Assignment #1 – 207 - **(Employee Management System)**

Part #1

- You need to implement a base class to represent the *Employee.*
- *Employee* class contains information about the employee such as *Full_Name* (string), *Email* (string), *Phone* (string), *Position* (string), and *Salary* (double), they're private.
- *Employee* class contains a method to display the employee's information, method name is *display_employee_info(),* to print the attributes of the *Employee* in a pretty way.
- Implement all required setters() and getters() for all attributes.
- You must implement a parameterized constructor inside the *Employee* class. This constructor will take Full_Name, Email, Phone, Position, and Salary as parameters, to set the attributes with the values received in the constructor, so that you can define a new Employee.

Part #2

- You must implement a Singly Linked List of type *Employee* using the concept of **Generic (You will find helpful code tips on Moodle)**, with name *EmployeeList* this linked list should contain the following methods, insert a new employee at the end of the list, edit employee's salary, delete an employee from the list, **print all employees in the system.**
- You must implement a Node class, which should be generic as well. (Look at the helpful code tips).
- The insert method should check if an employee already exists with the same *email*, if exists it should display that the employee already exists, you don't have to add that employee. If the employee does not exist, you will add that employee to the list, method name is *insert_employee()*.
    - ***HINT:*** *bool insert_employee(Template e)*
- The edit employee's salary method will take as a parameter the *email* and the new salary, you need to search for the employee using the email, then update the salary with the new one passed in the parameters, you need to handle any exception case regarding the salary, method name is *edit_salary()*.
    - ***HINT:*** *bool edit_salary(double new_salary)*
- The delete method will delete an employee by taking the *email* as a parameter. If the employee exists you should delete it from the list, else you will show/display that the employee doesn't exist, method name is *delete_employee()*.
    - ***HINT:*** *bool delete_employee(string email)*

Part #3

- To run the program the main method should print a welcome message to the user, then a menu of options as follows:
    - 1- Insert a new employee.
    - 2- Edit employee's salary.
    - 3- Delete an employee.
    - 4- Print all employees.
    - 5- Close the program.
- The user will choose an option by typing for example 2, so you will call the edit salary method, it will take the email and the new salary. And so on.
- You need to handle if the user wants to choose another option and so on.

Part #4: Bonus

- Implement a method to update any one of the attributes of the employee, the method will be in the linked list class as well, the method name is *update_info(),* the method takes the email as a parameter, check if the employee exists, if so, you will ask the user for what attribute he/she wants to change and change it respectively.
    - ***HINT:*** *bool update_info(string email)*
    - *Inside the method you will ask for which attribute the user wants to change.*

IMPORTANT

- **Cheating will be graded zero.**
- You need to handle any exception case throughout the program.
- **Deadline is in 2 weeks**
- **You will submit a file .cpp with your id (example: 210000000.cpp)**
- **GRADING (10 Total):**
    - Part #1 => 2 Grades
    - Part #2 => 5 Grades
    - Part #3 => 2 Grades
    - Code quality and correctness => 1 Grade
    - Part #4 (Bonus) => 3 Grades (***Bonus***)
    - ***The final decision on grades will be made during the discussion.***