BLG252E OBJECT ORIENTED PROGRAMMING

Homework 1

Due Date: Wednesday, March 30, 2016, 23:00

[Hard deadline. Ninova HW submission closes automatically. Late homeworks will NOT be accepted.
Submit via Ninova, submission via e-mail will not be graded.]

Guidelines:

Academic dishonesty including but not limited to cheating, plagiarism, collaboration is unacceptable and subject to disciplinary actions. Any student found guilty will have -100 penalty points.

CHEATING: This is not a group assignment. It should be done individually. When a student receives information from another person about a program, it is considered cheating when the information is enough to precisely describe the code in a nontrivial part of the program. The most common example of cheating occurs when a student copies all or part of a program from another student and then changes the names of some of the program variables and functions. Your submissions will be checked using plagiarism detection software. If cheating is discovered, a report will be made recommending a course grade of "VF".

- Make sure you write your name and number at the top of all the files of your project. Otherwise, you will lose 5 points.
- Your program **should be compiled and run on Linux** environment using **g++**. (test your program using ssh) Include all necessary header files to your code. Do not use precompiled header files and Windows specific header files and functions. Do not use STL commands.
- Use comments wherever necessary in your code to explain what you did.
- Be careful with the methods/attributes which are supposed to be constant, static, private.
- Handle exceptions and display error messages wherever necessary.
- Submit a working copy at least half an hour before the due time to avoid last minute rush accidents.
- If you have any question about the homework, contact with research assistant Mustafa Esengün via email (esengun@itu.edu.tr).

PROBLEM DESCRIPTION

This assignment aims to create an address book of the instructors in the Computer Engineering department. This address book will maintain the details of instructors and it will allow the users to search for details of instructors by using different search parameters.

Implementation Details: First of all you need to develop a class called "**Instructor.h**" which will hold the following details about an individual instructor in the department:

- Title (Prof., Dr., Mr., Ms., etc.)
- First name;
- Last name;
- Telephone number;
- Room number;
- User name;
- Email address.
- Courses (a list of courses that are taught by this instructor, for the name of courses use departmental convention, e.g., BLG123).

It should be possible to create an object of the Instructor class by providing the above details in the form of parameters to a constructor. Your class should also have constructors that support instantiating this class with partial information. Make sure that your instance variables have the "private" access modifier and ensure that you provide both getter and setter methods. Make sure also that you include a method that displays the information about an instructor in a suitable textual format. Note that an instructor may have multiple courses, use an array to store courses for each instructor. Using the Instructor class defined above, create an "InstructorList" class that maintains a list of Instructor objects in an array. This class should be in a file called "**InstructorList.h**". This class should include the following methods:

- Add new instructor which allows the user to add a new instructor and his/her detail. This method should ensure that there is no duplicated instructors in the InstructorList. If the array is already full, then this method should be able to extend the size of the array to store the information about this new instructor. Your interface should ask the user to enter each field (Title, First name,…) of the Instructor class step by step.
- Delete an instructor from the existing list of instructors. Your interface should ask to user the first and the last name of the instructor to be deleted. You should check whether this functionality works or not by printing all the instructors.
- List all the instructors; simply print all the instructors in a readable format.
- Search for details of a given instructor: it should be possible to search for an instructor using any of the following fields: first name, last name, telephone number, room number.
- This class should also have two constructors, one that creates an array with a given size, and one that creates an array with a default size.

Create a class called "AddressBook" that has a main method and creates an instance of the InstructorList class. This class should be written in a file called "**Addressbook.cpp**". This AddressBook class should hold the user interface of the application that will be displayed on the

command line. In this class, you should create an instance of InstructorList class and 3 instances of Instructor class by using the details of the Computer Engineering instructors from this page: <http://www.bb.itu.edu.tr/calisanlar/ogretim-uyeleri>. If some information is missing for instructors, you can just populate them with test data. You should choose at least 3 instructors with their full details. After that, you should handle the flow of the program as shown in the Example Output section.
Your AddressBook command line interface should provide the following options:

- Add a new instructor;
- Delete an instructor;
- List all the instructors (should list all their details);
- Search by Name;
- Search by Surname;
- Search by Telephone Number;
- Search by Room Number;

Assessment Criteria

This assignment will be graded as follows:

Aspect	**Marks (Total 100)**
Instructor class	25
InstructorList class	25
AddressBook class	10
Add a new instructor	10
Delete an instructor	5
List all instructors	5
Search by name	5
Search by surname	5
Search by telephone number	5
Search by room number	5

Example Output

Your program should give the same output as the one shown below, which shows outputs of each of the functionalities.

Submission Components

You should submit below files as a **.zip** folder through ninova.

1. **Instructor.h**
2. **InstructorList.h**
3. **Addressbook.cpp**

```
1. Add a new instructor
2. Delete an instructor
3. List all instructors
4. Search by Name
5. Search by Surname
6. Search by Telephone Number
7. Search by Room Number
8. Exit
Enter the number for operation you want to perform:
1
Enter the title:
Prof.
Enter the First Name:
Hasan
Enter the Last Name:
Hüseyin
Enter the Telephone Number:
1234
Enter the Room Number:
9876
Enter the User Name:
hhasan
Enter the Courses:
BLG123
New instructor is added!
Do you want to perform another operation?(Y/N)
Y
1. Add a new instructor
2. Delete an instructor
3. List all instructors
4. Search by Name
5. Search by Surname
6. Search by Telephone Number
7. Search by Room Number
8. Exit
Enter the number for operation you want to perform:
2
Enter the first name of the record to be deleted:
Hasan
Enter the last name of the record to be deleted:
Hüseyin
Record has been deleted!
Do you want to perform another operation?(Y/N)
Y
```

```
1. Add a new instructor
2. Delete an instructor
3. List all instructors
4. Search by Name
5. Search by Surname
6. Search by Telephone Number
7. Search by Room Number
8. Exit
Enter the number for operation you want to perform:
3
----------List of all Instructors in Computer Engineering of ITU----------

Title: Assist. Prof. Dr.
First Name: Gülşen
Surname: Eryiğit
Telephone Number: 1234
Room Number: 3303
User Name: gulsenc
Email: gulsenc@itu.edu.tr
Courses: [BLG123, BLG456, BLG789]
---------------------------------------
Title: Assist. Prof. Dr.
First Name: Sanem
Surname: Kabadayı
Telephone Number: 2994
Room Number: 3311
User Name: kabadayi
Email: kabadayi@itu.edu.tr
Courses: [BLG456, BLG321, BLG357]
---------------------------------------
Title: Assist. Prof. Dr.
First Name: Hatice
Surname: Köse
Telephone Number: 2901
Room Number: 3315
User Name: kose
Email: kose@itu.edu.tr
Courses: [BLG357, BLG789, BLG246]
---------------------------------------
Do you want to perform another operation?(Y/N)
Y
```

```
1. Add a new instructor
2. Delete an instructor
3. List all instructors
4. Search by Name
5. Search by Surname
6. Search by Telephone Number
7. Search by Room Number
8. Exit
Enter the number for operation you want to perform:
4
----------Search Results----------
Enter the first name:
Hatice
Title: Assist. Prof. Dr.
First Name: Hatice
Surname: Köse
Telephone Number: 2901
Room Number: 3315
User Name: kose
Email: kose@itu.edu.tr
Courses: [BLG357, BLG789, BLG246]
Do you want to perform another operation?(Y/N)
Y
1. Add a new instructor
2. Delete an instructor
3. List all instructors
4. Search by Name
5. Search by Surname
6. Search by Telephone Number
7. Search by Room Number
8. Exit
Enter the number for operation you want to perform:
5
----------Search Results----------
Enter the last name:
Kabadayı
Title: Assist. Prof. Dr.
First Name: Sanem
Surname: Kabadayı
Telephone Number: 2994
Room Number: 3311
User Name: kabadayi
Email: kabadayi@itu.edu.tr
Courses: [BLG456, BLG321, BLG357]
Do you want to perform another operation?(Y/N)
Y
```

```
1. Add a new instructor
2. Delete an instructor
3. List all instructors
4. Search by Name
5. Search by Surname
6. Search by Telephone Number
7. Search by Room Number
8. Exit
Enter the number for operation you want to perform:
6
----------Search Results----------
Enter the telephone number:
1234
Title: Assist. Prof. Dr.
First Name: Gülşen
Surname: Eryiğit
Telephone Number: 1234
Room Number: 3303
User Name: gulsenc
Email: gulsenc@itu.edu.tr
Courses: [BLG123, BLG456, BLG789]
Do you want to perform another operation?(Y/N)
Y
1. Add a new instructor
2. Delete an instructor
3. List all instructors
4. Search by Name
5. Search by Surname
6. Search by Telephone Number
7. Search by Room Number
8. Exit
Enter the number for operation you want to perform:
7
----------Search Results----------
Enter the room number:
3311
Title: Assist. Prof. Dr.
First Name: Sanem
Surname: Kabadayı
Telephone Number: 2994
Room Number: 3311
User Name: kabadayi
Email: kabadayi@itu.edu.tr
Courses: [BLG456, BLG321, BLG357]
Do you want to perform another operation?(Y/N)
N
```