COP3014-Foundations of Computer Science Assignment #6

You will implement a program called **"nursery_inv3.cpp"** to process customer purchase orders (orders) for a nursery. You will read the data stored in a data file into a static array of purchase orders records, then process each purchase order record in the array, and finally print the array of purchase order records to a datafile. The purchase orders will be stored in order records. Each order record contains nine fields, which are as follows: 1) a plant name (string, no spaces), 2) a county name (string, no space), 3) the cost of the plant (double), 4) the quantity of plants in the purchase (integer), 5) the net cost of the purchase (double), 6) tax rate (double), 7) the tax on the purchase (double), 8) the discount on the purchase (double), and 9) the total cost of the purchase. **Your program will have 4 functions: input, process, output, and count_inventory. Your main program will call (invoke) input, process, and output until the end of the data file has been reached. Once the program is finished reading, it will then call count_inventory. Following are the descriptions of the functionality of each function:**

Following are the descriptions of the functionality of each function:

1. The void function "**input**" will have two parameters: the order record array called '**STR**", and "**count**". The **capacity** (SIZE) of STR **should be initialized to 100**, and "count" should be initialized to 0. The input data file will be opened and closed inside this function. The function will read all the data until the end-of-file has been reached; the pname (plant name), cname (county name), the plant_cost (the cost of the plant), and the quantity (number of plants in purchase order), will be read into the array of order_records (STR) from the data file. Remember, "count" should be passed by reference.
2. The function "**process**" will have two parameters: the order record array called '**STR**", and "**count**". Process will determine the ***net cost*** of the purchase (***net_cost***), the tax rate (***tax_rate***), the ***tax*** on the purchase (***purchase_tax***), the ***discount*** on the purchase (***discount***), and the ***total cost*** of the purchase (***total_cost***). Please consider the following information to help you implement the necessary calculations:
    a. The ***tax rate (in percent) on a purchase*** is based on the ***county*** where the purchase was made. If the county was ***dade***, the tax rate is 6.5%; if the county is ***broward***, the tax rate is 6%; if the county was **palm**, the tax rate is 7%.
    b. The ***net cost of a purchase*** is calculated by the following formula:
    net_cost = (quantity x plant_cost)
    c. The ***discount*** is based on the ***quantity of plants*** in the purchase. ***The discount is determined is follows:***
        - If ***quantity*** equals 0, then the ***discount*** is 0% of the net cost;
        - If 1<= ***quantity*** <=5 then ***discount*** = 1% of the net cost; 6<= ***quantity*** <=11 then ***discount*** = 3% of the net cost ; if 12<= ***quantity*** <=20 then ***discount*** = 5% of the net cost; 21<= ***quantity*** <=50 then ***discount*** = 8% of the net cost; ***quantity*** >50 then ***discount*** =12% of the net cost) . Apply discount after the net cost has been calculated.
    d. The ***tax on a purchase*** is calculated by the following formula:
        - ***purchase_tax = (net_cost * tax_rate / 100*** *(drop /100 if you converted the rate from a percentage)*

e. The ***total cost of a purchase*** *(rounded to the nearest hundredth)* is calculated by the following formula:

total_cost = net_cost +purchase_tax - discount.

Note: All tax and cost calculations should be rounded to the nearest hundredths.

3. The function"**output**" will have two parameters: the order record array called '**STR**", and "**count**". "output" will print every field of an order record stored in the array **STR** to the file, "***nursery_run_results.txt***". The output data file will be opened and closed inside this function. The function will print every field of every order record stored in **STR**. The fields should be printed in the following order: plant name, county name, plant cost, quantity of plants, net cost of purchase, tax rate, purchase tax, discount, and total cost. See the sections below called "Input Stream" and "**Format of Output**" for more information. See the section "**Format of Input Data File (input filename is "nursery_stock.txt")**".

4. The double function "**count_inventory**" which will calculate the average of the "**total order cost**". The average should be rounded to the nearest hundredth and returned to main. "**count_inventory**" will be the last function your main program calls. **The main program will print the following message after the function output has executed.** (Note: the average appears as XXXXXXX.XX to represent a double values.).

Average Total Order Cost = XXXXXX.XX

You may implement more functions if you find it necessary. Please start the assignment ASAP, and ask questions to make sure you understand what you must do. It is always good to start with the skeleton program I provide. Remember to follow all style rules and to include all necessary documentation (consistent, indentation, proper variable names, pre/post conditions, program header, function headers, and so forth.) .

Output Format for the Function "output":

1. Use the following format information to print the variables:

Field	Format
Plant Name	**string**
County Name	**string**
Plant Cost	**XXXX.XX**
Quantity of Plants	**XXXX**
Net Cost of Purchase	**XXXXX.XX**
Tax Rate	**X.XXX**
Purchase Tax	**XXXXX.XX**
Discount on Purchase	**XXXX.XX**
Total Cost of Purchase	**XXXXXXX.XX**

2. Consider the following sample output table when designing and implementing the function "**output**":

(**The output is in the following order: plant name, county name, plant cost, quantity, net cost, tax rate, purchase tax, discount, total cost**).

```
owl        dade       10.55     100     1055    0.065    68.58    126.6   996.98
hibiscus   broward    15.82      15    237.3     0.06    14.24    11.87   239.67
rose       dade        9.99      45   449.55    0.065    29.22    35.96   442.81
carnation  palm        7.99      32   255.68     0.07     17.9    20.45   253.12
```

3. **Input Stream**
   In the assignment you will declare one ifstream to bind your input to the file "***nursery_stock.txt***" to an input file stream. Whenever a program performs file i/o you must include the "***fstream***" library. **Add the following statements to your program:**

   For source file, "***nursery_inv3.cpp***"

   - Add "#include <fstream>" to your # include statements in your source file.
   - Add "#include <string>" to your # include statement in your source file.
   - Add "#include <iomanip>" all formatting of output

4. **Format of the input data file (input filename is "nursery_stock.txt"):**

   Do not include column titles in your data file. Field as order as follows: plant name, county name, plant cost, quantity.

```
Owl          dade      10.55   100
Hibiscus     broward   15.82    15
Rose         dade       9.99    45
carnation    palm       7.99    32
Rose         palm       7.99    60
Widow        palm      25.75     5
carnation    dade      12.55    10
carnation    dade      12.55     8
Lilly        broward    6.92   150
xerabtgemum  palm      13.63    50
Yarrow       dade      22.85    20
Zenobia      palm      37.19    32
zephyranthes broward   62.82    40
Daisy        broward   15.99    80
aconitum     dade      30.02    72
amaryllis    dade      16.14    65
Begonia      broward   18.45     3
Bellflow     broward    2.96   200
bergenia     palm      85.92    10
```

Format of Output:
(plant name, county name, plant cost, quantity, net cost, tax rate, purchase tax, discount, total cost)

Using the information in the data file “***nursery_stock.txt***” you should print the following output to the file, “***nursery_run_results.txt***”. Your output should not contain any titles, but the output must be in the proper order as stated in the assignment. The following contains an example of the output your program should produce if you use the contents in the data file “***nursery_stock.txt***”. You should double check the results.

```
owl            dade       10.55   100     1055   0.065    68.58    126.6    996.98
hibiscus       broward    15.82    15    237.3    0.06    14.24    11.87    239.67
rose           dade        9.99    45   449.55   0.065    29.22    35.96    442.81
carnation      palm        7.99    32   255.68    0.07     17.9    20.45    253.12
rose           palm        7.99    60    479.4    0.07    33.56    57.53    455.43
widow          palm       25.75     5   128.75    0.07     9.01     1.29    136.47
carnation      dade       12.55    10    125.5   0.065     8.16     3.76    129.89
carnation      dade       12.55     8    100.4   0.065     6.53     3.01    103.91
lilly          broward     6.92   150     1038    0.06    62.28   124.56    975.72
xerabtgemum    palm       13.63    50    681.5    0.07    47.71    54.52    674.69
yarrow         dade       22.85    20      457   0.065    29.71    22.85    463.85
zenobia        palm       37.19    32  1190.08    0.07    83.31    95.21   1178.18
zephyranthes   broward    62.82    40   2512.8    0.06   150.77   201.02   2462.54
daisy          broward    15.99    80   1279.2    0.06    76.75    153.5   1202.45
aconitum       dade       30.02    72  2161.44   0.065   140.49   259.37   2042.56
amaryllis      dade       16.14    65   1049.1   0.065    68.19   125.89     991.4
bogonia        broward    18.45     3    55.35    0.06     3.32     0.55     58.12
bellflow       broward     2.96   200      592    0.06    35.52    71.04    556.48
bergenia       palm       85.92    10    859.2    0.07    60.14    25.78    893.57
```