ECE 2305 – Introduction to C Programming

Programming Project 07
Histograms

The file named *"numdata.txt"* contains a list of type **`double`** numbers. Write a C++ program to open the file and read the data into a 1D array. Write the program so that the program determines the number of elements that are required for the array. Use the data in the array to create two output data files. One with a sorted list of numbers from smallest value to largest value. The second output data file shall be a histogram table with the following form.

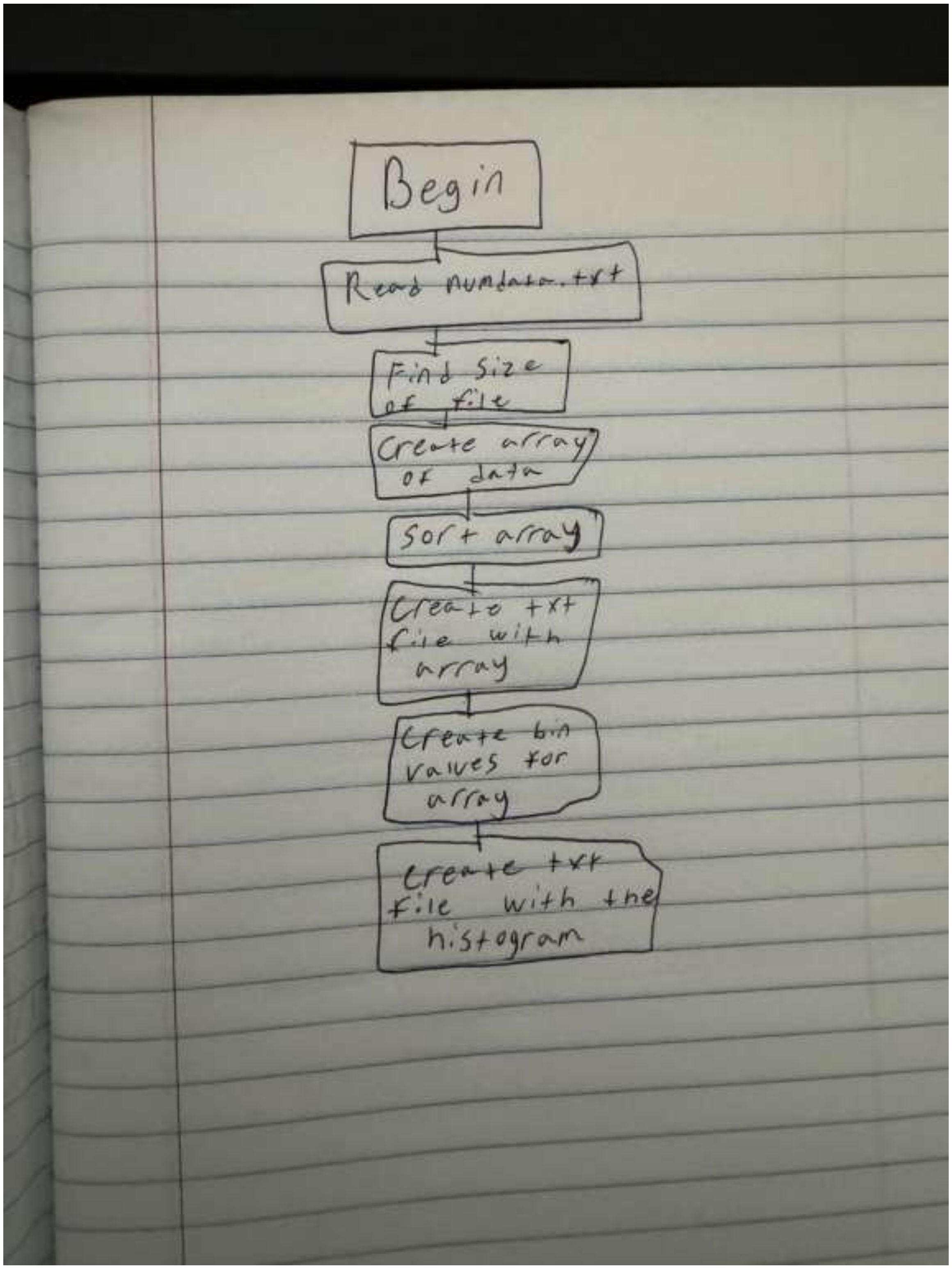
Bin	**Number of Values**
0-10	$0 \leq value < 10$
10-20	$10 \leq value < 20$
20-30	$20 \leq value < 30$
30-40	$30 \leq value < 40$
40-50	$40 \leq value < 50$
50-60	$50 \leq value < 60$
60-70	$60 \leq value < 70$
70-80	$70 \leq value < 80$
80-90	$80 \leq value < 90$
90-100	$90 \leq value < 100$

Document the program with:

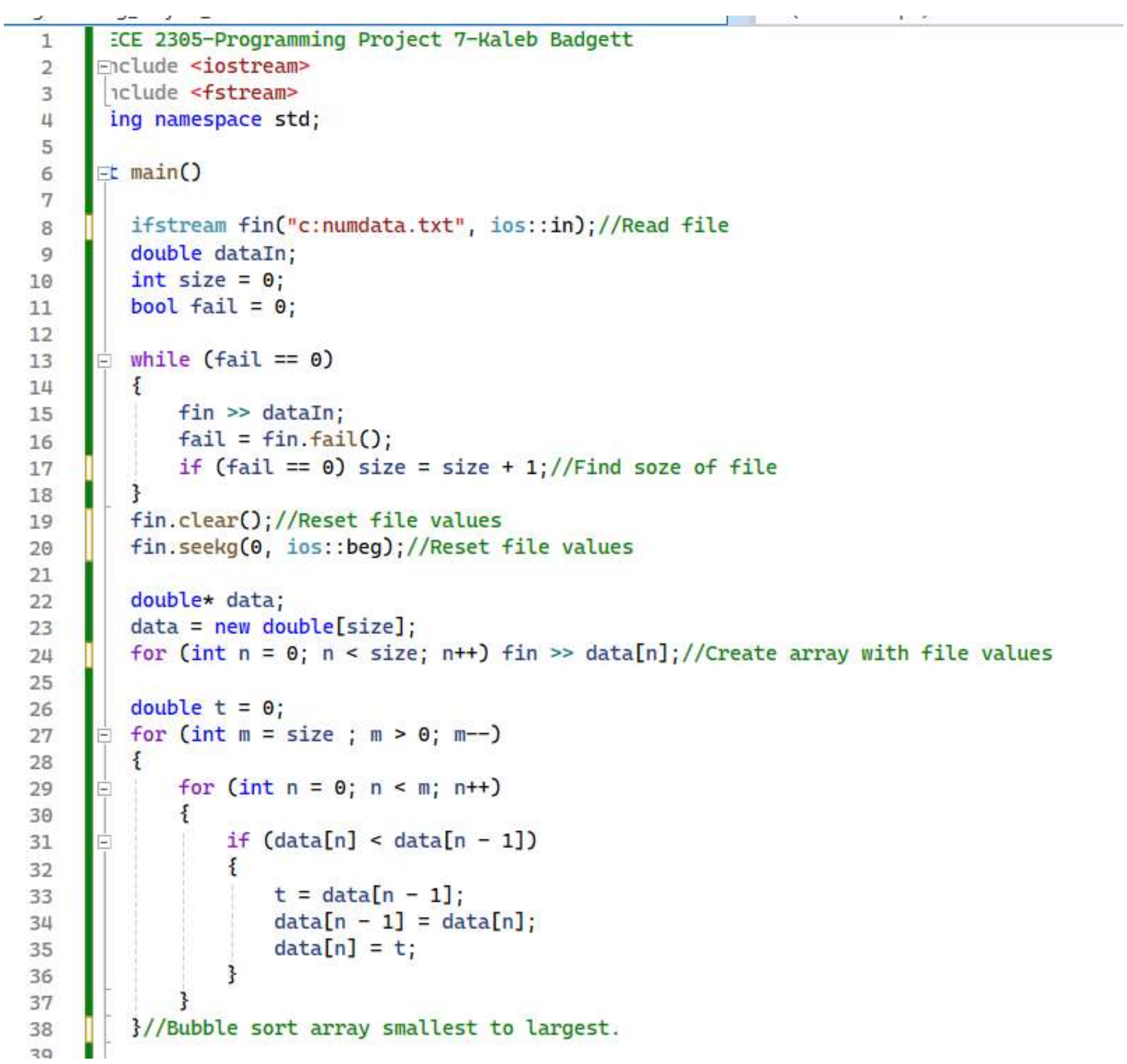
A. A brief description of the purpose of the program and the structure of the program.

The purpose of the program is to read a list of values, sort them, and create a histogram table with them. To do so, the main function begins by reading the file, then finds how many values there are. It then creates an array with those values and sorts them. Next it creates a text file with the sorted array, then a text file with the histogram file of the array.

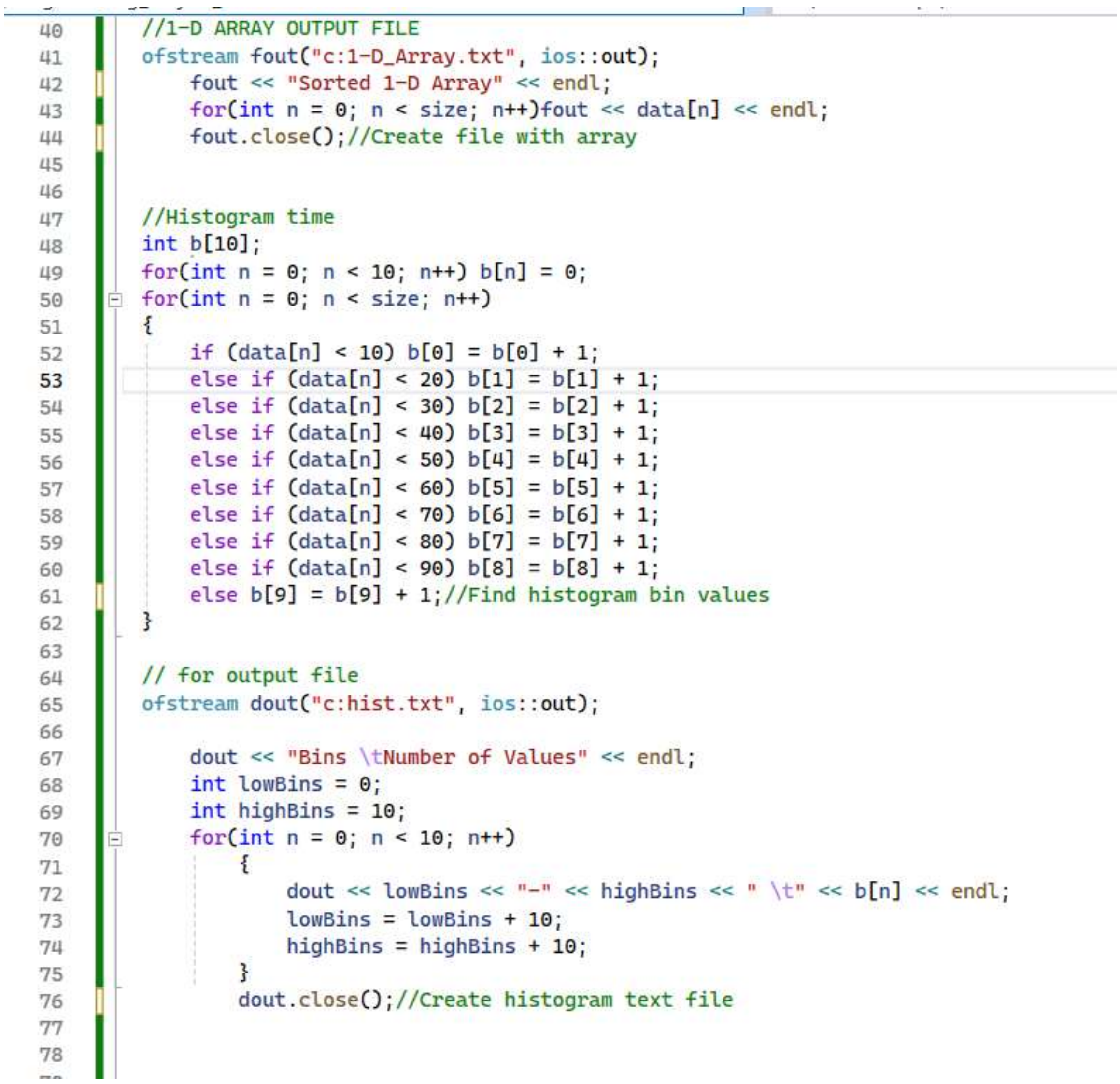
B. A flowchart to describe the structure of the program.


Begin
Read numdata.txt
Find size of file
Create array of data
Sort array
Create txt file with array
Create bin values for array
Create txt file with the histogram

C. The code listing.

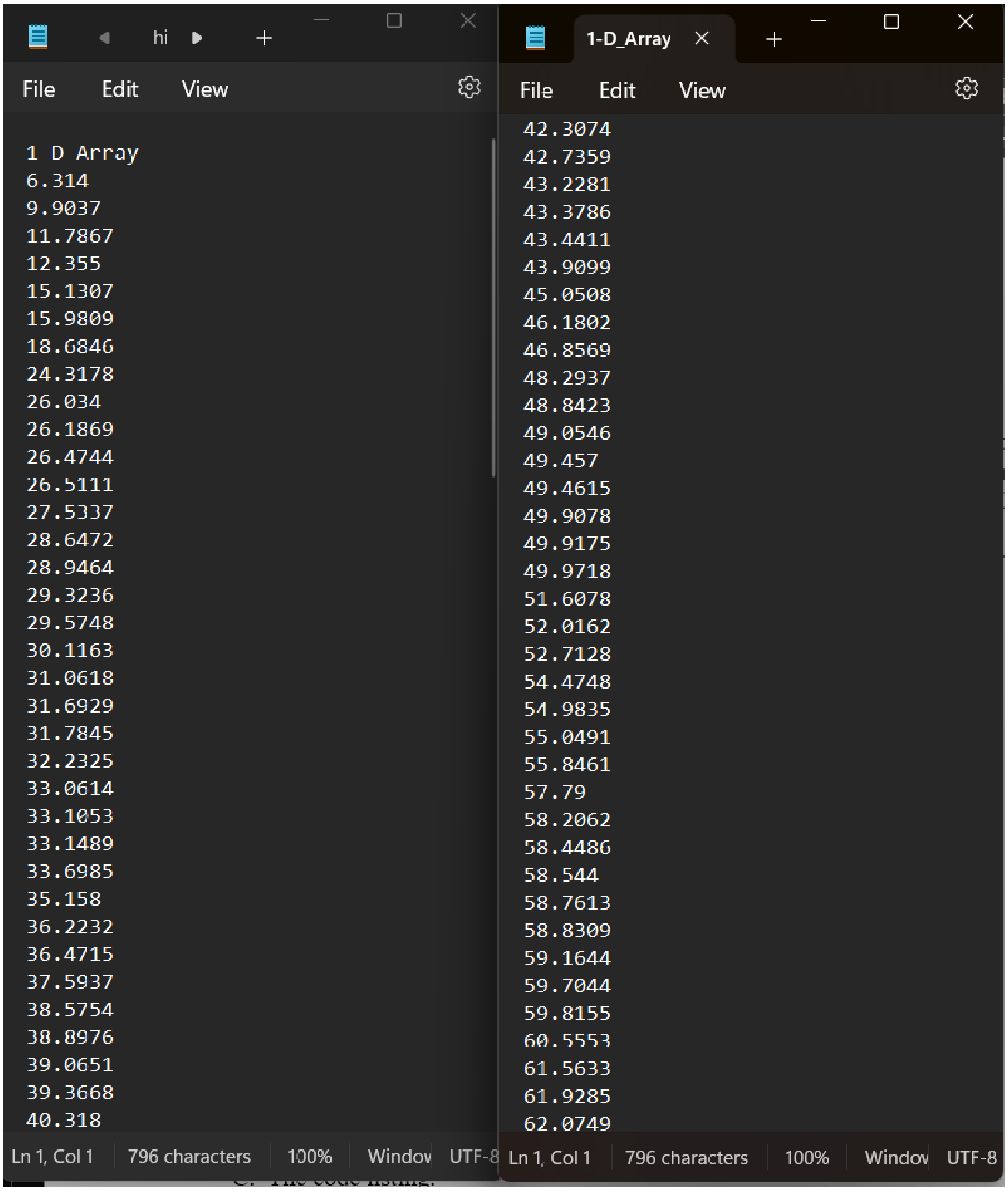


```
:CE 2305-Programming Project 7-Kaleb Badgett
nclude <iostream>
ıclude <fstream>
ing namespace std;

t main()

  ifstream fin("c:numdata.txt", ios::in);//Read file
  double dataIn;
  int size = 0;
  bool fail = 0;

  while (fail == 0)
  {
      fin >> dataIn;
      fail = fin.fail();
      if (fail == 0) size = size + 1;//Find soze of file
  }
  fin.clear();//Reset file values
  fin.seekg(0, ios::beg);//Reset file values

  double* data;
  data = new double[size];
  for (int n = 0; n < size; n++) fin >> data[n];//Create array with file values

  double t = 0;
  for (int m = size ; m > 0; m--)
  {
      for (int n = 0; n < m; n++)
      {
          if (data[n] < data[n - 1])
          {
              t = data[n - 1];
              data[n - 1] = data[n];
              data[n] = t;
          }
      }
  }//Bubble sort array smallest to largest.
```



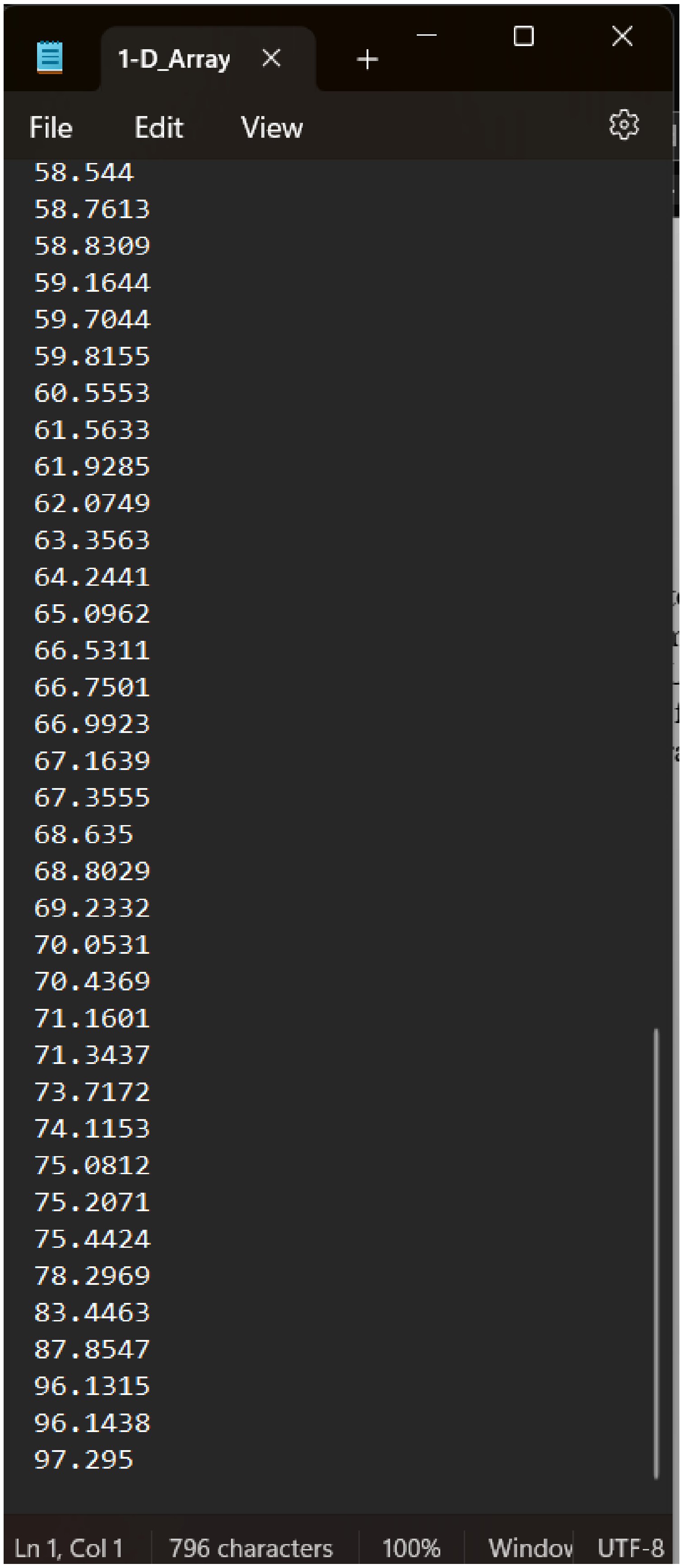
```
//1-D ARRAY OUTPUT FILE
ofstream fout("c:1-D_Array.txt", ios::out);
    fout << "Sorted 1-D Array" << endl;
    for(int n = 0; n < size; n++)fout << data[n] << endl;
    fout.close();//Create file with array


//Histogram time
int b[10];
for(int n = 0; n < 10; n++) b[n] = 0;
for(int n = 0; n < size; n++)
{
    if (data[n] < 10) b[0] = b[0] + 1;
    else if (data[n] < 20) b[1] = b[1] + 1;
    else if (data[n] < 30) b[2] = b[2] + 1;
    else if (data[n] < 40) b[3] = b[3] + 1;
    else if (data[n] < 50) b[4] = b[4] + 1;
    else if (data[n] < 60) b[5] = b[5] + 1;
    else if (data[n] < 70) b[6] = b[6] + 1;
    else if (data[n] < 80) b[7] = b[7] + 1;
    else if (data[n] < 90) b[8] = b[8] + 1;
    else b[9] = b[9] + 1;//Find histogram bin values
}

// for output file
ofstream dout("c:hist.txt", ios::out);

    dout << "Bins \tNumber of Values" << endl;
    int lowBins = 0;
    int highBins = 10;
    for(int n = 0; n < 10; n++)
        {
            dout << lowBins << "-" << highBins << " \t" << b[n] << endl;
            lowBins = lowBins + 10;
            highBins = highBins + 10;
        }
        dout.close();//Create histogram text file
```

D. Two output *.txt* data files.



```
1-D Array
6.314
9.9037
11.7867
12.355
15.1307
15.9809
18.6846
24.3178
26.034
26.1869
26.4744
26.5111
27.5337
28.6472
28.9464
29.3236
29.5748
30.1163
31.0618
31.6929
31.7845
32.2325
33.0614
33.1053
33.1489
33.6985
35.158
36.2232
36.4715
37.5937
38.5754
38.8976
39.0651
39.3668
40.318
```

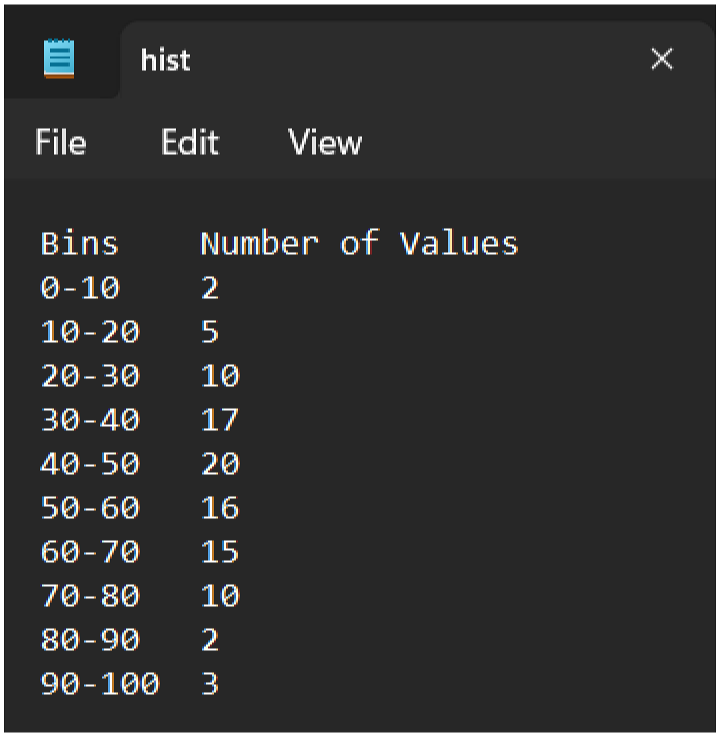
```
42.3074
42.7359
43.2281
43.3786
43.4411
43.9099
45.0508
46.1802
46.8569
48.2937
48.8423
49.0546
49.457
49.4615
49.9078
49.9175
49.9718
51.6078
52.0162
52.7128
54.4748
54.9835
55.0491
55.8461
57.79
58.2062
58.4486
58.544
58.7613
58.8309
59.1644
59.7044
59.8155
60.5553
61.5633
61.9285
62.0749
```


1-D_Array

File Edit View

```
58.544
58.7613
58.8309
59.1644
59.7044
59.8155
60.5553
61.5633
61.9285
62.0749
63.3563
64.2441
65.0962
66.5311
66.7501
66.9923
67.1639
67.3555
68.635
68.8029
69.2332
70.0531
70.4369
71.1601
71.3437
73.7172
74.1153
75.0812
75.2071
75.4424
78.2969
83.4463
87.8547
96.1315
96.1438
97.295
```

Ln 1, Col 1 | 796 characters | 100% | Windov | UTF-8



```
Bins     Number of Values
0-10     2
10-20    5
20-30    10
30-40    17
40-50    20
50-60    16
60-70    15
70-80    10
80-90    2
90-100   3
```

Upload the PDF and the TXT files to blackboard.