Cover Sheet

CV Project1: HistogramThreshold C++

Student: Fengzhang Du
Project Due Date: 02/14/2021

Algorithm Steps for Compute and Print the Histogram:

```
Step 0: read header form input.
Step 1: output numRows, numCols, minVal and maxVal to output file.
Step 2: hist [maxVal], dynamically allocate the hist array and filled with 0.
Step 3: read the input file 1 integer at a time, hist[current_value]++
Step 4: repeat until the file is empty.
Step 5: output to file 1 and close the file.
Step 6: move to the beginning of the input file for later use. No need to close it.
```

Algorithm Steps for Display the Histogram:

```
Step 0: output numRows, numCols, minVal and maxVal to output file.
Step 1: loop through the hist array.
        Output the index and the current value, padded with space if needed to align +.
Step 2: iterate through the current value, and print as much as the + sign.
        Stop when it's over 70 +. And continue the loop.
Step 5: output to file 2 and close the file.
```

Algorithm Steps for Computing the image with Threshold:

```
Step 0: read header form input.
Step 1: outFile3, outFile4 output numRows, numCols, minVal and maxVal.
Step 2: read from inFile one integer at a time. Store the whole image in a 2D array.
Step 3: iterate through the image,
        if current value >= thresholdVal
                outFile3 <-- write 1 followed by a space
                outFile4 <-- write 1 followed by a space
        else:
                outFile3 <-- write 0 followed by a space
                outFile4 <-- write . followed by a space
Step 4: repeat step 2 to 3 until the reach the end of the input file.
```

Steps for Main function:

```
step 0: open input file use argv[1]
        get thresholdVal from argv[2]
        open all 4 outFiles via argv[3], argv[4], argv[5], argv[6]
step 1: numRows, numCols, minVal, maxVal  read from inFile
step 2: hist array,  dynamically allocate and initialize to 0
step 3: ComputeHist (input)
step 4: printHist(outFile1)
step 5: dispHist (outFile2)
step 6: outFile3  "The threshold value uses is " thresholdVal
        outFile4  "The threshold value uses is " thresholdVal
step 7: threshold (inFile, outFile3, outFile4, thresholdVal)
step 8: close all files
```

Source Code

```
#include <iostream>
#include <fstream>
#include <cstdlib>
using namespace std;

class Image{
   public:
       int numRows=-1, numCols=-1, minVal=-1, maxVal=-1;
       int* hist; // 1D int array, size = maxVal+1;
       int** body; // 2D int array.
   // constructor
   public:
       Image(ifstream &input){
           read_header(input);
           // allocate new array as read in maxVal.
           hist = new int[maxVal+1];
           // initialize to all 0s.
           for (int i=0; i<maxVal+1; i++){
               hist[i]=0;
           }
           body = new int*[numRows];
           for(int i=0; i<numRows; i++){
               body[i] = new int[numCols];
               for(int j=0; j<numCols; j++){
                   body[i][j] = 0;
               }
           }
       }

   void read_header(ifstream &input) {
       input >> numRows >> numCols >> minVal >> maxVal;
   }

   void write_header(ofstream &w){
       w << numRows<< " " << numCols<< " " << minVal << " " << maxVal << endl;
   }

   void computeHist(ifstream &input){
       int num ;
       while( !input.eof() ) {
           input >> num ; // parse each char as int in the text skip whitespaces
           hist[num]++;
       }
       input.clear();
       input.seekg(0, input.beg);
   }

   void printHist(ofstream &output1){
       write_header(output1);
```

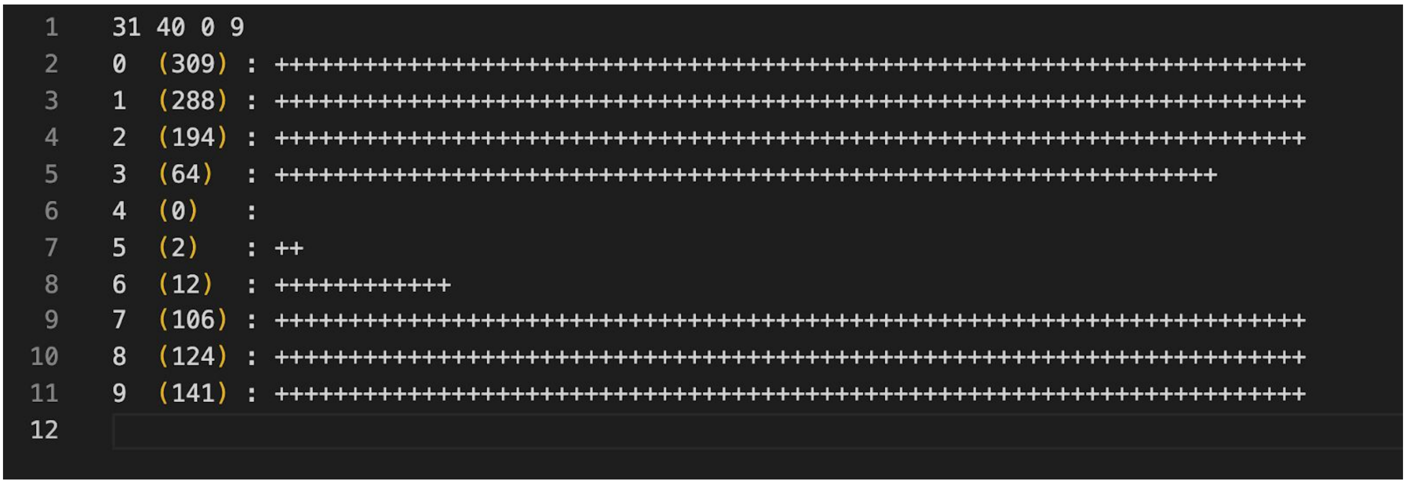
```
    for(int i=0; i<maxVal+1; i++){
        output1 << i << " " << hist[i] << endl;
    }
}

void dispHist(ofstream &output2){
    write_header(output2);
    for(int i=0; i<maxVal+1; i++){
        output2 << i << " ";
        if (i/10 == 0) output2 << " ";
        output2 << "(" << hist[i] << ") ";
        // align "+"s
        int temp = hist[i];
        if (temp/10 == 0){// 1-9
            output2 << "  ";
        }else if(temp/100 == 0){ //10-99
            output2 << " ";
        }else{  // 3 digits
        }
        output2 << ": ";

        int count = 0;
        for(int j=0; j<hist[i]; j++) {
            output2 << "+";
            count++;
            if(count >= 70) break;
        }
        output2 << endl;
    }
}

void threshold(ifstream &input, ofstream &output3, ofstream &output4, int thresholdVal){
    read_header(input);

    // output header;
    write_header(output3);
    write_header(output4);

    // read int from input file
    for(int i=0; i<numRows; ++i){
        for(int j=0; j<numCols; ++j){
            input >> body[i][j];
        }
    }
    // write to file3 and file4.
    for(int i=0; i<numRows; ++i){
        for(int j=0; j<numCols; ++j){
            if(body[i][j] >= thresholdVal){
                output3 << 1;
                output4 << 1;
            }else{
                    output3 << 0;
                    output4 << ".";
                }
                if((j+1) % (numCols) != 0) {
                    output3 << " ";
                    output4 << " ";
                }
            }
            if((i+1) % (numRows) != 0) {
                output3 << endl;
                output4 << endl;
            }
        }
    }
    void free_Heap(){
        for(int i=0; i<numRows; i++){
            delete[] body[i];
        }
        delete[] body;
        delete[] hist;
    }
};


int main(int argc, const char * argv[]) {
    ifstream input;
    input.open(argv[1]);

    int thresholdVal = atoi(argv[2]);
    cout << "The threshold value uses is " << thresholdVal << endl;


    ofstream output1, output2, output3, output4;
    output1.open(argv[3]);
    output2.open(argv[4]);
    output3.open(argv[5]);
    output4.open(argv[6]);

    if(input.is_open()){
        Image *img = new Image(input);
        img->computeHist(input);
        if(output1.is_open()){
            img->printHist(output1);
        }else{
            cout<<"Error: output file 1 is not open! "<<endl;
        }
        if(output2.is_open()){
            img->dispHist(output2);
        }else{
            cout<<"Error: output file 2 is not open! "<<endl;
        }
        if(output3.is_open() && output4.is_open()){
            img->threshold(input, output3, output4, thresholdVal);
        }else{
            cout<<"Error: output file 3 or file 4 are not open! "<<endl;
        }
        input.close();
        output1.close();
        output2.close();
        output3.close();
        output4.close();
    }else{
        cout<<"Error: input file is not open!" <<endl;
    }
    return 0;
}
```

Program Output

Output file 1 _ data1

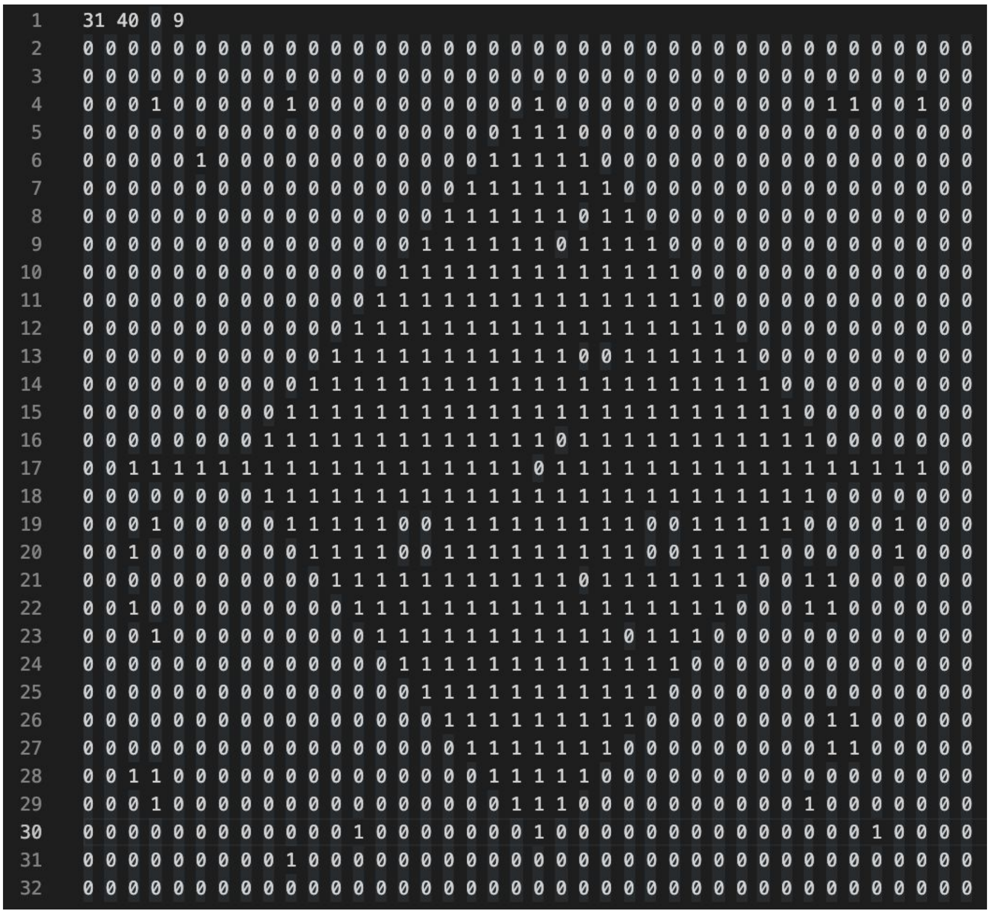
31 40 0 9
0 309
1 288
2 194
3 64
4 0
5 2
6 12
7 106
8 124
9 141

Output file 2 _ data1



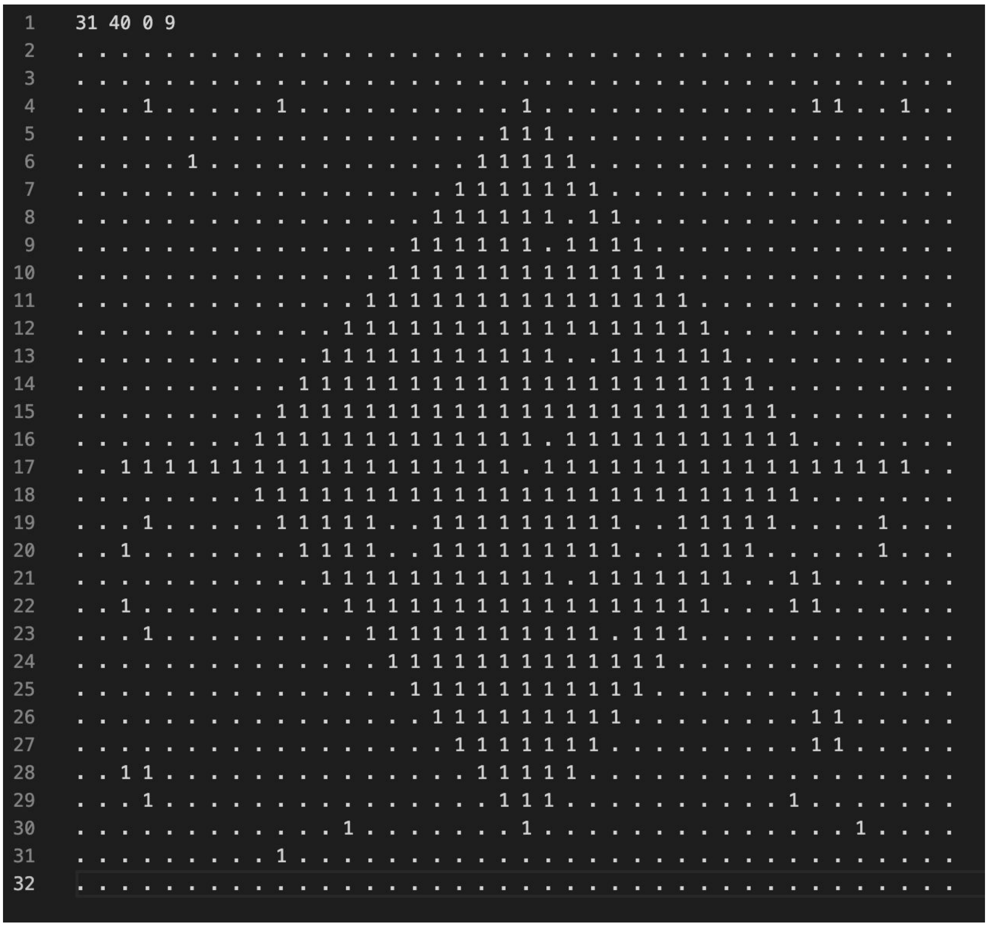
```
31 40 0 9
0 (309) : ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
1 (288) : ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
2 (194) : ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
3 (64)  : ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
4 (0)   :
5 (2)   : ++
6 (12)  : ++++++++++++
7 (106) : ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
8 (124) : ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
9 (141) : ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
```

Output file 3 _ data1



```
31 40 0 9
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 1 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 1 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 0 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 0 0 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0
0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0
0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0
0 0 0 1 0 0 0 0 0 1 1 1 1 1 0 0 1 1 1 1 1 1 1 1 1 0 0 1 1 1 1 1 0 0 0 0 1 0 0 0
0 0 1 0 0 0 0 0 0 0 1 1 1 1 0 0 1 1 1 1 1 1 1 1 1 0 0 1 1 1 1 0 0 0 0 0 1 0 0 0
0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 0 0 1 1 0 0 0 0 0 0
0 0 1 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 1 1 0 0 0 0 0 0
0 0 0 1 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0
0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0
0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

Output file 4 _ data1

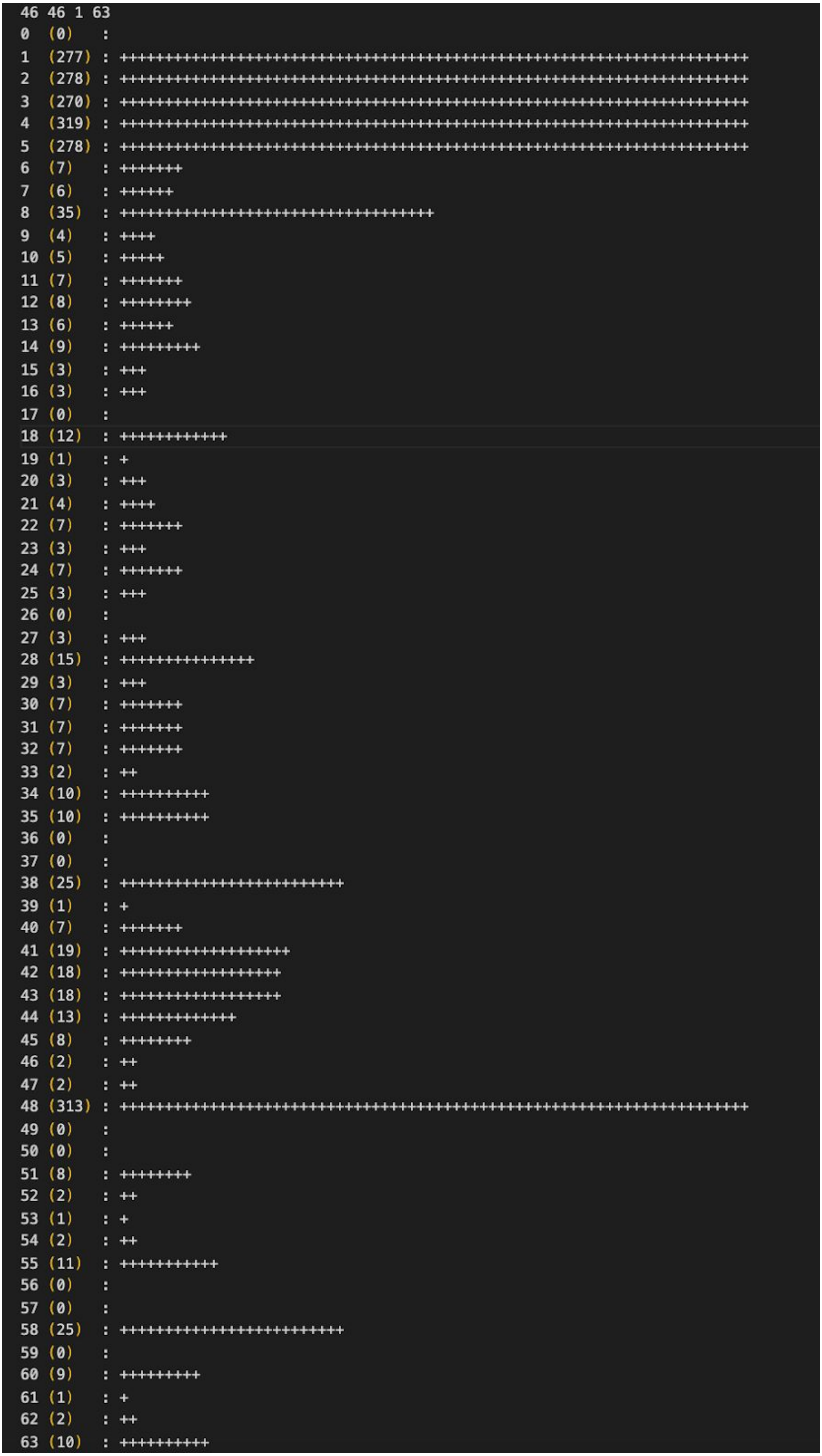


```
31 40 0 9
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
. . . 1 . . . . . 1 . . . . . . . . . . 1 . . . . . . . . . . . . 1 1 . . 1 . .
. . . . . . . . . . . . . . . . . . . 1 1 1 . . . . . . . . . . . . . . . . . .
. . . . . 1 . . . . . . . . . . . . 1 1 1 1 1 . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . 1 1 1 1 1 1 1 . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . 1 1 1 1 1 1 . 1 1 . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . 1 1 1 1 1 1 . 1 1 1 1 . . . . . . . . . . . . . .
. . . . . . . . . . . . . . 1 1 1 1 1 1 1 1 1 1 1 1 1 . . . . . . . . . . . . .
. . . . . . . . . . . . . 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 . . . . . . . . . . . .
. . . . . . . . . . . . 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 . . . . . . . . . . .
. . . . . . . . . . . 1 1 1 1 1 1 1 1 1 1 1 . . 1 1 1 1 1 1 . . . . . . . . . .
. . . . . . . . . . 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 . . . . . . . . .
. . . . . . . . . 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 . . . . . . . .
. . . . . . . . 1 1 1 1 1 1 1 1 1 1 1 1 1 . 1 1 1 1 1 1 1 1 1 1 1 . . . . . . .
. . 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 . 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 . .
. . . . . . . . 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 . . . . . . .
. . . 1 . . . . . 1 1 1 1 1 . . 1 1 1 1 1 1 1 1 1 . . 1 1 1 1 1 . . . . 1 . . .
. . 1 . . . . . . . 1 1 1 1 . . 1 1 1 1 1 1 1 1 1 . . 1 1 1 1 . . . . . 1 . . .
. . . . . . . . . . . 1 1 1 1 1 1 1 1 1 1 1 . 1 1 1 1 1 1 1 . . 1 1 . . . . . .
. . 1 . . . . . . . . . 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 . . . 1 1 . . . . . .
. . . 1 . . . . . . . . . 1 1 1 1 1 1 1 1 1 1 1 . 1 1 1 . . . . . . . . . . . .
. . . . . . . . . . . . . . 1 1 1 1 1 1 1 1 1 1 1 1 1 . . . . . . . . . . . . .
. . . . . . . . . . . . . . . 1 1 1 1 1 1 1 1 1 1 1 . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . 1 1 1 1 1 1 1 1 1 . . . . . . . . 1 1 . . . . .
. . . . . . . . . . . . . . . . . 1 1 1 1 1 1 1 . . . . . . . . . 1 1 . . . . .
. . 1 1 . . . . . . . . . . . . . . 1 1 1 1 1 . . . . . . . . . . . . . . . . .
. . . 1 . . . . . . . . . . . . . . . 1 1 1 . . . . . . . . . . 1 . . . . . . .
. . . . . . . . . . . . 1 . . . . . . . 1 . . . . . . . . . . . . . . 1 . . . .
. . . . . . . . . 1 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
```

Output file 1 _ data2

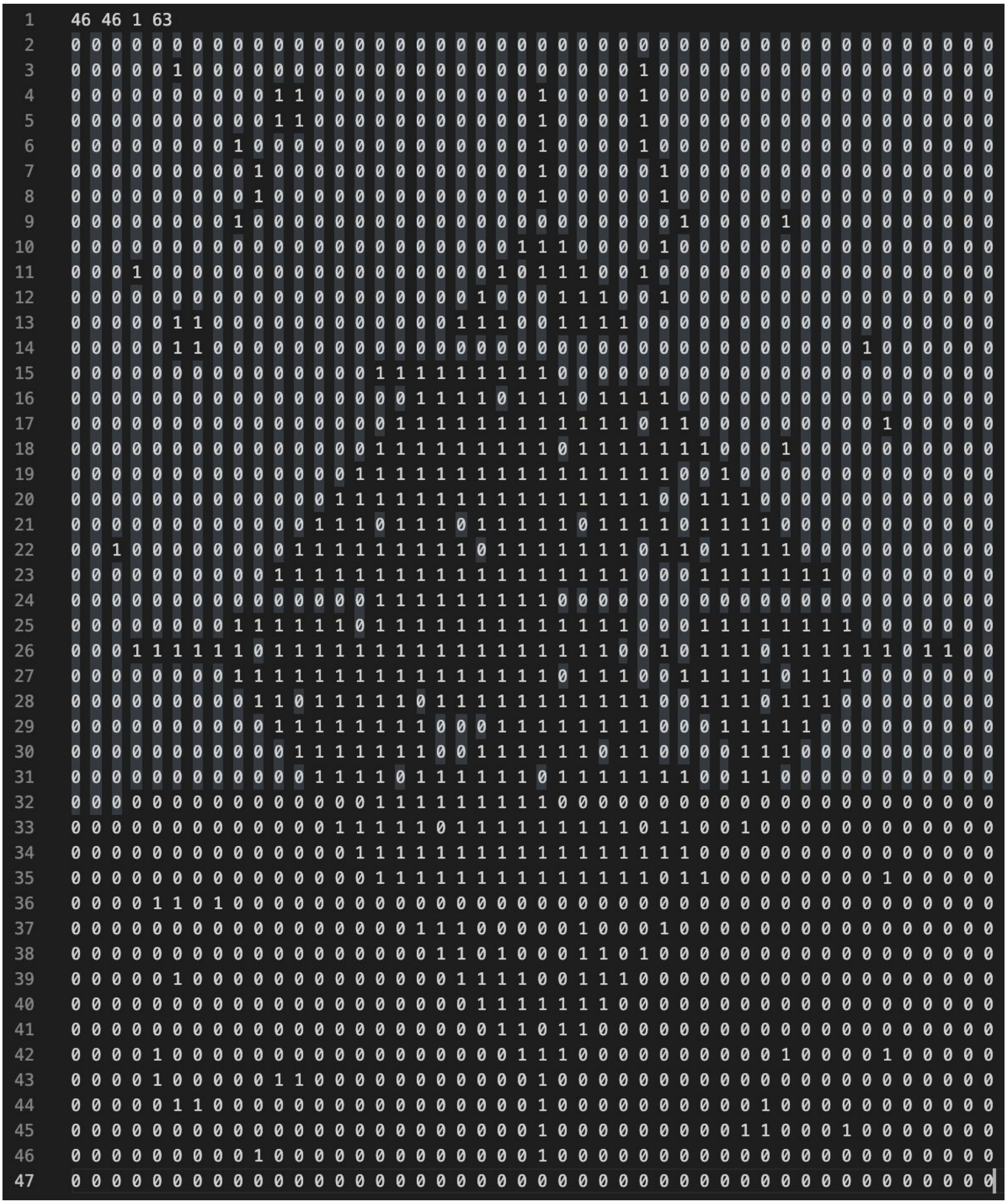
```
46 46 1 63
0 0
1 277
2 278
3 270
4 319
5 278
6 7
7 6
8 35
9 4
10 5
11 7
12 8
13 6
14 9
15 3
16 3
17 0
18 12
19 1
20 3
21 4
22 7
23 3
24 7
25 3
26 0
27 3
28 15
29 3
30 7
31 7
32 7
33 2
34 10
35 10
36 0
37 0
38 25
39 1
40 7
41 19
42 18
43 18
44 13
45 8
46 2
47 2
48 313
49 0
50 0
51 8
52 2
53 1
54 2
55 11
56 0
57 0
58 25
59 0
60 9
61 1
62 2
63 10
```

Output file 2 _ data2



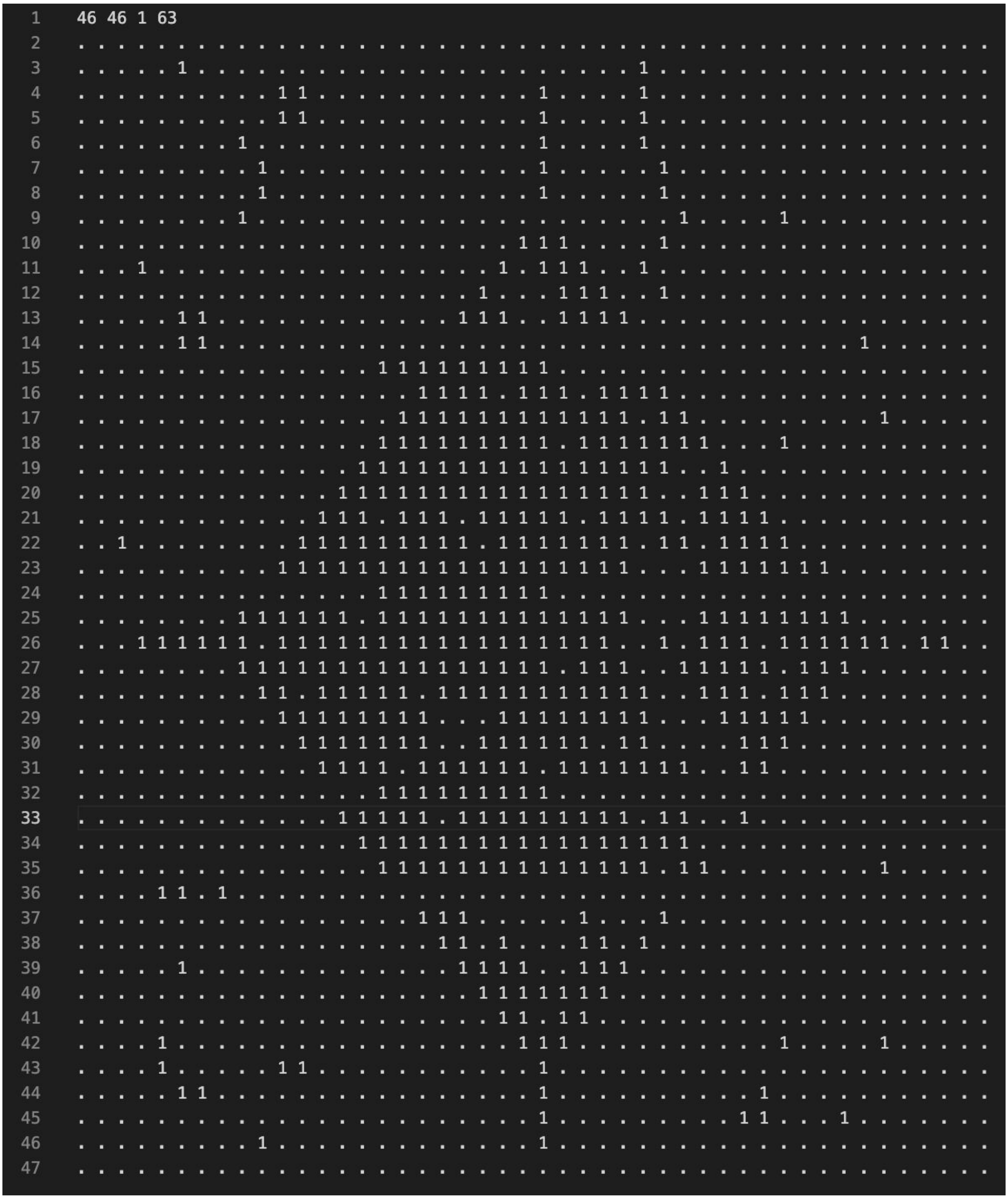
```
46 46 1 63
0  (0)   :
1  (277) : ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
2  (278) : ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
3  (270) : ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
4  (319) : ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
5  (278) : ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
6  (7)   : +++++++
7  (6)   : ++++++
8  (35)  : +++++++++++++++++++++++++++++++++++
9  (4)   : ++++
10 (5)   : +++++
11 (7)   : +++++++
12 (8)   : ++++++++
13 (6)   : ++++++
14 (9)   : +++++++++
15 (3)   : +++
16 (3)   : +++
17 (0)   :
18 (12)  : ++++++++++++
19 (1)   : +
20 (3)   : +++
21 (4)   : ++++
22 (7)   : +++++++
23 (3)   : +++
24 (7)   : +++++++
25 (3)   : +++
26 (0)   :
27 (3)   : +++
28 (15)  : +++++++++++++++
29 (3)   : +++
30 (7)   : +++++++
31 (7)   : +++++++
32 (7)   : +++++++
33 (2)   : ++
34 (10)  : ++++++++++
35 (10)  : ++++++++++
36 (0)   :
37 (0)   :
38 (25)  : +++++++++++++++++++++++++
39 (1)   : +
40 (7)   : +++++++
41 (19)  : +++++++++++++++++++
42 (18)  : ++++++++++++++++++
43 (18)  : ++++++++++++++++++
44 (13)  : +++++++++++++
45 (8)   : ++++++++
46 (2)   : ++
47 (2)   : ++
48 (313) : ++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
49 (0)   :
50 (0)   :
51 (8)   : ++++++++
52 (2)   : ++
53 (1)   : +
54 (2)   : ++
55 (11)  : +++++++++++
56 (0)   :
57 (0)   :
58 (25)  : +++++++++++++++++++++++++
59 (0)   :
60 (9)   : +++++++++
61 (1)   : +
62 (2)   : ++
63 (10)  : ++++++++++
```

Output file 3 _ data2



```
46 46 1 63
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 1 1 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 1 1 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 0 1 1 1 0 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 1 1 1 0 1 1 1 1 1 0 1 1 1 1 0 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0
0 0 1 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 0 1 1 0 1 1 1 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0
0 0 0 1 1 1 1 1 1 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 1 0 1 1 1 0 1 1 1 1 1 1 0 1 1 0 0
0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 1 0 0 1 1 1 1 1 0 1 1 1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1 1 0 1 1 1 1 1 0 1 1 1 1 1 1 1 1 1 1 1 0 0 1 1 1 0 1 1 1 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 0 0 0 1 1 1 1 1 1 1 1 0 0 0 1 1 1 1 1 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 0 0 1 1 1 1 1 1 0 1 1 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 0 1 1 1 1 1 1 0 1 1 1 1 1 1 1 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 0 1 1 1 1 1 1 1 1 1 0 1 1 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 1 1 1 1 1 1 1 0 1 1 0 0 0 0 0 0 0 0 1 0 0 0 0 0
0 0 0 0 1 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 1 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 1 0 0 0 1 1 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 1 0 0 0 0 0
0 0 0 0 1 0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 1 1 0 0 0 1 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
```

Output file 4 _ data2



```
46 46 1 63
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
. . . . . 1 . . . . . . . . . . . . . . . . . . . . . . 1 . . . . . . . . . . . . . . . . .
. . . . . . . . . . 1 1 . . . . . . . . . . . 1 . . . . 1 . . . . . . . . . . . . . . . . .
. . . . . . . . . . 1 1 . . . . . . . . . . . 1 . . . . 1 . . . . . . . . . . . . . . . . .
. . . . . . . . 1 . . . . . . . . . . . . . . 1 . . . . 1 . . . . . . . . . . . . . . . . .
. . . . . . . . . 1 . . . . . . . . . . . . . 1 . . . . . 1 . . . . . . . . . . . . . . . .
. . . . . . . . . 1 . . . . . . . . . . . . . 1 . . . . . 1 . . . . . . . . . . . . . . . .
. . . . . . . . 1 . . . . . . . . . . . . . . . . . . . . . 1 . . . . 1 . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . . . 1 1 1 . . . . 1 . . . . . . . . . . . . . . . .
. . . 1 . . . . . . . . . . . . . . . . . 1 . 1 1 1 . . 1 . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . 1 . . . 1 1 1 . . 1 . . . . . . . . . . . . . . . .
. . . . . 1 1 . . . . . . . . . . . . 1 1 1 . . 1 1 1 1 . . . . . . . . . . . . . . . . . .
. . . . . 1 1 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 1 . . . . . .
. . . . . . . . . . . . . . . 1 1 1 1 1 1 1 1 1 . . . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . 1 1 1 1 . 1 1 1 . 1 1 1 1 . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . 1 1 1 1 1 1 1 1 1 1 1 1 . 1 1 . . . . . . . . . 1 . . . . .
. . . . . . . . . . . . . . . 1 1 1 1 1 1 1 1 1 . 1 1 1 1 1 1 1 . . . 1 . . . . . . . . . .
. . . . . . . . . . . . . . 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 . . . 1 . . . . . . . . . . . . .
. . . . . . . . . . . . . 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 . . 1 1 1 . . . . . . . . . . . .
. . . . . . . . . . . . 1 1 1 . 1 1 1 . 1 1 1 1 1 . 1 1 1 1 . 1 1 1 1 . . . . . . . . . . .
. . 1 . . . . . . . . 1 1 1 1 1 1 1 1 1 . 1 1 1 1 1 1 1 . 1 1 . 1 1 1 1 . . . . . . . . . .
. . . . . . . . . . 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 . . . 1 1 1 1 1 1 1 . . . . . . . .
. . . . . . . . . . . . . . . 1 1 1 1 1 1 1 1 1 . . . . . . . . . . . . . . . . . . . . . .
. . . . . . . . 1 1 1 1 1 1 . 1 1 1 1 1 1 1 1 1 1 1 1 1 . . . 1 1 1 1 1 1 1 1 . . . . . . .
. . . 1 1 1 1 1 1 . 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 . . 1 . 1 1 1 . 1 1 1 1 1 1 . 1 1 . .
. . . . . . . . 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 . 1 1 1 . . 1 1 1 1 1 . 1 1 1 . . . . . . .
. . . . . . . . . 1 1 . 1 1 1 1 1 . 1 1 1 1 1 1 1 1 1 1 1 . . 1 1 1 . 1 1 1 . . . . . . . .
. . . . . . . . . . 1 1 1 1 1 1 1 1 . . . 1 1 1 1 1 1 1 1 . . . 1 1 1 1 1 . . . . . . . . .
. . . . . . . . . . . 1 1 1 1 1 1 1 . . 1 1 1 1 1 1 . 1 1 . . . . 1 1 1 . . . . . . . . . .
. . . . . . . . . . . . 1 1 1 1 . 1 1 1 1 1 1 . 1 1 1 1 1 1 1 . . 1 1 . . . . . . . . . . .
. . . . . . . . . . . . . . . 1 1 1 1 1 1 1 1 1 . . . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . 1 1 1 1 1 . 1 1 1 1 1 1 1 1 1 . 1 1 . . 1 . . . . . . . . . . . .
. . . . . . . . . . . . . . 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . 1 1 1 1 1 1 1 1 1 1 1 1 1 1 . 1 1 . . . . . . . . 1 . . . . .
. . . . 1 1 . 1 . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . 1 1 1 . . . . . 1 . . . 1 . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . 1 1 . 1 . . . 1 1 . 1 . . . . . . . . . . . . . . . . .
. . . . . 1 . . . . . . . . . . . . . 1 1 1 1 . . 1 1 1 . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . 1 1 1 1 1 1 1 . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . . 1 1 . 1 1 . . . . . . . . . . . . . . . . . . . .
. . . . 1 . . . . . . . . . . . . . . . . . 1 1 1 . . . . . . . . . . 1 . . . . 1 . . . . .
. . . . 1 . . . . . 1 1 . . . . . . . . . . . 1 . . . . . . . . . . . . . . . . . . . . . .
. . . . . 1 1 . . . . . . . . . . . . . . . . 1 . . . . . . . . . . 1 . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . . . . 1 . . . . . . . . . 1 1 . . . 1 . . . . . . .
. . . . . . . . . 1 . . . . . . . . . . . . . 1 . . . . . . . . . . . . . . . . . . . . . .
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .
```

This is the end of the report.