

Lab Manual – Association

Exercise 1:

Make a class **Student** that has following data members:

```
char name[50];  
char rollNo[8];  
float cgpa;
```

Students class provides a constructor with default arguments and a function **Print** that prints students name and roll number on screen in following format:

StudentName (RollNo.)

For Example, Aslam Baig (12L9356)

Do we need a Destructor for this class?

Exercise 2:

Write the following piece of code in your main function: it should create six students with information provided.

```
Student s1("12L1111", "Hashim Amla", 3.99);  
Student s2("13L1121", "Virat Kohli", 3.45);  
Student s3("13L1126", "Quinton de Kock", 2.98);  
Student s4("14L1361", "Joe Root", 2.99);  
Student s5("14L1124", "Martin Gupatil", 3.09);  
Student s6("15L1314", "Rohit Sharma", 3.19);
```

Exercise 3:

A Student **Society** has a president and five members from students. Make a class Society that has following private data members:

```
char name[50];  
Student* president;  
Student* members[5];
```

The Society class has a constructor with default arguments that takes the name of society as parameter. Why are we keeping Student pointers in Society class and what should the constructor do?

Exercise 4:

Write a member function of Society class **PrintInfo** that prints name of society and details of its members and president using the **Print** function of Student class. What should function do if some member does not exist?

Exercise 5:

Add following lines in your main function it should give following output:

```
Society sports ("Sports");  
  
sports.PrintInfo();
```

Output:

```
Society Name: Sports  
President:    Not Available  
Member 1:    Not Available  
Member 2:    Not Available  
Member 3:    Not Available  
Member 4:    Not Available  
Member 5:    Not Available  
Press any key to continue . . .
```

Why is it displaying Not Available in members' information? Because president and members pointers are currently pointing to NULL. We need to point these pointers to students' objects in order to create association between sports society and students.

Exercise 6:

Make a member function **AppointPresident** in Society class that takes a student object by reference and appoints it to president's position if the position is vacant and the cgpa of student is above 3.00. Display appropriate error message otherwise. Do you need to add Getters in Students class to accomplish this task?

Exercise 7:

Add following lines in your main function and verify the output:

```
sports.AppointPresident(s3);  
sports.AppointPresident(s1);  
sports.AppointPresident(s2);
```

Output:

```
...  
Quinton de Kock cannot be appointed as President. CGPA criteria not met.  
Hashim Amla has been appointed as President.  
Virat Kohli cannot be appointed as President. President position is NOT vacant.  
Press any key to continue . . .
```

Note: AppointPresident need to call GetName of student in order to print this message.

Exercise 8:

Make a member function **AddMember** in Society class that takes a student pointer and adds it in the list of members if there is any position vacant and displays the error message otherwise. Also the president cannot be in the list of society members. Also a student cannot be added as a member more than once.

Exercise 9:

Add following lines in your main function and verify the output:

```
cout << endl << endl << endl;
Student s7("15L1334", "Robert Elen", 2.19);
sports.AddMember(s3);
sports.AddMember(s2);
sports.AddMember(s3);
sports.AddMember(s1);
sports.AddMember(s4);
sports.AddMember(s5);
sports.AddMember(s6);
sports.AddMember(s7);
sports.PrintInfo();
```

Output:

```
...
Quinton de Kock has been added to members list successfully.
Virat Kohli has been added to members list successfully.
Quinton de Kock already exists in Members list.
President cannot be added in Members list.
Joe Root has been added to members list successfully.
Martin Guptil has been added to members list successfully.
Rohit Sharma has been added to members list successfully.
Robert Elen cannot be added to members list. Member position is NOT vacant.

Society Name: Sports
President: Hashim Amla (12L1111)
Member 1: Quinton de Kock (Roll No here)
Member 2: Virat Kohli (Roll No here)
Member 3: Joe Root (Roll No here)
Member 4: Martin Guptil (Roll No here)
Member 5: Rohit Sharma (Roll No here)
Press any key to continue . . .
```

You may keep currentMembersCount in your Society class. Should it be a static data member or non-static?

Note: This is responsibility of Student Class to print its name and roll number, use Print of Student class here.

Exercise 10:

Make a member function **SuspendMember** in Society class that takes a char* memberName as parameter searches that member from the list of members and removes that member from the list if found otherwise displays the error message. Should SuspendMember delete the Student?

Exercise 11:

Add following lines in your main function and verify the output:

```
cout << endl << endl << endl;  
sports.SuspendMember("abc");  
sports.SuspendMember("Martin Guptil");
```

Output:

...

abc NOT found in the Members list.

Martin Guptil has been suspended from Sports society.

Press any key to continue . . .

Print Society Information again to check if the student has been removed from the members list.

Exercise 12:

Add following lines of code in your main function and test the result.

```
cout << endl << endl << endl;  
  
Society ieee("IEEE");  
  
ieee.AppointPresident(s5);  
  
ieee.PrintInfo();
```