**Data Structures BS (CS, SE)-III**

**Fall 2019**

**Assignment 1**

**Dated: 22nd August 2019**

**Deadline: 29th August 2019 Instructor: Dr. Sajid Khan**

**Dear Students,**

**You have to send the solution of this assignment to the following email address**

[assignmentscollectiondrsajid@gmail.com](mailto:assignmentscollectiondrsajid@gmail.com)

**The subject of email should be YourName\_Section\_DSAssignmant1\_Fall19**

1. This task is based upon the relationship between a teacher, department and university. You need to create three classes named **Teacher**, **Department** and **University**. Those classes should have the following members
   1. **Teacher:**   
      **private members**  
      Name, email, employeeId, gender, designation, qualification.  
      **Public methods**  
      Funtions
      1. **getTeacherInput** that gets values of all the member variables of class teacher.
      2. **displayInfo** that display all the information of that person.
      3. **compareByName** that takes a string named compName as input and return true if the object that represents a teacher have Name equal to compName or not. This function will be used in Classes named **Department** and **University**.
      4. **compareByID** that takes an integer named compID as input and return true if the object that represents a teacher have ID equal to compID or not. This function will also be used in Classes named **Department** and **University**.
      5. **ChangeDesignation** that changes designation of employee.
      6. **ChangeQualification** that changes qualification of an employee.
   2. **Department:**

**Private members**  
DepartmentName, a pointer of type Teacher named TeachersInfo, an object named HOD of type Teacher, MaxDeptSize and currSize of type integers.  
**Public methods:**

* + 1. A **constructor** that takes a string named DeptName and store it in DepartmentName, an integer named DeptSize, stores it in MaxDeptSize and create a dynamic array of type Teacher of size MaxDeptSize. It should also reset the variable named currSize to 0.
    2. A **destructor** that when called, should delete the dynamic array of objects of Teacher of size MaxDeptSize.
    3. **ResetDepartment** that destroy the dynamic array, sets currSize to zero and reassign value to MaxDeptSize after getting it from user and create a new dynamic array.
    4. A function named **TerminateEmployee** that takes id of an employee as input and removes his/her data from the dynamic array.
    5. A function named **AddEmployee** that adds employee to the dynamic array after checking whether dynamic array have some space or not.
    6. A function named **searchEmpById** that takes id of an employee and display his/her complete details.
    7. A function named **searchEmpByName** that takes name of an employee and display his/her complete details.
    8. **PromoteEmployee** that takes an employee id as input and promote him/her to the next scale.
    9. **ChangeQualification** that just calls the **ChangeQualification** function of Teacher class in case if employee completed another degree program.
  1. **University:**

**Private Members:**

UniName, location, maxDepartments, currUniSize, VCName, RegistrarName, pointer named UniDepartments of type Department.

**Public methods:**

* + 1. **Constructor** that takes assign name to university as well as create a dynamic array of maxDepartments of type Department using pointer named UniDepartments. It should also set currUniSize to 0.
    2. **Destructor** that destroy the dynamic array.
    3. **ResetUni** that destroy the dynamic array, sets currUniSize to zero and reassign value to maxDepartments after getting it from user and create a new dynamic array.
    4. A function named **RemoveDept** that takes name of a department as input and removes its data from the dynamic array.
    5. A function named **AddDept** that adds a new department to the dynamic array after checking whether dynamic array have some space or not.
    6. A function named **dispDeptFaculty** that takes name of a department and display the information of all faculty members.