Q1. Basic Class

Write a program in C++ to create a class Car with data members name and speed.

- Use a member function display() to print values.
- Create two objects and display their details.

```cpp
#include <iostream>
#include <string>
using namespace std;
class Car{
   public:
   string name;
    double speed;
    Car(string name,double speed){
        this->name = name;
        this->speed = speed;
    }
    void display(){
        cout<<"name : " << name << endl;
        cout<<"speed : "<< speed << endl;
    }
};
    int main(){
        Car c("Mahi",43.5);
        Car c1("Anuj",89.8);
        c.display();
        c1.display();
    }
```

Output:

```
PS C:\Users\mahir> cd "c:\Users\mahir\OneDrive\Desktop\Sublime\Ass3 C++\"
name : Mahi
speed : 43.5
name : Anuj
speed : 89.8
```

Q2. Rectangle (Area & Perimeter)
Create a class Rectangle with data members length and width.
● Add member functions to calculate area and perimeter.
● Read values from user and display results.

```
#include <iostream>
using namespace std;
class Rectangle
{
public:
    double length;
    double width;

    void input()
    {
        cout << "Enter length : ";
        cin >> length;
        cout << "Enter width : ";
        cin >> width;
    }

    double area() const
    {

        return length * width;
    }

    double perimeter() const
    {
        return 2 * (length + width);
    }

    void display() const
    {
        cout << "length : " << length << endl;
        cout << "width : " << width << endl;
        cout << "Area of rectangle : " << area() << endl;
        cout << "Perimeter of rectangle : " << perimeter() << endl;
    }
};
int main()
{
    Rectangle r;
    r.input();
```

```
    r.display();
    return 0;
}
```

Output:

```
PS C:\Users\mahir> cd "c:\Users\mahir\OneDrive\Desktop\Sublime\Ass3
Enter length : 5
Enter width : 7
Enter width : 7
length : 5
width : 7
Area of rectangle : 35
Perimeter of rectangle : 24
```

Q3. Student Details

Create a class Student with data members rollNo, name, and marks.

- Add member function input() to take values.
- Add function display() to print them.
- Create an array of 3 students and display all details.

```
#include <iostream>
#include <string>
using namespace std;
class Student
{
public:
    int rollno;
    string name;
    double marks;

    void input()
    {
        cout << "Enter rollno : ";
        cin >> rollno;
        cin.ignore();
        cout << "Enter name : ";
        getline(cin, name);
        cout << "Enter marks : ";
```

```
        cin >> marks;
    }

    void display()
    {
        cout << "roll no : " << rollno << "\n";
        cout << "name : " << name << "\n";
        cout << "marks : " << marks << "\n";
    }
};

int main()
{
    Student s[3]; // Array of 3 student objects
    cout << "-----Enter details of 3 students -----" << endl;

    // Taking input
    for (int i = 0; i < 3; i++)
    {
        cout << "\nStudent " << i + 1 << ":\n";
        s[i].input();
    }

    cout << "\n --- Displaying Student Details ---" << endl;

    // Display Details
    for (int i = 0; i < 3; i++)
    {
        s[i].display();
        cout << "\n";
    }
    return 0;
}
```

Output:

```
PS C:\Users\mahir> cd "c:\Users\mahir\OneDrive\Desktop\Sublime\A
-----Enter details of 3 students -----

Student 1:
Enter rollno : 1
Enter name : Mahi Rusia
Enter marks : 234

Student 2:
Enter rollno : 2
Enter name : Harshul Rusia
Enter marks : 34

Student 3:
Enter rollno : 3
Enter name : vedant
Enter marks : 56

 --- Displaying Student Details ---
roll no : 1
name : Mahi Rusia
marks : 234

roll no : 2
name : Harshul Rusia
marks : 34

roll no : 3
name : vedant
marks : 56
```

Q4. Bank Account

Create a class BankAccount with:

- Data members: accountNumber, balance.
- Functions: deposit(), withdraw(), displayBalance().
- Perform deposit and withdrawal operations using objects.

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

class BankAccount
{
private:
    int accno;
    double balance;
public:
    BankAccount(int accno, double balance)
    {
        this->accno = accno;
        this->balance = balance;
    }

    void deposit(double amount)
    {
        if (amount > 0)
        {
            balance += amount;
            cout << amount <<" "<<"deposited successfully"<<"\n";
        }
        else
        {
            cout << "Invalid deposit amount"<<"\n";

        }
    }

    void withdraw(double amount)
{
    if (amount <= 0) // check for negative or zero amount
    {
        cout << "Invalid withdrawal amount.\n";
    }
```

```
    else if (amount <= balance) // check if enough balance
    {
        balance -= amount;
        cout << amount << " withdrawn successfully.\n";
    }
    else
    {
        cout << "Insufficient Balance.\n";
    }
}


    void display()
    {
        cout << "Account No: " << accno<<"\n";
        cout << "Balance: " << balance<<"\n";
    }
};

int main()
{
    BankAccount B1(1, 300);
    B1.display();

    double depositAmount;
    cout << "\nEnter amount to deposit: ";
    cin >> depositAmount;
    B1.deposit(depositAmount);

    double withdrawAmount;
    cout << "Enter amount to withdraw: ";
    cin >> withdrawAmount;
    B1.withdraw(withdrawAmount);

    cout << "\nUpdated account details:\n";
    B1.display();
}
```

Output:

```
PS C:\Users\mahir> cd "c:\Users\mahir\OneDrive\Desktop\Subli
Account No: 1
Balance: 300

Enter amount to deposit: 2000
2000 deposited successfully
Enter amount to withdraw: 1500
Enter amount to withdraw: 1500
1500 withdrawn successfully.

Updated account details:
Account No: 1
Balance: 800
```

Q5. Employee Salary (Parameterized Constructor)
Write a C++ program to create a class Employee with data members id, name, and salary.
● Use a parameterized constructor to initialize values.
● Display employee details using a function.

```cpp
#include <iostream>
using namespace std;
class Employee
{
public:
    int id;
    string name;
    double salary;
    Employee(int id, string name, double salary)
    {
        this->id = id;
        this->name = name;
        this->salary = salary;
    }
    void display()
    {
        cout << "Employee id : " << id<<"\n";
        cout << "Employee  name :" << name<<"\n";
        cout << "Employee salary : " << salary<<"\n";
```

```
    }
};
int main()
{
    Employee e(101, "mahi", 10000);
    e.display();
}
```

Output:

```
PS C:\Users\mahir\OneDrive\Desktop\Sublime\Ass3 C++> c
Employee id : 101
Employee  name :mahi
Employee salary : 10000
```

Q6. Complex Number (Object as Argument)
Create a class Complex with data members real and imag.
● Add a member function add() that takes another Complex object and returns the result as a new
object.
● Display the sum of two complex numbers.

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

class Complex {
public:
    double real, imag; // real and imaginary parts

    // Member function to take input
    void input() {
        cout << "Enter real part: ";
        cin >> real;
        cout << "Enter imaginary part: ";
        cin >> imag;
    }

    // Function to display complex number
    void display() {
        cout << real << " + " << imag << "i" << endl;
    }

    // Function to add two complex numbers (object as argument)
```

```
    Complex add(Complex c) {
        Complex temp;
        temp.real = real + c.real;
        temp.imag = imag + c.imag;
        return temp;
    }
};

int main() {
    Complex c1, c2, sum;

    cout << "Enter first complex number:\n";
    c1.input();

    cout << "Enter second complex number:\n";
    c2.input();

    // Adding two complex numbers
    sum = c1.add(c2);

    cout << "Sum of complex numbers: ";
    sum.display();

    return 0;
}
```

Output:

```
PS C:\Users\mahir> cd "c:\Users\mahir\OneDrive\Desktop\Sublime\Ass3 C+
Enter first complex number:
Enter real part: 3
Enter imaginary part: 2
Enter second complex number:
Enter real part: 6
Enter imaginary part: 4
Sum of complex numbers: 9 + 6i
```

Q7. Library Management

Create a class Book with data members title, author, and price.

- Write functions to input and display details.
- Create an array of 5 books and print the book with the highest price.

```
#include <iostream>
#include <string>
using namespace std;
class Book
{
public:
    string title;
    string author;
    double price;
    void input()
    {
        cout << "Enter title : ";
        cin >> title;
        cout << "Enter author : ";
        cin >> author;
        cout << "Enter price : " ;
        cin >> price;
    }
    void display()
    {
        cout << "Title : " << title;
        cout << "Author : " << author;
        cout << "Price : " << price;
    }
};
int main()
{
    Book b[5];

    // for taking input
    for (int i = 0; i < 5; i++)
    {
        cout << "Enter details of books : " << i + 1 << endl;
        b[i].input();
    }
```

```
    float maxPrice = b[0].price;
    int index = 0; // to store which book has the maximum price
    for (int i = 1; i < 5; i++)
    {
        if (b[i].price > maxPrice)
            maxPrice = b[i].price;
        index = i; // highest value
    }
    cout << "Books with the highest price" << endl;
    cout << b[index].title << endl;
    cout << b[index].author << endl;
    cout << b[index].price << endl;
}
```

Output:

```
PS C:\Users\mahir> cd "c:\Users\mahir\OneDrive\Desktop\Sublime
Enter details of books : 1
Enter title : gyan
Enter author : mahi
Enter price : 45
Enter details of books : 2
Enter title : yang
Enter author : anuj
Enter price : 34
Enter details of books : 3
Enter title : jsol
Enter author : wrddddd
Enter price : 345
Enter details of books : 4
Enter title : wytfs
Enter author : wnks
Enter price : 23
Enter details of books : 5
Enter title : eje
Enter author : werr
Enter price : 56
Books with the highest price
eje
werr
56
PS C:\Users\mahir\OneDrive\Desktop\Sublime\Ass3 C++>
```