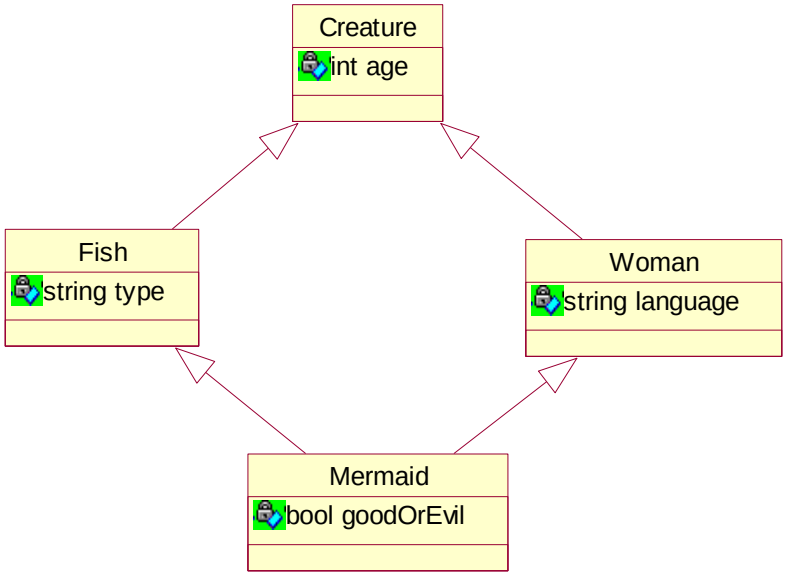
Third Assignment OOAD (Fall 2017) Due on Dec 1

Q1 (10)

Consider the following class diagram:



a) If we implement this diagram in C++ the compiler will raise error complaining that the class Mermaid has two copies of the attribute age. How will you resolve this issue? Briefly explain.

b) Consider the following code for the constructor of the class Mermaid:

```
Mermaid::Mermaid(int a, string t, string l, bool goe)
     : Fish(a, t), Woman(a, l), goodOrEvil(goe) {}
```

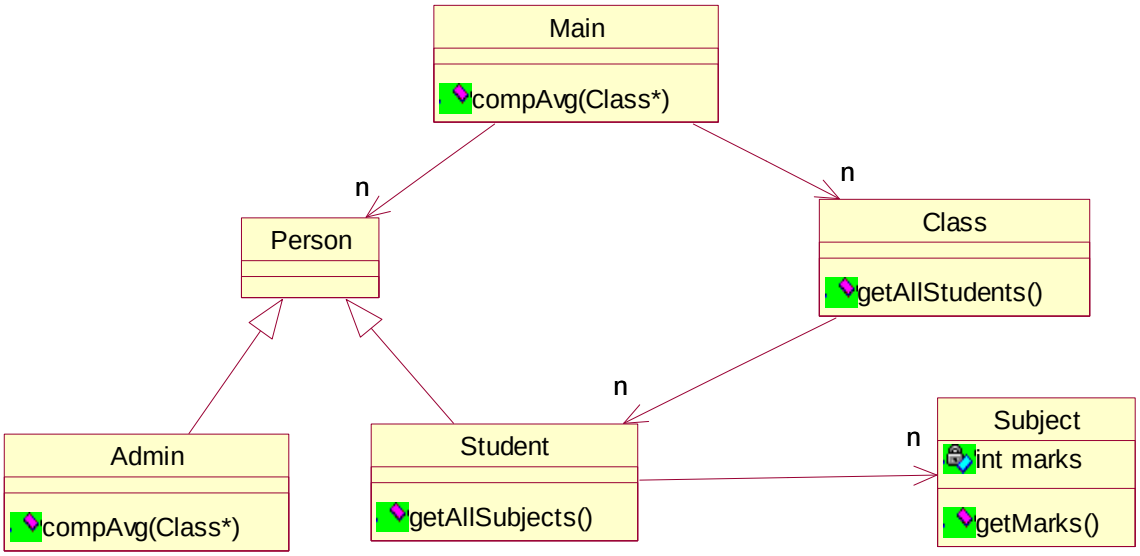
Identify and correct any error(s) in the above constructor.

Q2(10)

Consider the class diagram of the Airline Reservation System given on page 212 of your text book. Now give a design level sequence diagram to compute a travel log for a given person. Besides the person name, the report shall contain the following information for each flight: date, time, flight number, and seat no. Remember (a special case of) the Law of Demeter: do not talk to your neighbor's neighbor!

Q3(10)

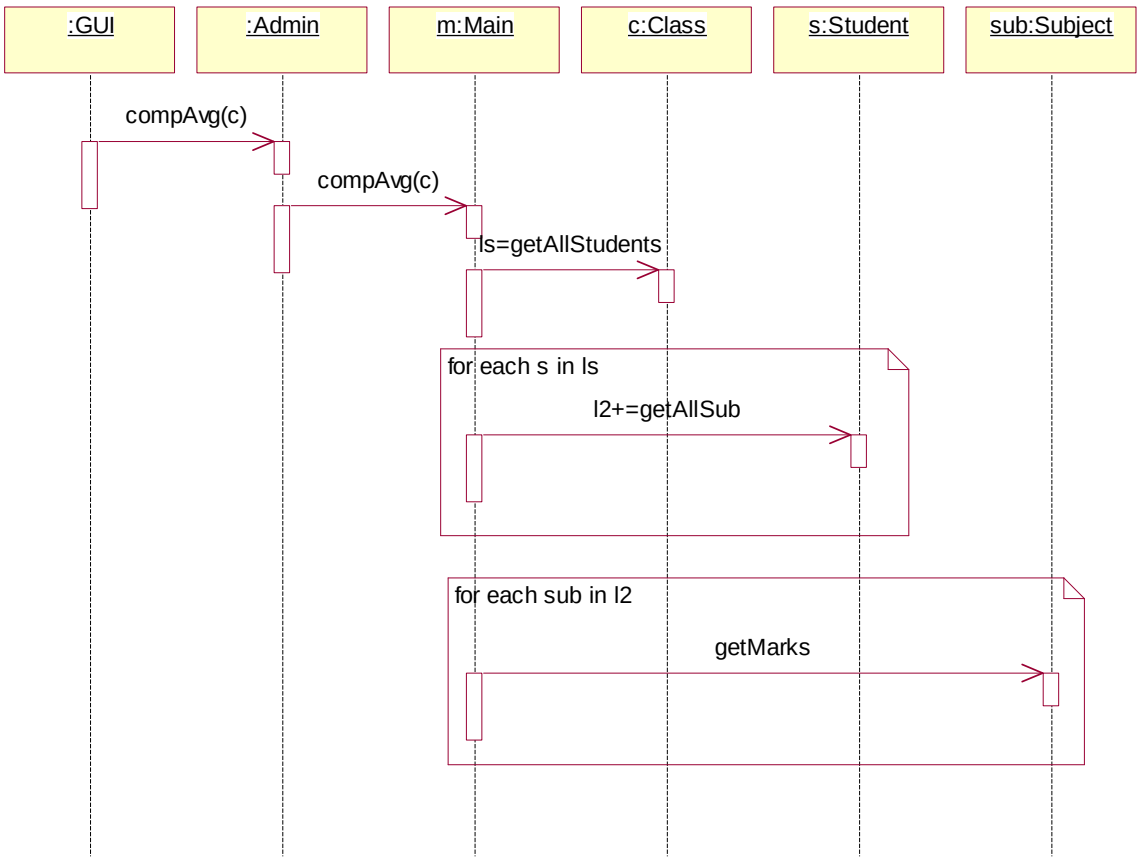
Consider the following class diagram modeling some academic environment:



Also consider the sequence diagram on the following page, depicting the object interactions required to compute class average.

The sequence diagram violates some design principles. Secondly this design is inefficient, especially when the computation starts. Refactor the design to improve quality. You may add new functions in the class diagram if need arises.

Q



Q4 (10)

Some software developers do not develop the business layer. What are the drawbacks of this approach. Briefly explain cause of each drawback.

Q5 (10)

In three-tier architecture a problem arises when we need to search many records of an entity stored in the database. It is very inefficient to dispatch the query every time to the remote database. On the other hand if we fetch all the attributes of all the records, the initialization may require unbearable time and memory!

How can we solve this problem using the Proxy design pattern? Briefly explain. Also give an example.