.close();
            }
            if (out != null) {
                out.close();
            }
        }
    }
}
```

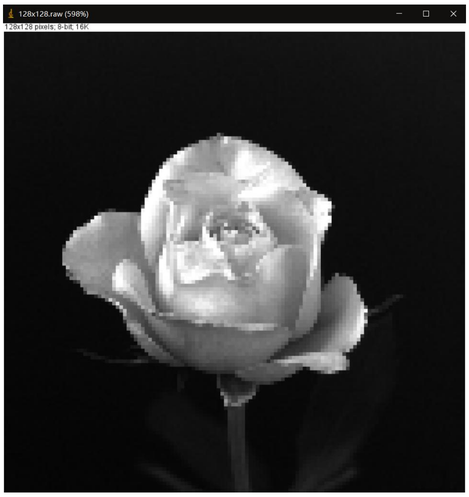
Three smaller-sized versions of the image

(1) 128x128

To ignore every 2 pixel per rows and columns to get the new picture

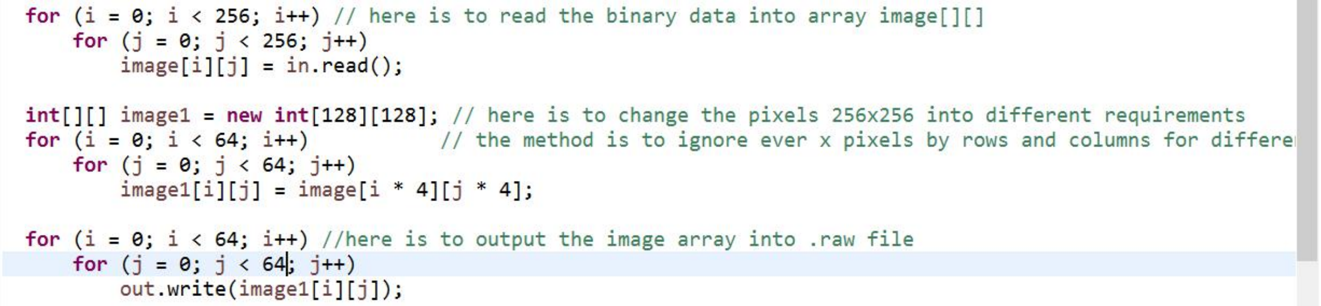


```
 for (i = 0; i < 256; i++) // here is to read the binary data into array image[][]
     for (j = 0; j < 256; j++)
         image[i][j] = in.read();

 int[][] image1 = new int[128][128]; // here is to change the pixels 256x256 into different requirements
 for (i = 0; i < 128; i++)           // the method is to ignore ever x pixels by rows and columns for differe
     for (j = 0; j < 128; j++)
         image1[i][j] = image[i * 2][j * 2];

 for (i = 0; i < 128; i++) //here is to output the image array into .raw file
     for (j = 0; j < 128; j++)
         out.write(image1[i][j]);

Finally
```

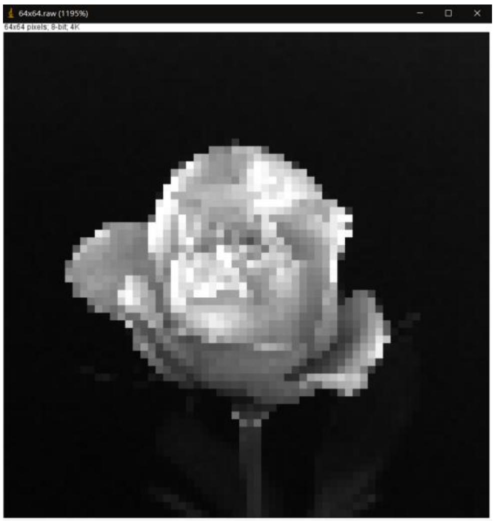


(2) 64x64

To ignore every 4 pixel per rows and columns to get the new picture

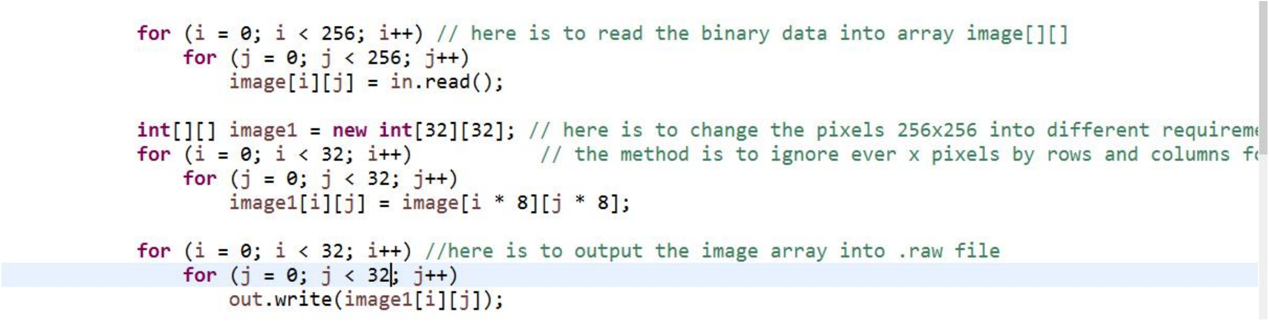


```
for (i = 0; i < 256; i++) // here is to read the binary data into array image[][]
    for (j = 0; j < 256; j++)
        image[i][j] = in.read();

int[][] image1 = new int[128][128]; // here is to change the pixels 256x256 into different requirements
for (i = 0; i < 64; i++)           // the method is to ignore ever x pixels by rows and columns for differe
    for (j = 0; j < 64; j++)
        image1[i][j] = image[i * 4][j * 4];

for (i = 0; i < 64; i++) //here is to output the image array into .raw file
    for (j = 0; j < 64; j++)
        out.write(image1[i][j]);
```

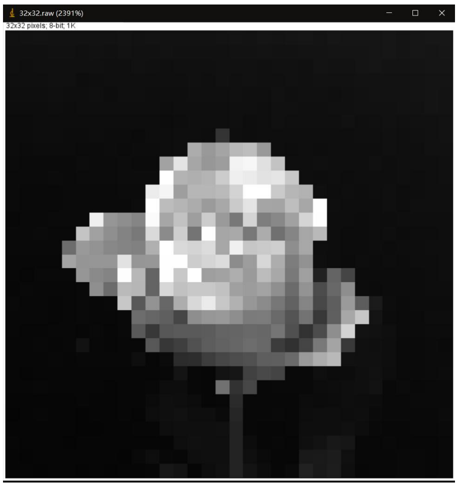


(2) 32x32

To ignore every 8 pixel per rows and columns to get the new picture



```
for (i = 0; i < 256; i++) // here is to read the binary data into array image[][]
    for (j = 0; j < 256; j++)
        image[i][j] = in.read();

int[][] image1 = new int[32][32]; // here is to change the pixels 256x256 into different requireme
for (i = 0; i < 32; i++)           // the method is to ignore ever x pixels by rows and columns fo
    for (j = 0; j < 32; j++)
        image1[i][j] = image[i * 8][j * 8];

for (i = 0; i < 32; i++) //here is to output the image array into .raw file
    for (j = 0; j < 32; j++)
        out.write(image1[i][j]);
```



Conclusion: Reducing the size of an image is reducing the resolution, which means the effects are that the image lose the pixels and become fuzzier.

Programming 2

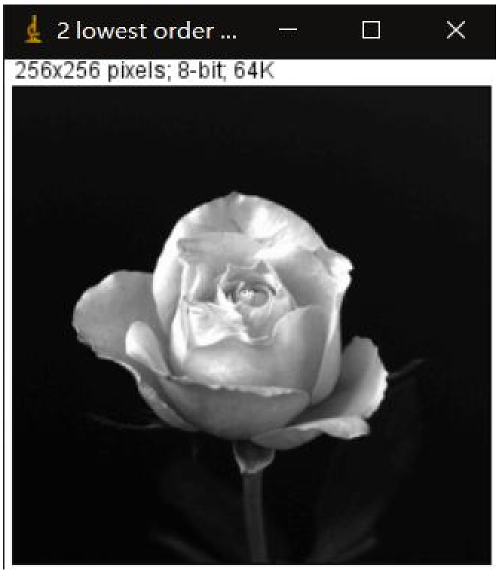
```
//source code
import java.io.*;

public class programming2 {
    public static void main(String args[]) throws IOException {
        FileInputStream in = null;
        FileOutputStream out = null;
        try {
            in = new FileInputStream("D:/rose.raw"); // file path to read
            out = new FileOutputStream("D:/rose+.raw"); // file path to write
            int i = 0, j = 0;
            int[][] image = new int[256][256];
            int r,t;
            int order=4,sum=0; // variable order is the numbers of the lowest order
bits
            int[] b = new int[8];
            for (i = 0; i < 256; i++)
                for (j = 0; j < 256; j++)
                    image[i][j] = in.read();
            int[][] image1 = new int[256][256];
            for (i = 0; i < 256; i++)
                for (j = 0; j < 256; j++) {
                    t = image[i][j];
                    for (int k = 0; k < 8; k++) {    // here is to change decimalism
data into binary array b[]
```

```
                        r = t % 2;
                        b[k] = r;
                        t /= 2;
                    }
                    for (int k = 0; k < order; k++) b[k]=0;  // here is to set 0
according to requirements
                    for (int k = 0; k < 8; k++) sum+=(Math.pow(2,k))*b[k]; // put
processed binary data array into output array image1
                    image1[i][j]=sum;
                    sum=0;
                }
            for (i = 0; i < 256; i++)
                for (j = 0; j < 256; j++)
                    out.write(image1[i][j]);
        } finally

        {
            if (in != null) {
                in.close();
            }
            if (out != null) {
                out.close();
            }
        }
    }
}
```

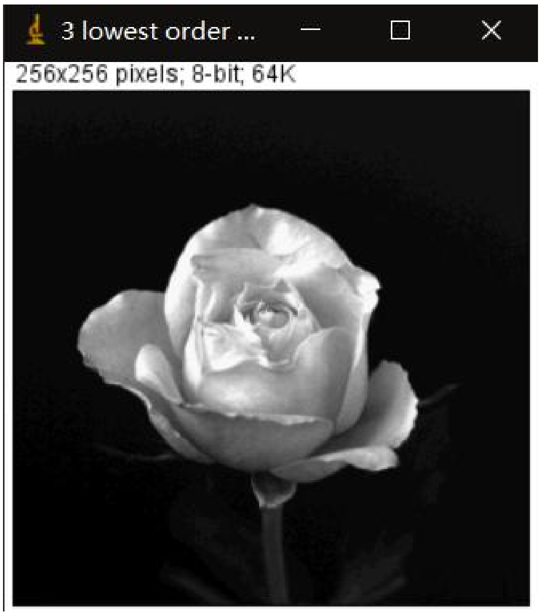
Create three different quantized versions of the image

In the first, the 2 lowest order bits are set to 0

Switch the variable order to 2

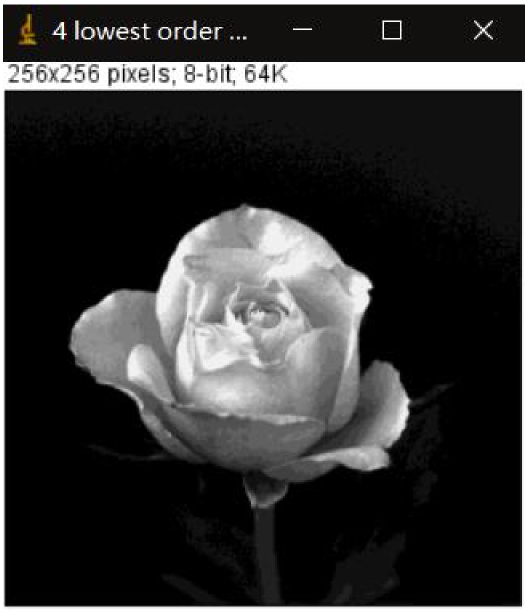


In the second, the 3 lowest order bits are set to 0
Switch the variable order to 3


3 lowest order ...

256x256 pixels; 8-bit; 64K

In the third, the 4 lowest order bits are set to 0
Switch the variable order to 4


4 lowest order ...

256x256 pixels; 8-bit; 64K

Conclusion: Reducing the number of bits of each pixel is reducing the the dynamic range of the image and make it appear “unnatural".