

```

1  package Ejercicio02;
2
3  import java.io.BufferedReader;
4  import java.io.File;
5  import java.io.FileInputStream;
6  import java.io.FileNotFoundException;
7  import java.io.IOException;
8
9  import javax.swing.JFileChooser;
10
11 public class U6E02C {
12
13
14     public static void main(String[] args) throws IOException {
15         //variables globales
16         File file = leerFichero(); //objeto File. Llamamos al metodo para seleccionar el fichero
17         float[] cantidad = new float[256]; //para almacenar las veces que aparece cada uno de los 256 bytes
18         FileInputStream st=null; //objeto FileInputStream
19         BufferedReader bis=null; //objeto BufferedReader
20         int bytes=0; //almacena el total de bytes
21         long tiempo; //almacena el tiempo transcurrido
22         int pos = 0;
23         double entropia=0;
24
25
26         //rellenamos el array con ceros
27         for(int i = 0; i < cantidad.length; i++) {
28             cantidad[i] = 0;
29         }
30
31         //cogemos el tiempo de inicio
32         tiempo= System.currentTimeMillis();
33
34
35         try {
36
37             //creamos el FileInputStream
38             st = new FileInputStream(file);
39
40             //lo encapsulamos en un bufferedInputStream
41             bis = new BufferedReader(st);
42
43             //calculamos cuantos bytes tiene el archivo seleccionado
44             bytes = bis.available();
45
46             //creamos el buffer
47             byte [] buffer = new byte[64];
48
49             //leemos el primer byte
50             int valor=0;
51

```

```

52         //mientras hay bytes que leer almacenamos en cada posicion del array las veces que aparece
53         while(0 < (valor=bis.read(buffer))) {
54             for(int i = 0; i < valor; i++) {
55                 if (buffer[i] < 0) pos = buffer[i]+256;
56                 else pos = buffer[i];
57                 cantidad[pos]++;
58             }
59         }
60
61     } catch (FileNotFoundException e) {
62         System.out.println(e.getMessage());
63     } catch (IOException e) {
64         System.out.println(e.getMessage());
65     }
66
67     finally { //cerramos cosas
68         st.close();
69         bis.close();
70     }
71
72     //calculamos el histograma
73     for(int i = 0; i < cantidad.length; i++) cantidad[i] = cantidad[i]/bytes;
74
75     //calculamos la entropia
76     for(int i = 0; i < cantidad.length; i++) {
77         entropia = entropia + (cantidad[i]*(Math.log(1/cantidad[i])/Math.log(2)));
78     }
79
80     //paramos el cromo
81     tiempo = System.currentTimeMillis() - tiempo;
82
83
84     //mostramos el nombre del fichero, su peso y el tiempo de calculo del histograma
85     System.out.println("Fichero: " + file.getName());
86     System.out.print("Peso: " + bytes + " bytes. ");
87     System.out.printf("Proceso realizado en %5.4f segundos.", (float)tiempo/1000);
88     System.out.println();
89     System.out.printf("La entropia del fichero es: %7.6f bits", entropia);
90     System.out.println();
91     System.out.printf("La redundancia es %7.6f bits", 8-entropia);
92     System.out.println();
93     if(8-entropia < 0.1) System.out.println("Fichero comprimido o aleatorio");
94     else if(8-entropia >= 0.1 && 8-entropia <= 3.5) System.out.println("Fichero parcialmente comprimido");
95     else System.out.println("Fichero no comprimido");
96
97
98
99
100 }
101
102 //metodo para seleccionar un fichero

```

```
103     public static File leerFichero() {
104         //variables
105         File archivo;
106
107         //creamos el objeto
108         JFileChooser fd = new JFileChooser();
109
110         //mostramos por pantalla
111         fd.setDialogTitle("Selecciona el fichero a leer");
112
113         fd.setSelectedFile(null); //no hay ninguno seleccionado
114
115         int opcion = fd.showOpenDialog(null);
116
117         if (opcion != JFileChooser.APPROVE_OPTION ) {
118             archivo = null;
119             return archivo;
120         }
121
122         archivo = fd.getSelectedFile();
123         return archivo;
124     }
125 }
126
127
128
129
130
```