PRACTICA 1 BIOMETRIA

TAREAS: Realizar un programa en Java que ejecute las siguientes tareas: convertir una imagen en color almacenada en el PC en dos imágenes en grises mediante las opciones A y B, salvando al PC el resultado.

1. **Definir la clase FingerPrintImage para guardar la imagen en grises:**

```
public class FingerPrintImage {
        private int width;
        private int height;
        private char[][] img;
        FingerPrintImage(int width, int height) {
                this.height = height;
                this.width = width;
                img = new char[width][height];}
        public int getHeight() {
                return height;}
        public int getWidth() {
                return width;}
        public void setPixel(int x, int y, char color) {
                img[x][y] = color;}
        public char getPixel(int x, int y) {
                return img[x][y];}
    }
```

2. **Cargar una imagen color desde un archivo con ImageIO.read**

```
BufferedImage imagenentrada = null;
imagenentrada = ImageIO.read(new File("imagen.png"));
System.out.println("Cargada la imagen");
```

3. **Convertir la imagen de RGB a una matriz de grises de 0 a 255:**

```
Entrada: BufferedImage imagenentrada
Salida:     FingerPrintImage imagensalida;
{
   for (int x = 0; x < imagenentrada.getWidth(); ++x){
     for (int y = 0; y < imaganentrada.getHeight(); ++y){
       int rgb = imagenentrada.getRGB(x, y);
       int r = (rgb >> 16) & 0xFF;
       int g = (rgb >> 8) & 0xFF;
       int b = (rgb & 0xFF);
       int nivelGris = (r + g + b) / 3;
       imagensalida.setPixel(x, y, nivelGris);
   }
  }
return imagensalida;
}
Opción B: Realizar el cálculo ponderado del gris utilizando:
Gris= 0.2126*R + 0.7152*G + 0.0722*B
```

4. **Convertir la imagen de grises a RGB para poder salvarla, modo 1 Grises:**

```
  Entrada:  FingerPrintImage imagenentrada;
            modo= 1 imagen en grises
  BufferedImage imagensalida = new BufferedImage(imagenentrada.getWidth(), imagenentrada.getHeight(),
  BufferedImage.TYPE_XXX_XXX);

            for (int x = 0; x < imagenentrada.getWidth(); ++x) {
                    for (int y = 0; y < imagenentrada.getHeight(); ++y) {
                            int valor = imagenentrada.getPixel(x, y);
                            if (modo == 0) {
                                    valor = valor * 255;
                            }
                            int pixelRGB = (255 << 24 | valor << 16 | valor << 8 | valor);
                            imagensalida.setRGB(x, y, pixelRGB);
                    }
            }
            return imagensalida;
                    }
```

5. **Guardar la imagen a un archivo con ImageIO.write:**

```
File outputfile = new File("imagen_en_grises.png");
ImageIO.write(bi, "png", outputfile);
```

6. **Entregar el código, la imagen original y las imágenes en grises mediante el método A (valor medio) y método B (colores ponderados) y un pdf con las funciones e imágenes de entrada y sálida.**