Simple Input/Output and Algebraic Calculations

One of the first things you will learn about C++ is how to perform numerical computations. In this project, you will design and implement a program that will read simple input, perform simple calculations, and write simple output.

The input file:

The input file for this program is named `"TripData.txt"`. A sample input file is given below. The input file begins with two header lines which are meaningful to a human reader but must be ignored by the program. The header is followed by one line of data pertaining to a particular trip that has been taken. This line specifies the name of the place where the trip began, the name of the destination where the trip ended, the number of miles traveled, and the time required for the trip.

```
Origin              Destination            Miles     Time
-------------------------------------------------------
Pecos, TX           Carslbad, NM              75     1:07
```

The names are strings of characters, which may contain spaces; in order to make the data easier to read in, a single tab character follows each name. For the sake of formatting output, the names are guaranteed to be no more than 19 characters long. A single tab character also separates the mile and time values, and there is a newline character immediately after the time. Note that the appearance of tabs is somewhat unreliable. In the sample above the tabs are displayed according to the settings in Microsoft Word and everything is neatly aligned. Viewed in a text editor, such as the Visual C++ IDE, a tab is usually interpreted as having a fixed width (such as 4 spaces) and the alignment will not appear to be nearly as nice. The visual appearance of the data doesn't matter to the program that will read it, but the presence of tabs will make it easier to write code to read that data.

You also may assume that the given data will always be logically valid, and that the input file will always conform to the format specification given here.

What must be calculated?

For the given trip, the program must calculate the average speed in miles per hour, MPH, and the time required for the trip in minutes.

Warning: the necessary calculations are relatively simple, but they may require some thought. It would be a violation of the Honor Code to explain to another student how to do the calculations or to post that information to a course listserv, discussion board, or newsgroup.

In order to produce the most accurate results possible in C++, all the mileage and time values should be stored as integers, not as decimal numbers, and the average speed should be stored using variables of type `double`, not type `float`.

The output file:

The output file is named `"Summary.txt"`. An output file, which corresponds to the given input file, is shown below. The first two lines specify the programmer and the assignment, and the third is blank. The fourth line specifies column headers for the results the program will write, and the fifth line separates the table data from the labels.

There is one line of output for the given trip, displaying the trip origin and destination, the trip mileage, the length of the trip in minutes, and average speed for that trip. This is followed by another delimiter line.

When a decimal number is written, the number of digits displayed after the decimal point is called the precision of the displayed value. The average speed for each must be written with precision 1.

```
Programmer: Bill McQuain
CS 1044 Notes Example

Origin              Destination         Mileage   Minutes     MPH
-------------------------------------------------------------------
Pecos, TX           Carslbad, NM             75        67    67.2
-------------------------------------------------------------------
```

If you have read the description of how the Curator scores your program in the *Student Guide*, you know that is important that you use the same spelling and capitalization for all the labels shown above. The horizontal spacing does not effect scoring unless you combine things that should be separate or separate things that should be combined. In any case, the given code will exactly match the labels and spacing shown above.