Object Oriented PROGRAMMING (CS217)

ASSIGNMENT-3

Instructions

You are free to consult each other for verbal help. However **copying or sharing the code with each other will not only result into the cancellation of the current assignment, it may impact your grade in all the assignments and exams as well.**

Note: You will use 3 file structure from now onwards.

Deadline: 21/03/2020

1. Create a class named 'Book' having following private member variable
Id (Randomly generate in constructor), Name, Author, Edition, DateofPublish.
All the books belong to a single library.

1.1. Write appropriate setter/getter for each of the above member variables.
1.2. Write appropriate constructors and destructor.
1.3. Use shallow/deep copy (better one) to sort the array of books according to their publish date.
1.4. Display the details of most recently published 5 books.
1.5. Dynamically allocate an array of objects.

---

2. It's time to help your academic officer to automate their Registration Process. Create a class student having all essential attributes to store the record of a student. The class should maintain a record of total number of students enrolled.

2.1. The registration no. of a student, once assigned, cannot be changed. (Assigned on the time of registration(object creation))
2.2. Each student can register at most 5 and at least 3 courses.
2.3. Registered courses can be dropped if that course has been registered for 5 days.
2.4. Students record (e.g. CNIC, Address etc.) can be updated.
2.5. Your program should be menu driven having following options
- **2.5.1.** Register a Student
- **2.5.2.** Register Courses
- **2.5.3.** Drop Courses
- **2.5.4.** Update Record
- **2.5.5.** Delete a Student
- **2.5.6.** Search a Student (Should return an object of Student Class)
- **2.5.7.** Get a list of students (Studying a specific course together)
- **2.5.8.** Total count of Students Enrolled

2.6. Write separate functions for each of the above operations.

---

3. Create a class **Employee** having private attributes id (constant), name, salaryPerHour, monthlyWorkedHours, taxPaid, yearofJoining. Write a menu driven program to perform following tasks, each task governed by a specific function.

3.1. Update Hours of an employee (Admin updates the hours of an employee by end of the day).

3.2. Change Salary of an Employee

3.3. Calculate monthly Salary of an employee

- **3.3.1.** 10% tax is deducted from the salary and is automatically added to the taxPaid.
- **3.3.2.** 1000 is deducted from salary as health insurance fee.
- **3.3.3.** A senior employee (working in the company for more than 8 years) gets 10% additional bonus after deduction.
  Note: Each operation should have its own function. E.g. isSeniorEmployee();
- **3.3.4.** Generate a Receipt of the Salary.

---

4. Create a small chat application. Each user has unique id(constant), name, contactNo. Each user can send/receive maximum 10 messages. Your Program should keep a record of sent and received messages of each user. If message memory gets full(10), user need to empty his message box to send/receive new messages.
Your Program should be able to perform following tasks, each govern by a function

4.1. Send a Message

4.2. View Received Messages

4.3. Empty Message Box

4.4. Update info (Name, ContactNo)

4.5. Memory Check (How much message box is empty)

Use setter/getter for each of the attribute.

If receiver's message box is full, sender should get a warning that Recipient can't receive new messages due to memory shortage at the moment.

Good Luck ☺

The only thing that overcomes hard luck is hard work.

Harry Golden