Lab Assingment-2

ROLL: 2005535 | NAME: SAHIL SINGH | DATE: 05/08/21

QUES 1: WAP to display the message "hello" followed by your name on screen.

SOLUTION:

```
#include <iostream>
using namespace std;
int main()
{
    cout << "Hello Sahil!!!" << endl;
    return 0;
}
```

OUTPUT:

```
Hello Sahil!!!
```

QUES 2: Create a class which stores name, roll number and total marks for a student. Input the data for a student and display it.

SOLUTION:

```
#include <iostream>
#include <string>
using namespace std;

class student_535
{
public:
    string name_535;
    int roll_535, marks_535;

    void get_student_data_535()
    {
        cout << "Enter Name: ";
        getline(cin, name_535);

        cout << "Enter Roll Number: ";
        cin >> roll_535;

        cout << "Enter Marks: ";
        cin >> marks_535;
    }

    void display_data_535()
    {
        cout << "--------------------" << endl;
        cout << "Name: " << name_535 << endl;
```

```
        cout << "Roll Number: " << roll_535 << endl;
        cout << "Marks: " << marks_535 << endl;
    }
};

int main()
{
    student_535 ob;
    ob.get_student_data_535();
    ob.display_data_535();
    return 0;
}
```

OUTPUT:

```
Enter Name: Sahil Singh
Enter Roll Number: 2005535
Enter Marks: 98
--------------------
Name: Sahil Singh
Roll Number: 2005535
Marks: 98
```

QUES 3: Modify the program ii) to store marks in 5 subjects. Calculate the total marks and percentage of a student and display it.

SOLUTION:

```
#include <iostream>
#include <string>
using namespace std;

class student_535
{
public:
    string name_535;
    int roll_535;
    int marks_535[5];
    int tmarks_535 = 0;
    double percentage_535;

    void get_student_data()
    {
        cout << "Enter Name: ";
        getline(cin, name_535);

        cout << "Enter Roll Number: ";
        cin >> roll_535;

        cout << "Enter Marks of five subjects:\n";
```

```
        for (int i = 0; i < 5; i++)
        {
            cout << "Enter Marks of Subject " << i + 1 << " : ";
            cin >> marks_535[i];
            tmarks_535 = (tmarks_535 + marks_535[i]);
        }
    }
    void display_data()
    {
        cout << "\n-----------------------" << endl;
        cout << "Student's Name: " << name_535 << endl;
        cout << "Roll Number: " << roll_535 << endl;
        for (int i = 0; i < 5; i++)
        {
            cout << "Marks of Subject " << i << " : ";
            cout << marks_535[i] << endl;
        }
        cout << "Total Marks: " << (tmarks_535) << endl;
        cout << "Percentage : " << (tmarks_535 / 5) << "%" << endl;
    }
};

int main()
{
    student_535 ob;
    ob.get_student_data();
    ob.display_data();
    return 0;
}
```

OUTPUT:

```
Enter Name: Sahil Singh
Enter Roll Number: 2005535
Enter Marks of five subjects:
Enter Marks of Subject 1 : 67
Enter Marks of Subject 2 : 65
Enter Marks of Subject 3 : 78
Enter Marks of Subject 4 : 67
Enter Marks of Subject 5 : 89

-----------------------
Student’s Name: Sahil Singh
Roll Number: 2005535
Marks of Subject 0 : 67
Marks of Subject 1 : 65
Marks of Subject 2 : 78
Marks of Subject 3 : 67
Marks of Subject 4 : 89
Total Marks: 366
```

```
Percentage : 73%
```

QUES 4: Create a class complex which stores real and imaginary part of a complex number. Input 10 complex numbers and display them.

SOLUTION:

```cpp
#include <iostream>
using namespace std;

class complex
{
public:
    int real_535;
    int img_535;

    void get_num_data()
    {
        cout << "\nEnter Real Part: ";
        cin >> real_535;

        cout << "Enter Imaginary Part: ";
        cin >> img_535;
    }

    void display_num()
    {
        static int n_535 = 1;
        cout << "Number " << n_535++ << " : " << real_535 << "+" << img_535 << "i" << endl;
    }
};

int main()
{
    complex ob[10];
    for (int i_535 = 0; i_535 < 10; i_535++)
    {
        ob[i_535].get_num_data();
    }
    cout << "----------------------" << endl;
    for (int i_535 = 0; i_535 < 10; i_535++)
    {
        ob[i_535].display_num();
    }
    return 0;
}
```

OUTPUT:

```
Enter Real Part: 34
```

```
Enter Imaginary Part: 23

Enter Real Part: 45
Enter Imaginary Part: 56

Enter Real Part: 312
Enter Imaginary Part: 45

Enter Real Part: 67
Enter Imaginary Part: 344

Enter Real Part: 456
Enter Imaginary Part: 5

Enter Real Part: 34
Enter Imaginary Part: 67

Enter Real Part: 345
Enter Imaginary Part: 67

Enter Real Part: 68
Enter Imaginary Part: 87

Enter Real Part: 45
Enter Imaginary Part: 3

Enter Real Part: 56
Enter Imaginary Part: 67
----------------------
Number 1 : 34+23i
Number 2 : 45+56i
Number 3 : 312+45i
Number 4 : 67+344i
Number 5 : 456+5i
Number 6 : 34+67i
Number 7 : 345+67i
Number 8 : 68+87i
Number 9 : 45+3i
Number 10 : 56+67i
```

QUES 5: Create a class distance which stores a distance in feet and inches. Input 2 distance values in objects, add them, store the resultant distance in and object and display it.

Write the above program in two ways.

a) store the resultant distance in the calling object:C3.add(C1,C2)

b) return the resultant object C3=C1.add(C2)

SOLUTION:

```cpp
#include <iostream>
using namespace std;
class dist
{
public:
    double feet_535;
    double inches_535;
    void getdata()
    {
        cout << "Enter Feet and Inches: ";
        cin >> feet_535 >> inches_535;
    }
    void display()
    {
        cout << feet_535 << "'" << inches_535 << "''" << endl;
    }
    void add(dist a, dist b)
    {
        inches_535 = a.inches_535 + b.inches_535;
        feet_535 = a.feet_535 + b.feet_535;
    }
    dist add(dist b)
    {
        dist temp_535;
        temp_535.inches_535 = inches_535 + b.inches_535;
        temp_535.feet_535 = feet_535 + b.feet_535;
        return temp_535;
    }
};
int main()
{
    dist c1, c2, c3, c4;
    c1.getdata();
    c2.getdata();
    cout << "Distance 1: ";
    c1.display();
    cout << "Distance 2: ";
    c2.display();
    c3.add(c1, c2);
    cout << "Output for c3.add(c1,c2): " << endl;
    c3.display();
    c4 = c1.add(c2);
    cout << "Output for c4=c1.add(c2): " << endl;
    c4.display();
    return 0;
}
```

OUTPUT:

```
Enter Feet and Inches: 5 4
```

```
Enter Feet and Inches: 4 8
Distance 1: 5'4''
Distance 2: 4'8''
Output for c3.add(c1,c2):
9'12''
Output for c4=c1.add(c2):
9'12''
```

QUES 6: Create a class which stores id, name, age and basic salary of an employee. Input data for n number of employees. Calculate the gross salary of all the employees and display it along with all other details in a tabular form. [Gross salary= Basic salary + DA + HRA, DA = 80% of Basic salary HRA=10% of Basic salary]

SOLUTION:

```
#include <iostream>
using namespace std;
class employee
{
public:
    string name_535;
    int age_535;
    int basic_535, DA_535, HRA_535;
    double gross_535;
    void getdata()
    {
        cout << "Enter Name: ";
        cin >> name_535;
        cout << "Enter Age: ";
        cin >> age_535;
        cout << "Enter Basic salary: ";
        cin >> basic_535;
        DA_535 = 0.8 * basic_535;
        HRA_535 = 0.1 * basic_535;
        gross_535 = basic_535 + DA_535 + HRA_535;
    }
    void display()
    {
        cout << "Name: " << name_535 << endl;
        cout << "Age: " << age_535 << endl;
        cout << "Basic salary = " << basic_535 << endl;
        cout << "Gross salary = " << gross_535 << endl;
    }
};
int main()
{
    int n_535 = 0;
    cout << "Enter number of employee: ";
    cin >> n_535;
    employee ob[n_535];
```

```
    for (int i_535 = 0; i_535 < n_535; i_535++)
    {
        cout << "Enter Employee " << i_535 + 1 << " details" << endl;
        ob[i_535].getdata();
    }
    cout << "------------------------" << endl;
    for (int i_535 = 0; i_535 < n_535; i_535++)
    {
        cout << "Employee " << i_535 + 1 << endl;
        ob[i_535].display();
        cout << "------------------" << endl;
    }
    return 0;
}
```

OUTPUT:

```
Enter number of employee: 2
Enter Employee 1 details
Enter Name: Sahil
Enter Age: 18
Enter Basic salary: 200000
Enter Employee 2 details
Enter Name: Singh
Enter Age: 19
Enter Basic salary: 300000
------------------------
Employee 1
Name: Sahil
Age: 18
Basic salary = 200000
Gross salary = 380000
------------------
Employee 2
Name: Singh
Age: 19
Basic salary = 300000
Gross salary = 570000
------------------
```

QUES 7: Create a class which stores x and y coordinates of a point. Calculate distance between two given points and display it.

SOLUTION:

```
#include <iostream>
#include <math.h>
using namespace std;
class dist
{
    int x_535, y_535;
```

```
public:
    void getdata()
    {
        cout << "Enter x and y coordinates : ";
        cin >> x_535 >> y_535;
    }
    void display()
    {
        cout << "(" << x_535 << "," << y_535 << ")" << endl;
    }
    double add(dist a, dist b)
    {
        return sqrt(pow(b.y_535 - a.y_535, 2) + pow(b.x_535 - a.x_535, 2));
    }
};
int main()
{
    dist c1, c2;
    c1.getdata();
    c2.getdata();
    cout << "Coordinate 1: ";
    c1.display();
    cout << "Coordinate 2: ";
    c2.display();
    c1.add(c1, c2);
    cout << "Distance between them = " << c1.add(c1, c2) << endl;
    return 0;
}
```

SOLUTION:

```
Enter x and y coordinates : 2 3
Enter x and y coordinates : 1 2
Coordinate 1: (2,3)
Coordinate 2: (1,2)
Distance between them = 1.41421
```

QUES 8: WAP to input name, roll number and marks in 5 subjects for a student, and display it.

SOLUTION:

```
#include <iostream>
using namespace std;
int main()
{
    string name_535;
    int roll_535;
    int marks_535[5];
    cout << "Enter Name then roll number and followed by 5 marks :-" << endl;
```

```
    cin >> name_535 >> roll_535;
    for (int i = 0; i < 5; i++)
    {
        cin >> marks_535[i];
    }

    cout << "\nDetails Entered :- \nName : " << name_535 << "\nRoll Number : " << roll_535 <<
 endl;

    for (int i = 0; i < 5; i++)
    {
        cout << "Marks " << i + 1 << " : " << marks_535[i] << endl;
    }
    return 0;
}
```

OUTPUT:

```
Enter Name then roll number and followed by 5 marks :-
Sahil
2005535
45
56
67
78
89

Details Entered :-
Name : Sahil
Roll Number : 2005535
Marks 1 : 45
Marks 2 : 56
Marks 3 : 67
Marks 4 : 78
Marks 5 : 89
```

QUES 9: WAP to input name, roll number and marks in 5 subjects for n number of students. Write functions to:- a. Find total marks and percentage of all n students. b. Display details of a student with a given roll number. c. Display the details for all the students having percentage in a given range. d. Sort the array in ascending order of marks.

SOLUTION:

```
#include <iostream>
using namespace std;
int main()
{
    int n, s;
    cout << "Enter number of students : ";
    cin >> n;
```

```
    string name[n + 1];
    int roll[n + 1];
    int marks[n + 1][5];
    int total[n + 1] = {};
    float percent[n + 1];
    for (int i = 0; i < n; i++)
    {
        cout << "Enter Name then roll number and followed by 5 marks of student " << i + 1 << " :-
" << endl;
        cin >> name[i] >> roll[i];
        for (int ii = 0; ii < 5; ii++)
        {
            cin >> marks[i][ii];
            total[i] = total[i] + marks[i][ii];
        }
        percent[i] = total[i] * 2;
    }
    cout << "Percentage :- " << endl;
    for (int i = 0; i < n; i++)
        cout << "Student " << i + 1 << " : " << percent[i] << endl;
    cout << "Enter a roll number to display details : ";
    cin >> s;
    for (int j = 0; j < n; j++)
        if (s == roll[j])
        {
            cout << "\n\nDetails :- \n\nName : " << name[j] << "\nRoll Number : " << roll[j] << endl;
            for (int i = 0; i < 5; i++)
                cout << "Marks " << i + 1 << " : " << marks[j][i] << endl;
            break;
        }
    int range1, range2;
    cout << "Enter the starting and ending range of percentage :- ";
    cin >> range1 >> range2;
    for (int i = 0; i < n; i++)
    {
        if (percent[i] >= range1 && percent[i] <= range2)
        {
            cout << "\n\nDetails :- \n\nName : " << name[i] << "\nRoll Number : " << roll[i] << endl;
            for (int ii = 0; ii < 5; ii++)
                cout << "Marks " << ii + 1 << " : " << marks[i][ii] << endl;
        }
    }
    cout << "\n\nAfter Sorting :-- " << endl;
    for (int i = 0; i < n - 1; i++)
    {
        for (int j = i + 1; j < n; j++)
        {
            if (total[i] > total[j])
            {
                name[n] = name[i];
                name[i] = name[j];
                name[j] = name[n];
                roll[n] = roll[i];
                roll[i] = roll[j];
                roll[j] = roll[n];
                for (int ii = 0; ii < 5; ii++)
```

```
                {
                    marks[n][ii] = marks[i][ii];
                    marks[i][ii] = marks[j][ii];
                    marks[j][ii] = marks[n][ii];
                }
                total[n] = total[i];
                total[i] = total[j];
                total[j] = total[n];
                percent[n] = percent[i];
                percent[i] = percent[j];
                percent[j] = percent[n];
            }
        }
    }
    cout << "\n\nDetails :-\n"
         << endl;
    for (int i = 0; i < n; i++)
    {
        cout << "\n\nName : " << name[i] << "\nRoll Number : " << roll[i] << endl;
        for (int ii = 0; ii < 5; ii++)
            cout << "Marks " << ii + 1 << " : " << marks[i][ii] << endl;
    }
    return 0;
}
```

OUTPUT:

```
Enter number of students : 2
Enter Name then roll number and followed by 5 marks of student 1 :-
SAHIL
2005535
90
89
98
77
88
Enter Name then roll number and followed by 5 marks of student 2 :-
SINGH
2005536
67
56
45
67
23
Percentage :-
Student 1 : 884
Student 2 : 516
Enter a roll number to display details : 2005536

Details :-

Name : SINGH
```

```
Roll Number : 2005536
Marks 1 : 67
Marks 2 : 56
Marks 3 : 45
Marks 4 : 67
Marks 5 : 23
Enter the starting and ending range of percentage :- 80 90

After Sorting :--

Details :-

Name : SINGH
Roll Number : 2005536
Marks 1 : 67
Marks 2 : 56
Marks 3 : 45
Marks 4 : 67
Marks 5 : 23

Name : SAHIL
Roll Number : 2005535
Marks 1 : 90
Marks 2 : 89
Marks 3 : 98
Marks 4 : 77
Marks 5 : 88
```

QUES 10: WAP to enter id, name, age and basic salary of n number of employees. Calculate the gross salary of all the employees and display it along with all other details in a tabular form, using pointer to structure. [Gross salary= Basic salary + DA + HRA, DA = 80% of Basic salary, HRA=10% of Basic salary]

SOLUTION:

```cpp
#include <iostream>
using namespace std;
struct emp
{
    int id_535;
    string nam_535;
    int age_535;
    int sal_535;
    float gross_535;
};
void display(struct emp *st, int l)
{
    cout << "Details Entered : - " << endl;
    for (int i = 0; i < l; i++)
    {
```

```
        cout << "Employee number " << i + 1 << " ID : " << st->id_535 << " Name : " << st-
>nam_535 << " Age : " << st->age_535 << " Sal : " << st->sal_535 << " Gross Sal : " << st-
>gross_535 << endl;
        st++;
    }
}
int main()
{
    int n;
    cout << "Enter number of employees : ";
    cin >> n;
    struct emp a[n];
    for (int i = 0; i < n; i++)
    {
        cout << "Enter ID, name, age, basic sal of Employee " << i + 1 << endl;
        cin >> a[i].id_535 >> a[i].nam_535 >> a[i].age_535 >> a[i].sal_535;
        a[i].gross_535 = 1.9 * a[i].sal_535;
    }
    struct emp *s = a;
    display(s, n);
}
```

OUTPUT:

```
Enter number of employees : 2
Enter ID, name, age, basic sal of Employee 1
2005535
SAHIL
18
180000
Enter ID, name, age, basic sal of Employee 2
2005536
SINGH
19
200000
Details Entered : -
Employee number 1ID : 2005535 Name : SAHIL Age : 18 Sal : 180000 Gross Sal : 342000
Employee number 2ID : 2005536 Name : SINGH Age : 19 Sal : 200000 Gross Sal : 380000
```