

ECE 71/191T – Data Structures and Algorithms

Dr. Gregory R. Kriehn, Fresno State
C++ Homework Assignment: Chapter 6

Code Due By: Midnight on Mon, Jan 30

Write-up Due By: Class on Tue, Jan 31

HOMEWORK #11 – Number Classification

You have been given a data file “Numbers.txt” and need to classify the following information:

1. The total number of values in the data file.
2. The total number of even values.
3. The total number of odd values.
4. The total number of zeros.
5. The sum of the numbers.
6. The average of the numbers.

The numbers read in from the data file should be printed to the screen, as well as written to a data file called “Output.txt” – 10 numbers per line. After classifying the information in the data file, the results should also be written to the text file.

Specifications:

Write a program that attempts to open the data file and prints out a warning if the file cannot be opened. If the data file contains an unknown number data points. As data is read in from the file, the positive and negative numbers should be classified as odd, even, etc. Any numbers printed to the screen or to the output data file should be printed 10 numbers per line using a width of 5 spaces per number. If there are no numbers in the data file, print out an appropriate warning and end the program. You may assume all values, except the average, are integers. The average should be printed to the nearest tenth.

To help you solve the problem, the program must be broken into a number of functions with the following function prototypes:

```
// Initialize all input variables to 0
void initialize (int &totalCount, int &evenCount,
               int &oddCount, int &zeroCount, int &totalSum,
               double &totalAverage);

// Get the current number from the data file
// Print the number to the screen and the data file
void getNumber(int &num, ifstream &inputFile,
              ofstream &outputFile);

// Classify the number as even, odd, and/or zero.
```

```

// Keep track of the tally for each number.
void classifyNumber (int num, int &evenCount, int &oddCount,
    int &zeroCount);

// Print the results of the classification of the numbers
// to the output data file.
void printResults(int totalCount, int evenCount,
    int oddCount, int zeroCount, int totalSum,
    double totalAverage, ofstream &outputFile);

```

Notice that a number of the formal parameters for the function prototypes contain references. If there is no input data file present:

```
Cannot open input file 'Numbers.txt'. Program Terminating.
```

If the data file is present, but there is no valid data in the data file:

```

Processing Data...
Numbers Read from File

No valid input. Please check input file.

```

If there is valid data in the file:

```

Processing Data...
Numbers Read from File

  43   67   82    0   35   28  -64    7  -87    0
   0    0    0   12   23   45    7   -2   -8   -3
  -9    4    0    1    0   -7   23  -24    0    0
  12   62  100  101 -203 -340  500    0   23    0
  54    0   76
~> more Output.txt
  43   67   82    0   35   28  -64    7  -87    0
   0    0    0   12   23   45    7   -2   -8   -3
  -9    4    0    1    0   -7   23  -24    0    0
  12   62  100  101 -203 -340  500    0   23    0
  54    0   76

```

```

Number of values in data file: 43
Total number of evens: 27
Total number of odds: 16
Total number of zeros: 12
Sum of the numbers: 558
Average of numbers: 13.0

```

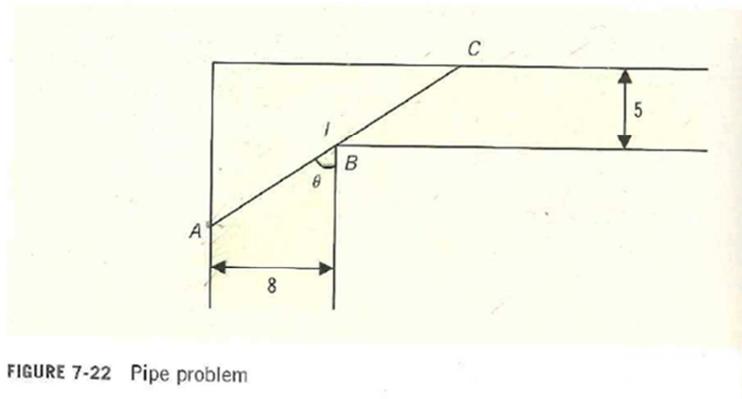
Once you verify the operation of your program, submit your source code to the Grader Program.

HOMEWORK #12 – Hallway Pipe Problem

A pipe is to be carried around the right-angled corner of two intersecting corridors. Suppose that the widths of the intersecting hallways are measured in feet. In the example shown below, the widths happen to be 8 feet and 5 feet, respectively. Your objective is to find the length of the longest pipe, floored to the nearest foot, that can be carried level around the right-angled corridor.

Specifications:

Write a program that prompts the user to input the widths of both hallways. The vertical hallway is considered to be the first hallway, and the horizontal hallway the second hallway. The program should determine critical angle that constrains the length of the pipe that can be carried around the corner, and print the result in degrees using two values after the decimal point. The length of the pipe should be floored to the nearest foot and printed as well.



The length of the longest pipe should be calculated using a function with the following function prototype:

```
// Calculates the length of longest pipe
void longestPipe(double wHallway1, double wHallway2,
                 double &mTheta, double &mLength);
```

The function returns the maximum theta and length values. Assume $0 < \theta < \pi/2$, and that the incremental change in θ is 0.01 rad.

HINT: The length of the pipe for any given angle can be calculated by determining the length $l = AB + BC$.

```
Enter the length of the first hallway: 13.5
Enter the length of the second hallway: 9.0
```

```
The length constraint occurs at Theta: 48.70 deg.
Length of longest pipe is: 31 ft.
```

Once you verify the operation of your program, submit your source code to the Grader Program.