Background:

The local Kroger has self check-out lanes. The key to the self check-out is quick retrieval of information about a product, based on the products, UPC. This project will simulate reading one products UPC and computing the cost, including tax, of that product.

Details:

For this project, you will read information from a file that contains information about a particular product. You will need to read the products UPC number, products unit price, the number of units purchased. You will compute the total purchase price, including tax, and print a summary statement into another file.

The input file will be called "Stock.in" and will have an unknown number of data line in the following format: First line will be a header line. The remaining lines will be the product's name, product's unit price, product's price if the purchaser's has a Kroger card, number of units purchased, whether or not they have a Kroger card, and the purchaser's id number. If the purchaser has a Kroger card, then the id number will follow, otherwise it will not. Each item on the line will be separated by a tab character. An example follows:

```
UPC             Prd Name          UnitPrc     Units Krgr Prc    Krgr Crd    Id Num
012-34230-6574  Chocolate Milk    $1.54       13    $1.04       Yes         X3-42F9
123-24524-6543  Shampoo           $5.32       2     $2.34       No
532-43576-7862  Dog Bones         $3.67       4     $3.01       Yes         52-AG52-RR
```

The output file will be called "Stock.out" and will have the following format.
The first line will be a header line indicating what each column is. The following lines will give the will the information for each item read in along with the subtotal, tax and total. After all the lines for each item, there will be a summary section for the entire amount of the purchases, including the total units purchased, the amount saved and the percent saved. The format given below must be followed exactly. An example follows:

```
UPC            Product Name   Price   Units   Kroger Price   Kroger Card    Id Num         Subtotal       Tax     Total
-------------------------------------------------------------------------------------------------------------------------
012-34230-6574 Chocolate Milk $1.54   13      $1.04          Yes            X3-42F9        13.52          0.54    14.06
123-24524-6543 Shampoo        $5.32   2       $2.34          No                            10.64          0.42    11.06
532-43576-7862 Dog Bones      $3.67   4       $3.01          Yes            52-AG52-RR     12.04          0.48    12.52
-------------------------------------------------------------------------------------------------------------------------
                                                                                        36.20          1.44    37.64

Total Units Purchased: 3
Amount Saved:  $9.50
Percent Saved: 20%
```

Programming Considerations:

The tax rate will be constant throughout the program and is set to be 4%.

When programming the solution to this problem, you want to use the stream functions discussed in class. Decide the most efficient way to read or parse the input file and produce the output file.

- You are **not** to use doubles or floating point data types at all for any monetary amount in this project. The tax rate is probably best implemented as a double.
- You are only to use what we have discussed so far in class.
- You are not allowed to use any user defined functions.
- You are not to use any other code or ideas that we haven't discussed.
- You should have a header comment identifying yourself, and describing what the program does.
- Every constant and variable you declare should have a comment explaining its logical significance in the program.
- Every major block of code should have a comment describing its purpose.
- Adopt a consistent indentation style and stick to it.
- Your implementation must also meet the following requirements:
  - Choose descriptive identifiers when you declare a variable or constant. Avoid choosing identifiers that are entirely lower-case.
  - Use C++ streams for input and output, not C-style constructs.
  - Use C++ string variables to hold character data, not C-style character pointers or arrays.
  - Note: you are explicitly forbidden to write any user-defined functions for this program. This will make the program somewhat repetitive, and physically longer than the alternative. To some extent, that's the reason for this restriction.

Submitting your project

You will submit this assignment to the Curator System (read the *Student Guid*e), and it will be graded automatically. Instructions for submitting, and a description of how the grading is done, are contained in the *Student Guid*e.

You will be allowed up to five submissions for this assignment. Use them wisely. Test your program thoroughly before submitting it. Make sure that your program produces correct results for every sample input file posted on the course website. If you do not get a perfect score, analyze the problem carefully and test your fix with the input file returned as part of the Curator e-mail message, before submitting again. The highest score you achieve will be counted.

The *Student Guide* and submission link can be found at: http://www.cs.vt.edu/curator/

Pledge

Each of your program submissions must be pledged to conform to the Honor Code requirements for this course. Specifically, you **must** include the following pledge statement in the header comment for your program:

```
// On my honor:
//
// - I have not discussed the C++ language code in my program with
//   anyone other than my instructor or the teaching assistants
//   assigned to this course.
//
// - I have not used C++ language code obtained from another student,
//   or any other unauthorized source, either modified or unmodified.
//
```

```
// - If any C++ language code or documentation used in my program
//   was obtained from another source, such as a text book or course
//   notes, that has been clearly noted with a proper citation in
//   the comments of my program.
//
// - I have not designed this program in such a way as to defeat or
//   interfere with the normal operation of the Curator System.
//
// <Student Name>
```

Failure to include this pledge in a submission is a violation of the Honor Code.