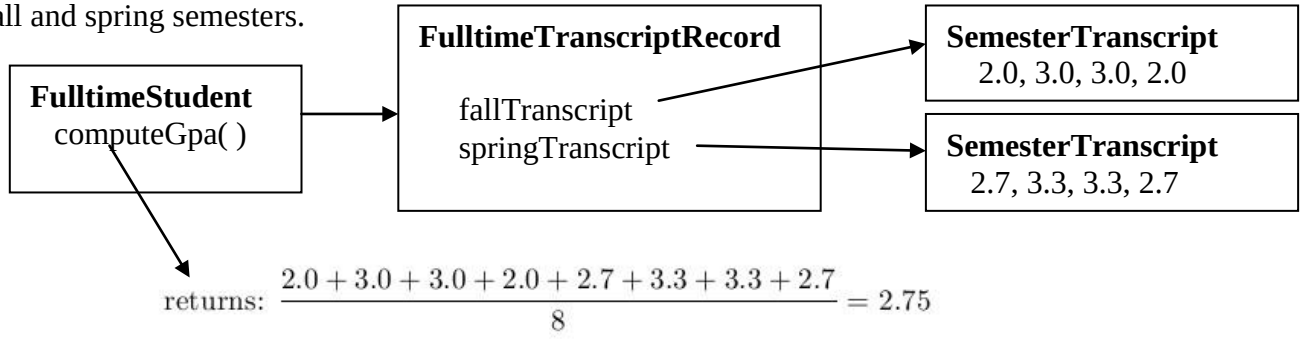
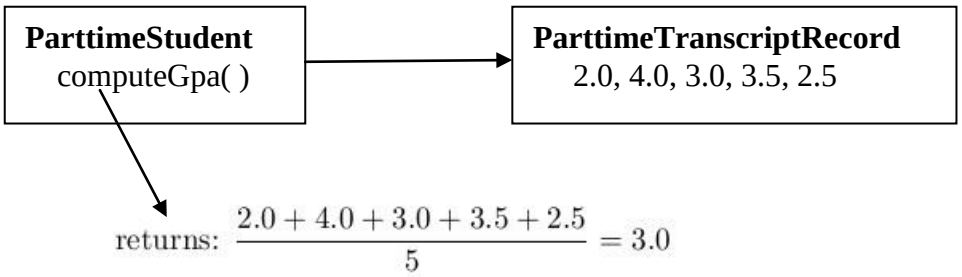
Exam Practice for OOP Question 2

The administrative office at a particular university keeps records of all student grade point averages (GPAs). Normal record formatting is done for full-time students, but a less detailed type of record formatting is done for part-time students.

In the `prob2` package, you will find the classes `FulltimeStudent`, `FulltimeTranscriptRecord` and `SemesterTranscript`. These classes work together to provide information about a full-time student's performance during a particular school year. A `SemesterTranscript` just contains a list of grades (numerical values in the range 0.0 – 4.0). A `FulltimeTranscriptRecord` contains a `fallTranscript` and a `springTranscript`, each of which is an instance of `SemesterTranscript`. And a `FulltimeStudent` has a `FulltimeTranscriptRecord`, which provides a complete record of a student's grades in both the fall and spring semesters.



The `FulltimeStudent` class provides a method `computeGpa` which sums all the grades stored in the student's `FulltimeTranscriptRecord`, computes the average, and returns it. Also in the `prob2` package, you will find the classes `ParttimeStudent` and `ParttimeTranscriptRecord`. These classes work together to provide information about a part-time student's performance during a school year. A `ParttimeTranscriptRecord` contains a list of grades (numerical values in the range 0.0 – 4.0), and every `ParttimeStudent` has a `ParttimeTranscriptRecord`. Note that grades for a part-time student are tracked for the whole year, and not considered separately for each semester as they are for full-time students.



A `ParttimeStudent` contains a method `computeGpa` which sums all the grades in the student's `ParttimeTranscriptRecord`, computes the average, and returns it.

The objective of this problem is to compute the average gpa for the current year of all students in a given input array. This is to be accomplished in two steps, by implementing the following two static methods contained in the `Admin` class:

```
public static List convertArray(Object[] studentArray)
public static int computeAverageGpa(List studentList)
```

The `convertArray` method converts the array of students that is passed to it by a calling class (in this problem, the calling class is the `Main` class) and converts it to a `List` of the proper type. The `computeAverageGpa` method uses this list to polymorphically compute the average gpa of all the students in the list.

In order to do your polymorphic computation of average gpa, in the `convertArray` method you will need to make use of a common type for both types of students (part-time and full-time) that may occur in the input array; the abstract class `Student` (unimplemented) has been provided in your `prob2` package for this purpose. With this common type, you will be able to do the necessary polymorphic computation in `computeAverageGpa`.

Requirements for this problem.

(1) You must compute average gpa *using polymorphism.*

(2) Your implementation of `computeAverageGpa` may not check types (using `instanceof` or `getClass()`) in order to `computeGpa` from any of the students in the input list.

(3) You must use parametrized lists, not "raw" lists. (Example: This is a parametrized list: `List<Duck> list`. This is a "raw" list: `List list`.) This means that all `Lists` that appear in the code (in the `Main` class and in the `Admin` class) must be given proper type parameters.

(4) You must implement both the methods `convertArray` and `computeAverageGpa` in the `Admin` class.

(5) Your computation of average gpa must be correct.

(6) You may not remove the `abstract` keyword from `Student` or change `Student` to be an interface.

(7) There must not be any compilation errors or runtime errors in the solution that you submit.