CS109 Assignment 4

The ppt only provides one possible problem-solving idea, which does not mean that this is the only idea or the best problem-solving idea.

Designer: Yueming ZHU

Credits

- In Course. The credits is a ArrayList<Integer>, which record the credit of cooresponding Student in enrollStudent list.

course.getCredits() -> return a list

- In Student. The credits is an int value, which represents the credits the student has right now.

student.getCredits() -> return a value

The method: public boolean enrollStudentInCourse(Student student, String courseId, int points)

It return false only when:

1. ifOpen is false.
2. The parameter points <=0
3. The course doesn’t exists. (traverse in this.courses)
4. The student is not enrolled in it.

   - traverse in student.getEnrollCourses(). If student has enrolled, it can get a course object.

5. The student doesn’t have enough credits to bid.

If successful, you need to do:

1. The student’s credits will be reduced by points.
2. In course:
   - add student into enrolled student list
   - add points into credits list
3. In student:
   - add course into enrolled courses list

The method: public boolean modifyStudentEnrollmentCredits(Student student, String courseId, int credits)

It return false only when:

1. ifOpen is false.
2. The course doesn’t exists. (traverse in this.courses)
3. The student is not enrolled in it.

   - traverse in student.getEnrollCourses(). If student has enrolled,

   it can get a course object.
   it can also get an index of enrolled course.
   from the index, we can get the corresponding credits of enrolled course

4. New bis is not with in the student’s available credits.

If successful, you need to do:

1. In students: update his/her remained credits
2. In course: update the corresponding credits according to the index.

   You can try public E set(int index, E element) in ArrayList

The method: public boolean dropStudentEnrollmentCourse(Student s, String courseId)

It return false only when:

1. ifOpen is false.
2. The course doesn't exists. (traverse in this.courses)
3. The student is not enrolled in it.

   - traverse in student.getEnrollCourses(). If student has enrolled,

   it can get a course object.

   it can also get an index of enrolled course.

If successful, you need to do:

1. In course:
   - Remove the corresponding credits in credits list according to index.
   - Remove the student in enrolled students list.
2. In Student:
   - Remove enrolled course in course list.
   - Update the credits.

public ArrayList<String> getEnrolledCoursesWithCredits(Student student)

It return null only when:

1. ifOpen is false.

If successful, it will return a String list:

1. New an ArrayList<String>
2. Traverse each course in enrolled courses:
    - Find the credits the student cost in each course like c.
        - In c, you can find two list:
            1. enrolled student list: find the index of student.
            2. credits list: get the credits of corresponding student according to index

public void finalizeEnrollments()

1. Set ifOpen by false.
2. For each couse in this.course, you can get two list:
   1) The enrolled students list
   2) The credits list
   - According to the credits list and the capacity, you can calculate **an integer**, larger then which, the students are all being selected successfully.
   - Traverse the credits list and if you find the credits >= the integer, which means the corresponding index of enrolled student is successful.
   - If we find a successful student, you wiil:
     1. In course: add the student into successStudents list
     2. In student: add the course into successCourses list