Q1 Write a C++ program to subtract two integer numbers of two different classes using friend function.

```
#include <iostream>
using namespace std;
class B; class A {
private: int num;
public: A(int n) { num = n; }
friend int subtract(A objA, B objB);
}; class B {
private:  int num;
public:  B(int n) { num = n; }
   friend int subtract(A objA, B objB);
};int subtract(A objA, B objB) {
   return objA.num - objB.num;
}int main() {
   A objA(10);
   B objB(5);   int result = subtract(objA, objB);
   cout << "The difference between " << objA.num << " and " << objB.num << " is " << result << endl;

   return 0;
}
```

Q2 Write a C++ program to read a text file and count number of Upper-case Alphabets, Lowercase Alphabets Digits and Spaces using File Handling

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

int main() {
   ifstream file("example.txt");
   if (!file.is_open()) {
      cout << "Failed to open file." << endl;
      return 1;
   }
   int uppercase_count = 0;
   int lowercase_count = 0;
   int digit_count = 0;
   int space_count = 0;

   char ch;
   while (file.get(ch)) {
      if (isupper(ch)) {
         uppercase_count++;
      } else if (islower(ch)) {
         lowercase_count++;
      } else if (isdigit(ch)) {
         digit_count++;
      } else if (isspace(ch)) {
         space_count++;
      }  }  cout << "Number of uppercase letters: " << uppercase_count << endl;
   cout << "Number of lowercase letters: " << lowercase_count << endl;
   cout << "Number of digits: " << digit_count << endl;
   cout << "Number of spaces: " << space_count << endl;  file.close();

   return 0;
}
```

Q3 Write a C++ program to overload function volume and find volume of cube, cylinder and sphere.

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

using namespace std;  class Shape

 { public :   virtual double volume() = 0; };

class Cube:  public Shape { private:  double
length; public:   Cube(double l) { length = l; }

double volume() { return pow(length, 3); } };
class Cylinder : public Shape {
private:  double radius;  double height;

public:  Cylinder(double r, double h) { radius =
r; height = h; }  double volume() { return M_PI
* pow(radius, 2) * height; }

};class Sphere : public Shape {

private:  double radius;

public:  Sphere(double r) { radius = r; }

  double volume() { return 4.0 / 3.0 * M_PI *
pow(radius, 3); } };  int main() {

    Shape *shapes[3];

    shapes[0] = new Cube(5);

    shapes[1] = new Cylinder(3, 4);

    shapes[2] = new Sphere(2);

    for (int i = 0; i < 3; i++) {

        cout << "Volume of shape " << i+1 << ": "
<< shapes[i]->volume() << endl;

    }  return 0;

}
```

Q4 Write a C++ program to find area and volume of cylinder using Inline function

```
#include <iostream>

using namespace std;

inline double area(double r, double h) {

 return 2 * 3.14159 * r * h + 2 * 3.14159 * r *
r; }   inline double volume(double r, double h)
{  return 3.14159 * r * r * h;

 }  int main() {  double radius, height;

   cout << "Enter the radius of the cylinder: ";

    cin >> radius;

    cout << "Enter the height of the cylinder: ";

    cin >> height;

 cout << "The surface area of the cylinder is: "
<< area(radius, height) << endl;

    cout << "The volume of the cylinder is: " <<
volume(radius, height) << endl;

  return 0;  }
```

Q5 Write a C++ program to accept length and width of a rectangle. Calculate and display perimeter as well as area of a rectangle by using Inline function

```
#include <iostream>

using namespace std;

inline double area(double l, double w) {

    return l * w;  }

inline double perimeter(double l, double w) {

  return 2 * (l + w);  }

int main() {   double length, width;

 cout << "Enter the length of the rectangle: ";

    cin >> length;

    cout << "Enter the width of the rectangle: ";

    cin >> width;

   cout << "The area of the rectangle is: " <<
area(length, width) << endl;

    cout << "The perimeter of the rectangle is: "
<< perimeter(length, width) << endl;

   return 0;

}
```

Q6 Write a C++ program using function to count and display the number of lines not starting with alphabet “A” in a text file

```
#include <iostream>

#include <fstream>

using namespace std;

int countLinesNotStartingWithA(string
filename) {

    int count = 0;

    string line;

   ifstream file(filename);

  if (file.is_open()) {

        while (getline(file, line)) {

            if (line[0] != 'A' && line[0] != 'a') {

                count++;

            }

        }

        file.close();

    } else {

        cout << "Error: Unable to open file." <<
endl;

        return -1;

    } return count;

} int main() {

    string filename;

    int count;



    cout << "Enter the name of the file: ";

    cin >> filename;

 count =
countLinesNotStartingWithA(filename);



    if (count >= 0) {

        cout << "The number of lines not starting
with the alphabet 'A' is: " << count << endl;

    }



    return 0;

}
```

Q7 Write a C++ program to swap two integer values and two float values by using function template

```
#include <iostream>

using namespace std;

template<typename T>

void swap(T &a, T &b) {

    T temp = a;

    a = b;

    b = temp;

}int main() {

    int a = 10, b = 20;

    float x = 3.14, y = 2.71;

  cout << "Before swapping: " << endl;

    cout << "a = " << a << ", b = " << b << endl;

    cout << "x = " << x << ", y = " << y << endl;

  swap(a, b);

    swap(x, y);



    cout << "After swapping: " << endl;

    cout << "a = " << a << ", b = " << b << endl;

    cout << "x = " << x << ", y = " << y << endl;



    return 0;

}
```

Q8 Write a C++ program to calculate area of cone, sphere and circle by using function overloading

```
#include <iostream>
using namespace std;
const float PI = 3.14159;
float area(float r) {
   return PI * r * r;
}  float area(float r, float h) {
   return 2 * PI * r * (r + h);  }
float area(float r, float h, float l) {
   return PI * r * (r + l);  }
int main() {  float r, h, l;
  cout << "Enter the radius of the circle: ";
   cin >> r;    cout << "Area of the circle: " << area(r) << endl;
  cout << "Enter the radius and height of the sphere: ";
   cin >> r >> h;
   cout << "Area of the sphere: " << area(r, h) << endl;
 cout << "Enter the radius, height, and slant height of the cone: ";
   cin >> r >> h >> l;
   cout << "Area of the cone: " << area(r, h, l) << endl;
  return 0;  }
```

Q9 Write a C++ program to create a class Item with data members Item_Code, Item_Name, Item_Price. Write member functions to accept and display Item information also display number of objects created for a class. (Use Static data member and Static member function)

```
#include <iostream>
#include <string>
using namespace std;
class Item {    private:   static int count;
  int item_code;   string item_name;
        float item_price;
   public:  Item() {item_code = 0;
  item_name = "";   item_price = 0.0;
   count++; }  void accept() {
    cout << "Enter item code: ";
    cin >> item_code;  cin.ignore();
           cout << "Enter item name: ";
           getline(cin, item_name);
           cout << "Enter item price: ";
           cin >> item_price;  }
  void display() {
   cout << "Item code: " << item_code << endl;
 cout << "Item name: " << item_name << endl;
  cout << "Item price: " << item_price << endl;
}    static void show_count()
 cout << "Number of items created: " << count << endl;  }  };
int Item::count = 0;
  int main() {
 Item item1, item2;
 item1.accept();
  item2.accept();  item1.display(); item2.display();
  Item::show_count();  // Calling static member function
  return 0;  }
```

Q10 Implement a class Complex which represents the Complex Number data type. Implement the following operations: 1. Constructor (including a default constructor which creates the complex number0+0i). 2. Overloaded operator+ to add two complex numbers.

```
 #include <iostream>

using namespace std;

class Complex {   private:

        double real;  double imag;

   public:  Complex() {

            real = 0;  imag = 0;  }

Complex(double r, double i) {  real = r;

            imag = i;  }

   Complex operator+(Complex const &obj) {

     Complex res;  res.real = real + obj.real;

    res.imag = imag + obj.imag;   return res;  }

void display() {

   cout << real << " + " << imag << "i" << endl;

        } };  int main() {

    Complex c1(3, 4);    Complex c2(2, 5);

    Complex c3;

    c3 = c1 + c2;

    c1.display();

    c2.display();

    c3.display();  return 0;

}
```

Q11 Write a C++ program create a calculator for an arithmetic operator (+, -, *, /). The program should take two operands from user and performs the operation on those two operands depending upon the operator entered by user. Use a switch statement to select the operation.

```
#include <iostream>

using namespace std;

int main() {    char op;

    float num1, num2;

    cout << "Enter operator (+, -, *, /): ";

    cin >> op;

    cout << "Enter two operands: ";

    cin >> num1 >> num2;

    switch(op) {  case '+':

            cout << num1 << " + " << num2 << " = "
<< num1+num2 << endl;    break;

        case '-':

            cout << num1 << " - " << num2 << " = "
<< num1-num2 << endl;

            break;

   case '*':

            cout << num1 << " * " << num2 << " = "
<< num1*num2 << endl;

            break;

  case '/':

            if (num2 != 0)

                cout << num1 << " / " << num2 << " =
" << num1/num2 << endl;

            else

                cout << "Cannot divide by zero" <<
endl;

            break;


        default:

            cout << "Invalid operator" << endl;

            break;

    }  return 0;

}
```

Q12 Write a function in C++ to count and display the number of lines not starting with alphabet 'A' present in a text file "STORY.TXT". Example: If the file "STORY.TXT" contains the following lines, The roses are red. A girl is playing there. There is a playground. An aeroplane is in the sky. Numbers are not allowed in the password. The function should display the output as 3.

```
#include <iostream>

#include <fstream>

#include <string>

using namespace std;

int main() {

    ifstream inputFile("STORY.TXT");

    string line;   int count = 0;

 if (inputFile.is_open()) {

        while (getline(inputFile, line)) {

            if (line[0] != 'A') {   count++;  }

        }   inputFile.close();

        cout << "Number of lines not starting with 'A': " << count << endl;

    } else {  cout << "Error opening file" << endl;

    } return 0;  }
```

Q13 Create a C++ class for a student object with the following attributes—roll no, name, number of subjects, marks of subjects. Write member function for accepting marks and display all information of student along with total and Percentage. Display marklist with Use of manipulators.

```
#include <iostream>

#include <iomanip>

using namespace std;

class Student {

private:

    int rollNo;  string name;

    int numOfSubjects;

    int* marks;

public: Student(int rollNo, string name, int numOfSubjects) {

        this->rollNo = rollNo;

        this->name = name;

        this->numOfSubjects = numOfSubjects;

        this->marks = new int[numOfSubjects];

    }void acceptMarks() {

        cout << "Enter marks of " << name << " (roll no " << rollNo << ") for " << numOfSubjects << " subjects:\n";

        for (int i = 0; i < numOfSubjects; i++) {

            cout << "Subject " << i+1 << ": ";

            cin >> marks[i]; } }  void displayInfo() {

        int total = 0;

        cout << "Roll no: " << rollNo << endl;

        cout << "Name: " << name << endl;

        cout << "Number of subjects: " << numOfSubjects << endl;

        cout << "Marks: ";

        for (int i = 0; i < numOfSubjects; i++) {

            cout << setw(3) << marks[i] << " ";

            total += marks[i];  }

        cout << endl;

        cout << "Total marks: " << total << endl;

        cout << fixed << setprecision(2) << "Percentage: " << (float)total/numOfSubjects << "%" << endl;  }

    ~Student() {   delete[] marks;} };

int main() {  Student s(123, "John Doe", 5);

    s.acceptMarks();  s.displayInfo(); return 0;  }
```