Introduction to C - Programming Assignment #4
Fuel Graph

Due date: *October 27, 2010, 11:59pm*

Note: In this assignment, you are required to write ONE program.

Objectives

1. Learn how to read information from files.
2. Learn how to store and use information in arrays.

References

CKnights: Sections 9, 10

Problem: Fuel Efficiency Display

The current technology prepares a nice graphical display of a car's fuel efficiency during a trip. In this program, you will do something comparable to the Toyota Prius fuel efficiency display. Since your computer can not directly get gas and tire rotation data from a car, your program will simulate doing so by reading in this data from a file.

In particular, your program will ask the user to enter the file storing the tire gas and tire rotation data. Then, your program will make the necessary calculations and display the fuel efficiency of the car in five minute increments.

Input File Format

The input file will be guaranteed to have the following file format:

The first line of the input file will contain two integers separated by spaces. The first integer, *n,* will represent the number of minutes in the trip. *n* is guaranteed to be a multiple of 5 that is no larger than 30. The second integer, *r,* will represent the radius of the tire of the car, in inches. *r* is guaranteed to be a positive integer.

The rest of the file will contain *60n* lines of data. Each line of data contains two values, representing the car's information for a single second of time. The first value on each of these lines will be a double representing the number of revolutions the tires made in that second. The second value of each of these lines will be a double representing the amount of gas in gallons that was used in that second. **Note: It is possible for either of these values to be 0. If no fuel is used for a whole five minute interval, and the car has traveled a positive distance in that interval, then we will print out that the car's fuel efficiency was 100 miles per gallon. If the car has not used any fuel AND has not moved either in a five minute interval, then its fuel efficiency will be 0 miles per gallon.**

Here is a portion consisting the first 11 lines of a valid input file:

```
10 13
1.131232 0.000183
2.872323 0.000312
4.233291 0.000422
5.823823 0.000367
6.464139 0.000316
6.554324 0.000275
6.001023 0.000245
5.293232 0.0
7.012323 0.000612
7.013323 0.000583
```

This file stores a trip that lasted 10 minutes long for a car with a tire radius of 13 inches. For the first second of the trip, the car averaged 5.25 miles an hour. In the second second of the trip, the car averaged 13.33 miles an hour. Furthermore, in this second second, the car used .000312 gallons of gas, for a fuel efficiency of 11.87 miles per gallon.

<u>Output Format</u>

Your output (except for the bottom header) will consist of 20 rows, labeled, 95, 90, 85, ..., 0, corresponding to fuel efficiencies (in mpg). These labels should take up exactly four spaces and be left-justified. An integer variable can be printed in this manner as follows:

```
printf("%-4d", variable);
```

Following the header for each row will be "bars" for each of the time intervals, going left to right in chronological order. Each "bar" will be exactly 5 characters wide. Each bar will be separated by a single space. The bottom header will be as follows:

```
    -----------------------------------
    00-05 05-10 10-15 15-20 20-25 25-30
```

Note that there are exactly four spaces at the beginning of these two lines. Here is an example of output for the posted sample:

```
95
90
85
80
75
70
65
60
55
50
45
40
35
30        *****
25        ***** *****
20        ***** *****
15        ***** *****
10  ***** ***** *****
5   ***** ***** *****
0   ***** ***** *****
    -----------------------------------
    00-05 05-10 10-15 15-20 20-25 25-30
```

This output corresponds to a trip where in the first five minutes the car averaged in between 10 and 15 miles/gallon. In the second five minutes the car averaged in between 30 and 35 miles/gallon and in the last five minutes, the car averaged in between 25 and 30 miles/gallon.

Implementation Requirements

You must use an array to store the average fuel efficiency for each five minute increment of the trip. This means that the array to be used should be of size 6, since there will be at most 6 five minute intervals for any trip.

Output Sample

A sample input file and corresponding output will be posted in separate files on the course web page.

Deliverables

One source file:

1) *fuelgraph.c*, for your solution

This file is to be submitted over WebCourses.

Restrictions

Although you may use other compilers, your program must compile and run using Dev C++. Each of your three programs should include a header comment with the following information: your name, course number, section number, assignment title, and date. Also, make sure you include comments throughout your code describing the major steps in solving the problem.

Grading Details

Your programs will be graded upon the following criteria:

1) Your correctness
2) Your programming style and use of white space. Even if you have a plan and your program works perfectly, if your programming style is poor or your use of white space is poor, you could get 10% or 15% deducted from your grade.
3) Compatibility to Dev C++ (in Windows). If your program does not compile in this environment, **the maximum credit you will receive is 50%.**