**CS211 - Project**

**Problem: Staff Directory**

You have a client who wants to store a simple staff directory in her computer that she can use for storage and retrieval of names and office numbers. She has a data (text) file that contains the names and office numbers of her teachers. She wants to be able to insert new names and office numbers, change the number for an entry, and retrieve selected names. She also wants to save any changes in her data (text) file.

**INPUTS**

**Initial staff directory:** Each name and office number will be read from separate lines of a text file. The entries will be read in sequence until all entries are read.

**Additional entries:** Each entry is typed by the user at the keyboard when requested.

**OUTPUTS**

**Name and office numbers:** The name and number of each person selected by the user are displayed on separate output lines.

**Updated staff directory:** Each name and office number will be written to separate lines of a text file. The entries will be written in sequence until all entries are written.

**Part 1. Array Implementation**

* Create a class called “StaffEntry” which can be used to store the name and office number of ONE staff.
* Create a class called “MyArrayList” which stores list of “StaffEntry” objects. Use an array of size (MAXSIZE = 1000) to hold the list.

**Notes:**

Add appropriate data fields to the class “MyArrayList”,

Add all of the required operations to the class “MyArrayList”. For example, insert, linearSearch, binarySearch, insertionSort, heapSort, … .

* Create an appropriate interface that use the class “MyArrayList” to solve the staff directory problem defined above.

**Part 2. Linked List Implementation**

* Create a class called “MyLinkedList” which stores list of “StaffEntry” objects. Use a linked list (SLL or DLL) hold the list.

**Notes:**

Use the class “StaffEntry” which you have created in Part1.

Add appropriate data fields to the class “MyLinkedList”,

Add all of the required operations to the class “MyLinkedList”. For example, insert, linearSearch, binarySearch, insertionSort, heapSort, … .

* Create an appropriate interface that use the class “MyLinkedList” to solve the staff directory problem defined above. You can also use the same interface that has been used in Part 1, but this time with the class “MyLinkedList”.

**Instructions**

**Software Platform:** NetBeans

**Submission:** This project to be performed by a group of students (2-3 students each), and any identical submissions will be regarded as plagiarism attempt and all will be given zero marks.

It is required to submit for each part of the project (before the due date) the following:

* A copy of the project on a CD or you can upload the softcopy of the project on the blackboard.
* A Hard copy document that includes both an interface (as a picture) and the code for the other classes of the project.

Best Wishes

Dr. Mohammed Kayed