Assignment 4 – Strings/User-Defined Classes

Deadline: **Sunday Nov. 10 by 11:59pm.**
Type: **Group Assignment**
Weight: **This assignment is worth 5% of your final grade**

Notes:
- Please do not submit exe files
- All submissions must be done through Moodle

Marking Scheme:
- Program correctness (80%)
- Program clarity (output format, comments, completeness, readability) (20%)

Questions:

1. **(25 marks)** We want to create a class called **Edevice**, which represents the electronic devices of company. A device is defined with the following attributes: category (**string**) (The possible values are" Smartphone", "Tablet", "Laptop" and "Smartwatch"), model number (**int**), color (**string**), status (**boolean**) (true for new and false for used), year built (**int**), price (**double**). The member functions of the class **Edevic** must perform the following operations:

- Return the model number
- Return the category
- Return the departure date
- Modify the status
- Modify the color
- Modify the price
- Return the year built
- Return the status

To test your class, you need to create, a driver, which is the cpp file that contains the main function. Let's call this, **testdevice.cpp**. In the main function, you should prompt the user to enter information about two devices, create two objects of the class **Edevice** with the information entered by the user, and finally, test the member functions of the class.

Deliverables:
- A file called **device.h** that contains the specification of the class.
- A file called **Edevice.cpp** that contains the implementation of the member functions of the class.
- A file called **testdevice.cpp** that contains the main function.

2. **(25 marks)** Define a class called **"House"**, that represents the information of a house. A House is defined with these attributes: age (**int**), type (**string**) (Detached, Semi-Attached, Attached), rooms (**int**) and cost (double). Functions of the **House** class must perform the following operations:

- Return the house age
- Modify the house age
- Return the house type
- Return the number of rooms
- A function called estimatePrice() that returns cost of a house based on type and age. An attached house, costs $100,000, appreciates 1% every year in first five years and 2% every year afterwards. A Semi-detached house, costs $150,000, appreciates 2% every year in first five years and 3% every year afterwards. A Detached house, costs $200,000, appreciates 2% every year in first five years and 2% every year afterwards.

Test your class by prompting the user to enter information about two different houses. Create two objects of the class **House** with the information entered by the user, and finally, test the member functions of the class. Also calculate the estimated price of houses given type and age (include 1 attached and 1 detached)

Deliverables:

- A file called **house.h** that contains the specification of the class.
- A file called **house.cpp** that contains the implementation of the member functions of the class.
- A file called **testhouse.cpp** that contains the main function.

3. **(25 marks)** We want to create a class called **Employee**, which represents the employees of a department at the univerity. An employee is defined with the following attributes: employee id (**int**), first name (**string**), last name (**string**), date of birth (**string**), address (**string**), year hired (**int**), salary (**double**) and telephone (area code (**int**) and 7-digit telephone number(**string**). The member functions of the class **Employee** must perform the following operations:

- Return the employee id number.
- Return the first name of the employee
- Modify the first name of the employee
- Return the last name of the employee
- Modify the last name of the employee
- Return the hired year of the employee
- Return the full name, i.e., first name and last name
- Return the date of birth
- Modify the date of birth
- Return the salary of the employee
- Modify the salary of the employee
- Return the address of the employee
- Modify the address of the employee

- Return the telephone number
- Modify the telephone number
- Return true if two given employees have the same last name. Return false otherwise
- Return true if two employees have the same salary or they were hired on the same year. Return false otherwise

Test your class by prompting the user to enter information about two particular employees. Create two objects of the class **Employee** with the information entered by the user, and finally, and test the member functions of the class.

Deliverables:

- A file called **employee.h** that contains the specification of the class.
- A file called **employee.cpp** that contains the implementation of the member functions of the class.
- A file called **testemployee.cpp** that contains the main function.

4. **(25 marks)** We want to create a class called **Department**, which represents the information of a department at the university. A department is defined with the following attributes: identification number (**string**), name (**string**), department history (**string**), and a list of employees worked in the department (array of type **Employee** from previous question). You can assume that a department cannot have more than 25 employees. The member functions of the class **Department** must perform the following operations:

- Return the department identification number
- Return the department name
- Modify the department name
- Return the department history
- Modify the department history
- Add a new employee to the department
- Remove one employee from a department
- Search if an employee with a certain employee id works in the department
- Output list of employees
- Output the number of employees

Test your class by prompting the user to enter information about one of the university department and five employees. Create the department object and five objects of the class Employee. Assign the employees to the department**.** Test the member functions of the class.

Deliverables:

- A file called **employee.h** that contains the specification of the class.
- A file called **employee.cpp** that contains the implementation of the member functions of the class.
- A file called **department.h** that contains the specification of the class.
- A file called **department.cpp** that contains the implementation of the member functions of the class.
- A file called **testdepartment.cpp** that contains the main function.