

**PROGRAM TITLE:Reduce the Noise in an image using Median Method.**

**PROGRAM CODE:**

```
import java.io.*;
import java.awt.Color;
import java.awt.image.*;
import javax.imageio.*;

class NoiseReduceMedian
{
    String meta = "";
    int length, width;
    int image[][];

    public NoiseReduceMedian(String imgLoc)//Reading the image
    {
        try
        {
            BufferedReader br = new BufferedReader(new
FileReader(imgLoc));
            meta += br.readLine() + "\n";//First Line is Format
            meta += br.readLine() + "\n";//Comment Line

            String rc = br.readLine();//3rd Line Contains Length and
Breadth
            length = Integer.parseInt(rc.split(" ")[0]);
            System.out.println("Length = " + length);
            width = Integer.parseInt(rc.split(" ")[1]);
            System.out.println("Width = " + width);

            meta += br.readLine() + "\n";//Contains Max Intensity

            image = new int[length][width];

            for (int i = 0; i < length; i++)
                for (int j = 0; j < width; j++)
                    image[i][j] = Integer.parseInt(br.readLine());

            br.close();
        }
        catch (Exception e) {
            System.out.println(e);
        }
    }

    public void reduceNoise()
    {
        int newImage[][] = new int[length][width];
        for (int i = 0; i < length; i++)
        {
            for (int j = 0; j < width; j++)
            {
                if (i == 0 || j == 0 || i == length - 1 || j == width - 1)
                    newImage[i][j] = image[i][j];
                else
```

```

        newImage[i][j] = getMedian(i, j);
    }
}
image = newImage;
}

public int getMedian(int x, int y)
{
    //Taking 8 Nearest Neighbour Values
    int values[] = { image[x - 1][y + 1], image[x - 1][y], image[x
- 1][y - 1], image[x][y + 1], image[x][y],
        image[x][y - 1], image[x + 1][y - 1], image[x + 1][y],
image[x + 1][y + 1] };
    //Sorting to find Median
    for (int i = 0; i < values.length; i++)
    {
        for (int j = 0; j < values.length - i - 1; j++)
        {
            if (values[j] > values[j + 1])
            {
                int temp = values[j];
                values[j] = values[j + 1];
                values[j + 1] = temp;
            }
        }
    }
    //Returning Median
    return values[values.length / 2];
}

public void output()
{
    try
    {
        PrintWriter printer = new PrintWriter(new
FileWriter("./img/o-noise-reduced-median.pgm"));
        printer.println(meta.split("\n")[0]);
        printer.println(meta.split("\n")[1]);
        printer.println(length + " " + width);
        printer.println(meta.split("\n")[2]);

        for (int i = 0; i < length; i++)
        {
            for (int j = 0; j < width; j++)
            {
                printer.println(image[i][j]);
            }
        }
        printer.close();
        System.out.println("Image has been written to file");
    }
    catch (Exception e) {
        System.out.println(e);
    }
}

public static void main(String args[])
{

```

```
NoiseReduceMedian nr = new  
NoiseReduceMedian("./img/noisy.pgm");  
nr.reduceNoise();  
nr.output();  
}  
}
```

### OUTPUT :



**L:** Noisy Image **R:** Image after applying Noise Reduction with Median