

## Lab 9 Class and Objects

Please test the correctness of your programs using **PASS**.

### Q-1.

Design and implement a class named "**Employee**" to compute the net salary of an employee using the basic salary, medical allowance, and house rent (45% of basic salary).

The design of **Employee** class should include:

- 1) Three private members **basic\_sal**, **med\_allow**, and **h\_rent**, which present *basic salary*, *medical allowance* and *house rent* of an employee.
- 2) One private member **net\_sal** to preserve the computed *net salary* of an employee.
- 3) A **default constructor** (i.e., **Employee()**) to initialize the **basic\_sal**, **med\_allow**, and **h\_rent** as 3000, 500, and 1350 (as 45% of 3000).
- 4) A **parameter constructor** (i.e., **Employee(int, int)**) to initialize the values of **basic\_sal** and **med\_allow** of an employee.
- 5) A **set method** (i.e., **set(int, int)**) to enter the new *basic salary* and *medical allowance* of an employee.
- 6) A method (i.e., **computeNetSalary()**) to compute the *net salary* of an employee.

The net salary of an employee should be computed through the following formula: net salary = basic salary + medical allowance + house rent, where house rent = 45% of basic salary.

- 7) A **get method** (i.e., **get()**) to display the *net salary* of an employee.

According to the requirements above, write a program to compute the net salary of two employees: Create first Employee object (i.e., **Emp1**) with default constructor.

- 1) Enter the value of *basic salary* and *medical allowance* of second employee (i.e., **Emp2**) and create it with parameter constructor.
- 2) Compute the *net salary* of both employees (**Emp1** and **Emp2**) by calling **computeNetSalary()** method.
- 3) Display the *net salary* of both employees (**Emp1** and **Emp2**) by calling **get()** method.
- 4) Enter new *basic salary* and medical allowance of **Emp1** by calling **set()** method.
- 5) Compute and display the *net salary* of **Emp1**.

### Expected Outputs:

```
Enter basic salary for Emp2:
4000
Enter medical allowance for Emp2:
600
The net salary for Emp1 is 4850
The net salary for Emp2 is 6400
Enter basic salary for Emp1:
5000
Enter medical allowance for Emp1:
1000
The net salary for Emp1 is 8250
```

**Q-2.**

Design and implement a class to compute the area of a triangle using the values of their sides. The design of class (named **Triangle**) should include:

- 1) Three private members **side1**, **side2**, and **side3**, which present sides of a triangle.
- 2) One private member **area** to preserve the computed area of triangle.
- 3) A default constructor (i.e., **Traingle()**) to initialize sides of triangle and area of triangle as zero.
- 4) A parameter constructor (i.e., **Traingle(double, double, double)**) to initialize the values of sides for corresponding triangle and initialize area of triangle as zero.
- 5) A set method (i.e. **setSides(double, double, double)**) to enter the new values for the sides.
- 6) A method (i.e. **computeArea()**) to compute the area of triangle.
- 7) A get method (i.e. **getArea()**) to display the area.

**Hint-1.** Area of Triangle can be computed through the following Heron's Formula:  
 $Area = \sqrt{s(s-a)(s-b)(s-c)}$ , where  $s=(a+b+c)/2$ .

**Hint-2.** You need to include `<cmath>` library to use the `sqrt()` function and include `<iomanip>` library to use `setprecision` and `fixed` manipulators to change the precision value and printing format.

According to the requirements above, write a program to:

- 1) Create first Triangle object (i.e., **triangle1**) with default constructor and display its area.
- 2) Create second Triangle object (i.e., **triangle2**) with parameter constructor and display its area.  
 The values of sides are 3, 4, and 5.
- 3) Call the set method using first Triangle object (i.e., **triangle1**) and display its area. Assume that all inputs are valid, i.e., new sides can form a triangle.

**Expected Outputs:**

Example 1.	Example 2.
Area of triangle1: 0.00 Area of triangle2: 6.00 Enter new sides for triangle1: <b>4 2 3</b> Area of triangle1: 2.90	Area of triangle1: 0.00 Area of triangle2: 6.00 Enter new sides for triangle1: <b>3 3 3</b> Area of triangle1: 3.90

**Q-3.**

Extend the program in **Q-2** to define a non-member function `largertriangle()`, which accepts two `Triangle` objects (i.e., `triangle1` and `triangle2`) as input. The function should print whether the first `Triangle` (i.e., `triangle1`) has a larger area than the second one (i.e., `triangle2`) or not. You may use the following template for `largertriangle()`.

```
void largertriangle(Triangle t1, Triangle t2)
{
}

```

Modify the program in **Q-2** by creating two objects with default parameters and call the set method to enter the new values for the sides of both triangles. Note: You don't need to consider two triangles with the same areas.

**Expected Outputs:****Example 1.**

```
Enter new sides for triangle1:
4 5 6
Enter new sides for triangle2:
3 4 5
The area of triangle1 (9.92) is greater than that of triangle2 (6.00).

```

**Example 2.**

```
Enter new sides for triangle1:
3 4 5
Enter new sides for triangle2:
4 5 6
The area of triangle1 (6.00) is not greater than that of triangle2 (9.92).

```

**Q-4. (To be marked)**

Modify the program in **Q-2** such that the program defines an array of triangles `Triangle arr[10]`.

The program should let the user input the sides of  $n$  triangles (where  $n$  is an integer input by the user and  $1 \leq n \leq 10$ ) and store them in the array. The program should print the area of all triangles and also print the name of the triangle with the largest area (if there are two or more triangles with the same maximum area, only print the first one in the original order that they are defined).

Assume that all inputs are valid, i.e.,  $n$  is within the range specified, and those three sides can form a triangle.

**Expected Outputs:****Example 1.**

```
Enter a number between 1 and 10:
4
Enter the sides of triangle 1:
3 3 3
Enter the sides of triangle 2:
3 4 5
Enter the sides of triangle 3:
3 4 5
Enter the sides of triangle 4:
3 2 3
Area of triangle 1: 3.90
Area of triangle 2: 6.00
Area of triangle 3: 6.00
Area of triangle 4: 2.83
Triangle 2 has the largest area: 6.00
```

**Example 2.**

```
Enter a number between 1 and 10:
5
Enter the sides of triangle 1:
6 6 6
Enter the sides of triangle 2:
4 5 6
Enter the sides of triangle 3:
7 8 9
Enter the sides of triangle 4:
3 4 5
Enter the sides of triangle 5:
5 7 5
Area of triangle 1: 15.59
Area of triangle 2: 9.92
Area of triangle 3: 26.83
Area of triangle 4: 6.00
Area of triangle 5: 12.50
Triangle 3 has the largest area: 26.83
```

Please test the correctness of your programs in **Q1**, **Q2a**, **Q2b**, and **Q3** using **PASS**.