

In this assignment you will implement a program called *"nursery_inv3.cpp"* to process the customer purchase orders for a nursery. The purchase orders will be stored in order records. Each order record contains ten 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 tax on the purchase (double), 6) the net cost of the purchase (double), 7) the discount rate (double), 8) discount on the purchase (double), 9) the purchase tax (double), and 10) the total cost of the purchase (double). Your program will have 3 functions: input, process, and output. Your main program will call (invoke) each function until the end of the data file has been reached. Following are the descriptions of the functionality of each function:

1. The void function **"input"** will have two parameters: an input file stream called **"in"**, and a customer purchase order record called **"order_record"**. The function will read a plant name (**pname**), the name of the county the plant was purchase in (**cname**), the cost of the plant (**plant_cost**), and the quantity of the plant purchase (**quantity**) into an order record from the data file, *"nursery_stock.txt"*.
2. The void function **"process"** will calculate the **tax rate** (**tax_rate**), the **net cost** of the purchase (**net_cost**), the **discount rate** (**discount_rate**), the **discount** on the purchase (**discount**), the **tax** on the purchase (**purchase_tax**), 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 5.5%; if the county is **broward**, the tax rate is 5%; if the county was **palm**, the tax rate is 6%.
 - b. The **net cost of a purchase** is calculated by the following formula:
 - **$net_cost = (quantity \times plant_cost)$**
 - c. The **discount** is based on the **quantity of plants** in the purchase. **The discount is determined as follows:**
 - If **quantity** equals 0, then the **discount** is 0% of the net cost;
 - If $1 \leq quantity \leq 6$ then **discount** = 2% of the net cost; $7 \leq quantity \leq 13$ then **discount** = 4% of the net cost ; if $14 \leq quantity \leq 25$ then **discount** = 7% of the net cost; $26 \leq quantity \leq 60$ then **discount** = 9% of the net cost; **quantity** > 60 then **discount** = 14% 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 void function “**output**” will print every field of an order record to the screen. The fields should be printed in the following order: plant name, county name, plant cost, quantity of plants, tax rate, net cost of purchase, discount rate, discount on purchase, purchase tax, and total cost. See the section below called “**Input Stream**” and “**Format of Output**” for more information. See “**Format of Input data file, “nursery_stock.txt”**”.

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
Tax Rate	XX.XX
Net Cost of Purchase	XXXXX.XX
Discount Rate	XX.XX
Discount on Purchase	XXXX.XX
Purchase Tax	XXXX.XX
Total Purchase	XXXXXXXX.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.) .

Finally, your input data file (*nursery_stock.txt*) should be in the same directory as your program source file (nursery_inv3.cpp) for your program to produce the correct output.

Output Format for the Function “output”:

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, tax rate, net cost, discount rate, discount, purchase tax, total cost).

owl	dade	10.55	100	5.50	1055.00	14.00	147.70	58.02	965.33
hibiscus	broward	15.82	15	5.00	237.30	7.00	16.61	11.87	232.55
rose	dade	9.99	45	5.50	449.55	9.00	40.46	24.73	433.82
carnation	palm	7.99	32	6.00	255.68	9.00	23.01	15.34	248.01
rose	palm	7.99	60	6.00	479.40	9.00	43.15	28.76	465.02
widow	palm	25.75	5	6.00	128.75	2.00	2.58	7.72	133.90
carnation	dade	12.55	10	5.50	125.50	4.00	5.02	6.90	127.38
carnation	dade	12.55	8	5.50	100.40	4.00	4.02	5.52	101.91
lilly	broward	6.92	150	5.00	1038.00	14.00	145.32	51.90	944.58
xerabtgum	palm	13.63	50	6.00	681.50	9.00	61.34	40.89	661.05
yarrow	dade	22.85	20	5.50	457.00	7.00	31.99	25.14	450.14
zenobia	palm	37.19	32	6.00	1190.08	9.00	107.11	71.40	1154.38
zephyranthes	broward	62.82	40	5.00	2512.80	9.00	226.15	125.64	2412.29
daisy	broward	15.99	80	5.00	1279.20	14.00	179.09	63.96	1164.07
aconitum	dade	30.02	72	5.50	2161.44	14.00	302.60	118.88	1977.72
amaryllis	dade	16.14	65	5.50	1049.10	14.00	146.87	57.70	959.93
bogonia	broward	18.45	3	5.00	55.35	2.00	1.11	2.77	57.01
bellflow	broward	2.96	200	5.00	592.00	14.00	82.88	29.60	538.72
bergenia	palm	85.92	10	6.00	859.20	4.00	34.37	51.55	876.38

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_inv2.cpp*”

- Add “#include <iostream>” to your #include statements in your source file.
- 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

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

Use the following skeleton to help you start this program which is included as a cpp file in the module: nursery_inv3.cpp

```
//THIS IS ONLY AN EXAMPLE TO HELP YOU GET STARTED WITH THIS ASSIGNMENT
```

```
//EXAMPLE OF PROGRAM HEADER
```

```
/******  
*****
```

```
Name:                               Z#:  
Course: Foundations of Computer Science (COP3014)  
Professor: Dr. Lofton Bullard  
Due Date:                           Total Points: 20  
Assignment#:Assignment#3 (nursery_inv3.cpp)
```

```
Description:
```

```
*****  
*****/
```

```
//Include the following
```

```
#include <iostream>  
#include <string>  
#include <fstream> //you must include this library if you wish to do file i/o  
#include <iomanip>  
using namespace std;
```

```
/******
```

```
//Following is the declaration of a order record
```

```
*****/
```

```
class order_record  
{  
public:  
    string pname;  
    string cname;  
    double plant_cost;  
    int quantity;  
    double tax_rate;  
    double net_cost;  
    double discount_rate;  
    double discount;  
    double purchase_tax;  
    double total_cost;  
};
```

```
//Prototypes for your functions: input, output, and process will go here
```

```
void input(ifstream&, order_record&);  
void output(ofstream &, const order_record&);  
void process(order_record&);
```

```
//Function Implementations will go here
```

```
/******
```

```
//Name: input
```

```
//Precondition: State what is true before the function is called.
```

```

// Example: the variables (formal parameters) have not been initialized
//Postcondition: State what is true after the function has executed.
// Example: the variables (formal parameters) have been initialized
//Description:
// Example: Get input (values of plant name, county name, plant cost, quantity )
//from data file order record (p_record).

//PURPOSE: SHOW ME THAT YOU KNOW HOW TO READ INPUT AND USE INPUT (CALL BY VALUE) & OUTPUT (CALL BY
VALUE) PARAMETERS

//*****

void input(ifstream& in, order_record& plant_record) // example using the call by reference mechanism
in C++ -- call record is passed by reference --note & operator
{
    in >> plant_record.pname;
    //the rest of your code goes here....
}

//*****
//Name: process
//Precondition: The state what is true before the function is called.
//Postcondition: State what is true after the function has executed.
//Description: Describe what the function does (purpose).
//*****

//Note: there is one 1 input/output parameter
void process(order_record & plant_record)
{
    //put your code here for your calculations
}

//*****
//Name: output
//Precondition: State what is true before the function is called.
//Postcondition: State what is true after the function has executed.
//Description: Describe what the function does (purpose).
//*****

void output(ofstream& out, const order_record& plant_record)
{
    //Use the following statement to help you format your output. These statements are called
the magic formula.
    /*out.setf(ios::showpoint);
    out.precision(2);
    out.setf(ios::fixed);*/
    cout.setf(ios::showpoint);
    cout.precision(2);
    cout.setf(ios::fixed);

    //put your code here to print the formatted output. Check out the following code sample.
    //cout << left << setw(15) << plant_record.pname << left << setw(8) << plant_record.cname
        //<< right << setw(8) << plant_record.plant_cost << right << setw(8) <<
plant_record.quantity
}
//Here is your driver to test the program
int main()
{

```

```

order_record purchase_record;

ifstream in;    //declaring an input file stream
in.open("nursery_stock.txt"); //opening the input file stream
ofstream out;
out.open("sample_calculations.txt");

if (in.fail())
{
    cout << "Input file did not open correctly" << endl;
}
else
{
    //the following while loop processes one order record at a time
    while (!in.eof())
    {
        input(in, purchase_record);
        process(purchase_record);
        output(out, purchase_record);
    }
}

in.close(); //always close opened files.
out.close();
return 0;
}

```

Handing in your program:

Electronically submit "*nursery_inv3.cpp*" in the Assignments area of Canvas before the due date and time. **Remember, complete the assignment no matter if it is late. It is very important that you do all assignments to master the C++ programming language and get a passing grade.**

