MIDDLE EAST TECHNICAL UNIVERSITY, NORTHERN CYPRUS CAMPUS
CNG242 Programming Language Concepts – Assignment 2 – C++

A Membership Management System
Deadline: 29 April 2016, Friday 23:55

Learning Outcomes

On successful completion of this assignment, a student will:

- Write a C++ program that utilizes multiple classes;
- Appreciate the usefulness of reusing code;
- Develop a class based on storing a collection of objects using arrays.

Purpose

In this homework you will implement several C++ classes for a membership management system. We are going to assume that a university facilitate memberships of academic staff, administrative personnel and students for a professional association called Institute of Computer Scientists and Engineers (ICSE). Although the name of the association sounds like it is specialised only for computing discipline, it welcomes all the professionals with engineering background, who make use of computing facilities.

This assignment aims to create a record book for different types of memberships within the university and it will allow the users to search for the details of memberships as well as members. Please note that the memberships are annual. However, in case of consecutive renewals for academics, there are some discounts as explained below.

*****You should put each of the class declaration in a separate header file and implement the member methods in separate .cpp files.**

Implementation Details

In order to implement this system you will need to implement at least the following classes with the attributes, constructors, destructors, copy constructors and other member functions. Please note that you may need to add more data members and implement more functions.

PART ONE: C++ CLASSES

Member

For each member, you should store:

- First name
- Last name
- Telephone number
- Email address

For each member, you should implement the following member functions:

- A default constructor
- A constructor to initialise First name, Last name, Telephone number, Address, and Email address.
- A copy constructor
- An assignment operator
- A destructor
- Getter and setter functions for each of the member variables
- A function to print all these details

Academic Member

Each academic member is a member with the following additional properties.

- Employee ID
- Title (Assistant, Instructor, Doctor, Assistant Professor, Associate Professor or Professor)
- Date membership initiated

For each academic member, you should implement the following member functions:

- A default constructor
- A constructor to initialise all of the member variables
- Getter and setter functions for each of the member variables
- A function to compute membership fee

The annual membership fee for academics with various titles is $200. However, each academic member gets 10% discount for each year (using the date membership initiated) they renew the membership. The discount cannot be more than 50%. In other words, all academic members have to pay at least $100.

Student Member

Each student member is a member with the following additional properties.

- Student ID
- Student Member type (Full or Competitor)
- Number of competitions to participate for one year

For each student member, you should implement the following member functions:

- A default constructor
- A constructor to initialise all of the member variables
- Getter and setter functions for each of the member variables
- A function to compute membership fee

The annual membership fee for each full student is $50. However, students can enrol as competitor which means that they are not interested in the facilities provided by the membership. Instead they only want to compete in competitions organised by ICSE. In this case, they need to buy credits per competition they would like to participate which is $10 each.

Administrative Member

Administrative personnel can also be interested in ICSE membership since some of the magazines they provide on state of the art technologies and for gaming in particular are quite popular. Each administrative member is a member with the following additional properties.

- Employee ID
- Admin type (Manager or Secretary)
- Social member type (All Magazines or Game Geek)

For each administrative member, you should implement the following member functions:

- A default constructor
- A constructor to initialise all of the member variables
- Getter and setter functions for each of the private member variables
- A function to compute membership fee

The annual memberships for all magazines for admin staff are $100, membership only to the magazine "Game Geek" is $25.

ICSE Teams

This year the university would like each member to be a part of a team with various genres. Each team is established from a number of academics, students, and administrative staff. Team class holds the following details

- Team Name (e.g. IoT, AI, WSNs, WirelessCom, EmbeddedSys.)
- Team Leader: Academics who are Associate Professor or Professor have their own teams. In other words each team can have at most one Associate Professor or Professor.
- Team Members: Team members consist of academics, students and admins.

For each team, you should implement the following member functions:

- A default constructor

- A constructor to initialise the team name.
- Getter and setter functions for the team name and team leader.
- A function to add a new team member
- A function to list the team leader and the members
- A function to calculate annual cost of membership for each team
- A function to search if a given employee (with Employee ID) or student (with student ID) is in the team. If in the team, you need to print his/her details.

University ICSE Memberships
Using the classes mentioned above create a University_ICSE_Memberships class that maintains various teams. This class should be able to:

- Add a new team
- Search for the details of a given employee (with Employee ID) or student (with Student ID). Hint: First you need to ask if the searched member is student, academic staff or administrative staff, then you need to check from the array of teams)
- Print the team leader and members of each team in the university.
- Print the team leader and members of a given team (with Team Name)
- Print the annual cost of all memberships.
- Print the annual cost of a given team (with Team Name)

PART TWO: A UNIVERSITY ICSE MEMBERSHIP MANAGEMENT SYSTEM

You need to implement a command line interface for the University ICSE Membership management system. This system is not for storing the details of previous memberships. It is mainly for entering all the details at the same time to calculate the total annual cost based on the current date.

This management system should provide the following menu:

1. Create a team
2. Search a member by ID
3. List all members
4. List total annual cost
5. List all members of a given team
6. List total annual cost of a given team
7. Exit

Grading Policy

Item	Marks (Total 20)
Member Class	3
Academic Class	2
Student Class	2
Admin Class	2
Team Class	4
University ICSE Memberships Class	3
University ICSE Membership Management System	4

Submission Rules

- **You need to compress your C++ project (including header files and C++ implementation files)**
- **You need to submit the compressed file to ODTU-CLASS.**
- **Header files should contain class definitions with prototypes only, no function implementation.**
- **You are not allowed using std::string for this assignment.**
- **You are not allowed using vectors for this assignment.**

Sample Run

```
Welcome to University ICSE Membership
Management System!

1. Create a team
2. Search a member by ID
3. List all members
4. List total annual cost
5. List all members of a given team
6. List total annual cost of a given team
7. Exit
Your choice: 1

Enter Team Name: WirelessCom
Enter the details of the team leader
Employee ID: 10
First name:  John
Last name:  Brown
Telephone number:  1010
Email address: jb@uni.uk
Academic Title (1) Professor (2) Associate
Professor
Date membership initiated: 19.04.2016
The team leader has been added to the
team successfully!

Would you like to add a new member to the
team (Y/N): Y
1. Academic Staff
2. Student
3. Administrative Staff
Select type: 1

Enter the details of the team member:
Employee ID: 120
First name: Peter
Last name: Chech
Telephone number: 2224
Email: pc@uni.uk
Academic Title (1) Assistant Professor (2)
Doctor (3) Instructor (4) Assistant: 1
Date membership initiated: 19.04.2016
The academic member has been added to
the team successfully!

Would you like to add a new member to the
team (Y/N): Y
1. Academic Staff
2. Student
3. Administrative Staff
Select type: 3

Enter the details of the team member:
Employee ID: 140
First name: Philip
Last name: Lahm
Telephone number: 2226
Email: pl@uni.uk
Admin Type: (1) Manager (2) Secretary: 1
Social member type (1) All Magazines (2)
Game Geek Monthly: 1
The administrative member has been added
to the team successfully!

Would you like to add a new member to the
team (Y/N): Y
1. Academic Staff
2. Student
3. Administrative Staff
Select type: 2

Enter the details of the team member:
Student ID: 50
First name: Tyrion
Last name: Fields
Telephone number: 4342
Email: tf@uni.uk
Student Member type (1) Full (2) Competitor:
1
The student member has been added to the
team successfully!

Would you like to add a new member to your
department (Y/N): N
The team has been added to the university
successfully!

1. Create a team
2. Search a member by ID
3. List all members
4. List total annual cost
5. List all members of a given team
6. List total annual cost of a given team
7. Exit
Your choice: 2
Please indicate his/her role (1) Academic
Staff (2) Student (3) Administrative Staff: 1
Enter Employee ID: 10

The member has been found in the
WirelessCom team 1. The details are listed
below
10 John Brown 1010 jb@uni.uk

1. Create a team
2. Search a member by ID
3. List all members
```

```
4. List total annual salary
5. List all members of a given team
6. List total annual salary of a given team
7. Exit
Your choice: 3

Team: WirelessCom
Team Leader:
10 John Brown 1010 jb@uni.uk

Academic Staff Members:
120 Peter Chech 2224 pc@uni.uk

Student Members:
50 Tyrion Fields 4342 tf@uni.uk

Administrative Members:
140 Philip Lahm 2226 pl@uni.uk

1. Create a team
2. Search a member by ID
3. List all members
4. List total annual cost
5. List all members of a given team
6. List total annual cost of a given team
7. Exit
Your choice: 4
Total annual cost is $550

1. Create a team
2. Search a member by ID
3. List all members
4. List total annual cost
5. List all members of a given team
6. List total annual cost of a given team
7. Exit
Your choice: 5
Team Name: WirelessCom

Team: WirelessCom
Team Leader:
10 John Brown 1010 jb@uni.uk

Academic Staff Members:
120 Peter Chech 2224 pc@uni.uk

Student Members:
50 Tyrion Fields 4342 tf@uni.uk

Administrative Members:
140 Philip Lahm 2226 pl@uni.uk

1. Create a team
2. Search a member by ID
3. List all members
4. List total annual cost
5. List all members of a given team
6. List total annual cost of a given team
7. Exit
Your choice: 6
Team Name: WirelessCom
Total annual cost for the WirelessCom team is
$550

1. Create a team
2. Search a member by ID
3. List all members
4. List total annual cost
5. List all members of a given team
6. List total annual cost of a given team
7. Exit
Your choice: 7
Goodbye!
```