SVKM's NMIMS

Mukesh Patel School of Technology Management and Engineering, Mumbai

Programming for Problem Solving (Exp 10)

Roll No: J001	**Name: Adith Ramakrishna**
Program: B. Tech Data Science (1st)	**Batch: J1**
Date of Experiment: 30/11/2022	**Date of Submission: 4/12/2022**

Task 1:

Code:

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

class student {
   string name;
   int roll_no;

   public:
   student(string n = "", int r = 0) {
      name = n;
      roll_no = r;
   }

   void showData() {
      cout << "\nStudent Data:\nName: " << name << "\nRoll No: " <<
roll_no << endl;
   }
};

int main() {
   student stud("John", 2);
   stud.showData();
   return 0;
}
```

Task 2:

Code:

```
#include <iostream>
#include <cmath>

using namespace std;

class triangle {
   private:
   int s1, s2, s3;

   public:
   triangle(int side1 = 0, int side2 = 0, int side3 = 0) {
      s1 = side1;
      s2 = side2;
      s3 = side3;
   }

   int getArea() {
      int s = (s1 + s2 + s3) / 2;
      return sqrt(s * (s - s1) * (s - s2) * (s - s3));
   }

   int getPerimeter() {
      return s1 + s2 + s3;
   }
};

int main() {
   triangle t(3, 4, 5);
   cout << "Area: " << t.getArea() << "\nPerimeter: " << t.getPerimeter()
<< endl;
   return 0;
}
```

Task 3:

Code:

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

class Complex {
   private:
   int real, imag;

   public:
   Complex(int r = 0, int i = 0) {
      real = r;
      imag = i;
   }

   Complex operator + (Complex const & obj) {
      Complex res;
      res.real = real + obj.real;
      res.imag = imag + obj.imag;
      return res;
   }

   Complex operator - (Complex const & obj) {
      Complex res;
      res.real = real - obj.real;
      res.imag = imag - obj.imag;
      return res;
   }

   void print() {
      cout << real << " + " << imag << "i\n";
   }
};
```

```
int main() {
   int r1, i1, r2, i2;

   cout << "Enter Complex Number 1: ";
   cin >> r1 >> i1;
   Complex c1(r1, i1);

   cout << "Enter Complex Number 2: ";
   cin >> r2 >> i2;
   Complex c2(r2, i2);

   cout << "\nSum: ";
   (c1 + c2).print();

   cout << "Difference: ";
   (c1 - c2).print();
}
```

Task 4:

Code:

```
#include <iostream>
#include <string>

using namespace std;

class student {
   private:
   int roll_no = 0;
   string name = "";
   string branch = "";
```

```
  public:
  void getData() {
     cout << "Enter Roll No: ";
     cin >> roll_no;
     cout << "Enter Name: ";
     cin >> name;
     cout << "Enter Branch: ";
     cin >> branch;
  }

  void showData() {
     cout << "\n\nStudent Data:\nName: " << name;
     cout << "\nRoll No: " << roll_no;
     cout << "\nBranch: " << branch;
     cout << endl;
  }
};

int main() {
  student stud;
  stud.getData();
  stud.showData();
  return 0;
}
```

Homework Questions:

1:

Class is a blueprint or a template from which an object is created. Whereas an object is an instance of the class.

2:

Access specifiers define how the members of a class can be accessed.