

Lab. (9) ***structs***

Answer the following questions

Question One:

Give the output of the following program. Assuming all the desired header files are already included, which are required to run the code.

```
struct Pixel
{
    int C, R;
};
void Display (Pixel P)
{
    cout << "Col "<< P.C << " Row " << P.R << endl;
}
int main()
{
    Pixel X = {40,50}, Y, Z;
    Z = X;
    X.C += 10;
    Y = Z;
    Y.C += 10;
    Y.R += 20;
    Z.C -= 15;
    Display(X);
    Display(Y);
    Display(Z);
    return 0;
}
```

Question Two:

Find the output of the following program. Assuming all the desired header files are already included, which are required to run the code.

```
struct Play
{
    int score, bonus;
};
void calculate(Play &P, int N = 10)
{
    P.score++;
}
```

```

        P.bonus += N;
    }

int main()
{
    Play PL = { 10, 15 };
    calculate(PL, 5);
    cout << PL.score << ":" << PL.bonus << endl;
    calculate(PL);
    cout << PL.score << ":" << PL.bonus << endl;
    calculate(PL, 15);
    cout << PL.score << ":" << PL.bonus << endl;
    return 0;
}

```

Question Three:

Find the output of the following program. Assuming all the desired header files are already included, which are required to run the code.

```

struct MyBox
{
    int length, breadth, height;
};
void dimension (MyBox M)
{
    cout << M.length << "x" << M.breadth << "x";
    cout << M.height << endl;
}

int main ()
{
    MyBox B1 = { 10, 15, 5 }, B2, B3;
    ++B1.height;
    dimension(B1);
    B3 = B1;
    ++B3.length;
    B3.breadth++;
    dimension(B3);
    B2 = B3;
    B2.height += 5;
    B2.length--;
    dimension(B2);
    return 0;
}

```

Question Four:

Declare a structure to represent a complex number (a number having a real part and imaginary part). Write C++ functions to add, subtract, multiply and divide two complex numbers.

Question Five:

An array stores details of 25 students (roll no, name, marks in three subject). Write a program to create such an array and print out a list of students who have failed in more than one subject.

Question Six:

A company has six salespeople. Every month they go on road trips to sell the company's product. At the end of each month, the total sales for each salesperson, salesperson's ID, and the month, are recorded in a file. At the end of each year, the manager of the company asks for a report

----- Annual Sales Report -----					
ID	QT1	QT2	QT3	QT4	Total
12345	1892.00	0.00	494.00	322.00	2708.00
32214	343.00	892.00	9023.00	0.00	10258.00
23422	1395.00	1901.00	0.00	0.00	3296.00
57373	893.00	892.00	8834.00	0.00	10619.00
35864	2882.00	1221.00	0.00	1223.00	5326.00
54654	893.00	0.00	392.00	3420.00	4705.00
Total	8298.00	4906.00	18743.00	4965.00	

Max Sale by Salesperson: ID = 57373, Amount = \$10619.00
 Max Sale by Quarter: Quarter = 3, Amount = \$18743.00

QT1 stands for quarter 1 (months 1 to 3), QT2 for quarter 2 (months 4 to 6), QT3 for quarter 3 (months 7 to 9) and QT4 for quarter 4 (months 10 to 12)

The salespeople IDs are stored in one file; sales data are stored in another file

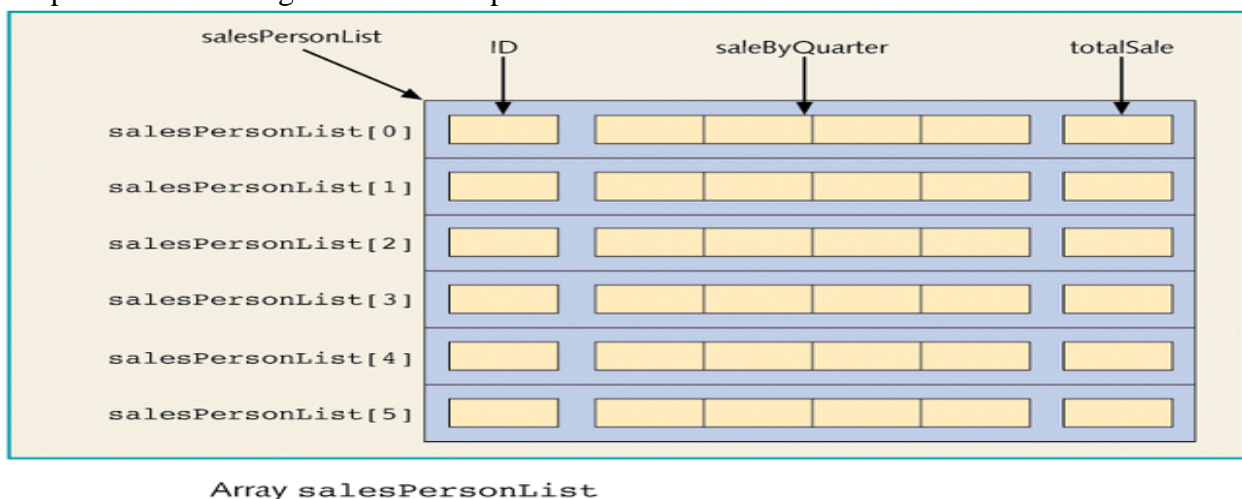
The sales data is in the following form:

salesPersonID month saleAmount

Sales data are not ordered

Input: file containing each salesperson's ID, and a second file containing the sales data

Output: file containing annual sales report in the above format



Program Analysis

- For each entry in the file containing the sales data
 1. Read ID, month, sale amount for the month
 2. Search salesPersonList to locate the component corresponding to this salesperson
 3. Determine the quarter corresponding to the month
 4. Update the sales for the quarter by adding the sale amount for the month
- Once the sales data file is processed
 1. Calculate the total sale by salesman
 2. Calculate the total sale by quarter
 3. Print the report
- Translates into the following algorithm
 1. Initialize the array sales
 2. Process the sales data
 3. Calculate the total sale by salesman
 4. Calculate the total sale by quarter
 5. Print the report
 6. Calculate and print maximum sale by salesman
 7. Calculate and print maximum sale by quarter

Main Algorithm

1. Declare the variables
2. Prompt user to enter name of file containing the salesperson's ID data
3. Read the name of the input file
4. Open the input file
5. If input file does not exist, exit
6. Initialize the array salesPersonList by calling the function initialize
7. Close input file containing salesperson's ID
8. Prompt user to enter name of file containing sales data
9. Read the name of the input file
10. Open the input file
11. If input file does not exist, exit
12. Prompt user to enter name of output file
13. Read the name of the output file
14. Open the output file
15. Output data to two decimal places
16. Process sales data
 1. Call the function getData
17. Calculate the total sale by quarter by calling the function saleByQuarter
18. Calculate the total sale by salesman by calling the function totalSaleByPerson
19. Print the report in the tabular form. Call the function printReport
20. Find and print the salesperson who produces the maximum sales for the year by calling the function maxSaleByPerson
21. Find and print the quarter producing the maximum sale for the year by calling the function maxSaleByQuarter
22. Close files