

CS136L: Computer Science II Lab – Fall 2018

Assignment	Points	Announced	Due
#6	40	10/22	11/09

Questions

1 Overview

In this assignment, we will develop a simple hierarchy of classes. Most likely, you have taken computer-graded quizzes. A quiz consists of questions, and there are different kinds of questions:

- Fill-in-the-blank
- Choice (single or multiple)
- Numeric (where an approximate answer is ok; e.g., 1.33 when the actual answer is $4/3$)
- Free response



2 Learning Outcomes

By the end of this project students should be able to:

- write algorithms based on a technical problem description;
- utilize inheritance;
- work effectively with a partner using pair-programming;
- write an effective report that describes the students' problem-solving process.

3 Pre-Lab Instructions

Do this part before you come to lab:

- Study the question hierarchy described in chapter 9 of the textbook.
- Study the classes attached to this assignment.

4 Submission Instructions

1. Use a basic text editor (e.g. Notepad, Notepad++, Atom...) and not an IDE for this lab assignment.
2. In addition to the lab report, submit the four Java files described below.
3. Don't submit .class files nor .jar files nor archives (e.g. .zip, .rar ...)
4. Add Javadoc comments for all classes, fields and methods that you implement

5 Lab Instructions*Do this part in lab:*

1. Add a class `NumericQuestion` to the `Question` hierarchy. If the response and the expected answer differ by no more than 0.01, then accept the response as correct.
2. Add a class `FillInQuestion` to the `Question` hierarchy. Such a question is constructed with a string that contains the answer, surrounded by `_ _`, for example, "The inventor of Java was `_James Gosling_`". The question should be displayed as "The inventor of Java was _____".
3. Add a class `AnyCorrectChoiceQuestion` to the `Question` hierarchy that allows multiple correct choices. The respondent should provide any one of the correct choices. The answer string should contain all the correct choices, separated by spaces. Provide instructions in the question text.
4. Update the attached `QuestionsApp` class so that it creates a single `ArrayList` of 6 questions (2 numeric questions, 2 fill-in questions and 2 any-correct choice questions). Use a single enhanced for-loop to present these questions to a user and to collect the answers. Assume that a correct numeric question is worth 1 point, a correct fill-in question answer is worth 2 points and a correct any-correct choice question answer is worth 3 points. Calculate and display the total score for the user.

A sample screenshot is below:

What is the value of PI to the nearest thousandth?

Your answer: 3.141

Correct answer :)

What is the value of the Euler's number to the nearest thousandth?

Your answer: 2.711

Correct answer :)

The inventor of Java was _____

Your answer: James Gosling

Correct answer :)

The inventor of Pascal was _____

Your answer: Nicolas Wirth

Wrong answer :(

The inventor of Java lived in:

1: Australia

2: Canada

3: Denmark

4: United States

Note, there may be several correct answers.

Your answer: 3

Wrong answer :(

The inventor of Pascal lived in:

1: Australia

2: Canada

3: Denmark

4: United States

5: France

Note, there may be several correct answers.

Your answer: 1

Wrong answer :(

Total score is 4 out of 12

Thank You!

6 Lab Report

Each pair of students will write a single lab report together and each student will turn in that same lab report on BBLearn. Submissions from each student on a pair should be identical.

Your lab report should begin with a preamble that contains:

- The lab assignment number and name
- Your name(s)
- The date
- The lab section number

It should then be followed by four numbered sections:

1. Problem Statement

In this section you should describe the problem in *your* own words. The problem statement should answer questions like:

- What are the important features of the problem?
- What are the problem requirements?

This section should also include a reasonably complete list of requirements in the assignment. Following your description of the problem, include a bulleted list of specific features to implement. If there are any specific functions, classes or numeric requirements given to you, they should be represented in this bulleted list.

2. Planning

In the second section you should describe what planning you did in order to solve the problem. You should include planning artifacts like sketches, diagrams, or pseudocode you may have used. You should also describe your planning process. List the specific data structures or techniques you plan on using, and why.

3. Implementation and Testing

In the third section you should describe how you implemented your plan. As directed by the lab instructor you should (as appropriate) include:

- a copy of your source code
- a screen shot of your running application / solution
- results from testing

4. Reflection

In the last section you should reflect on the project. Consider different things you could have done to make your solution better. This might include code organization improvements, design improvements, etc.

You should also ask yourself what were the key insights or features of your solution? Were there alternative approaches or techniques you could have employed? How would these alternatives have impacted a different solution?

5. Partner Rating

Every assignment you are required to rate your partner with a score -1, 0 or +1. This should be submitted in the comment section of the BBLearn submission, and not in the report document. If you don't want to give your partner a negative rating making sure not to use a dash before listing the number! You do not have to tell your partner the rating you assign them. A rating of 1 indicates that your partner was particularly helpful or contributed exceptional effort. A rating of 0 indicates that your partner met the class expectations of them. Rating your partner at -1 means that they refused to contribute to the project, failed to put in a reasonable effort or actively blocked you from participating. If a student receives three ratings of -1 they must attend a mandatory meeting with the instructor to discuss the situation, and receiving additional -1 ratings beyond that, the student risks losing a letter grade, or even failing the course.

6. Contribution

Every assignment you are required to describe your contribution to coding of the solution and writing of the report. You must include the percentage of your contribution. This should be submitted in the comment section of the BBLearn submission, and not in the report document.

7 Grading Rubric

This lab assignment will be graded according to this rubric:

Criteria / Component	Points
Report	10 pts
Correct implementation of <code>NumericQuestion</code>	7.5 pts
Correct implementation of <code>FillInQuestion</code>	7.5 pts
Correct implementation of <code>AnyCorrectChoiceQuestion</code>	7.5 pts
Correct implementation of <code>QuestionsApp</code>	7.5 pts

Lab assignment penalties:

Item	Points
Compilation errors	-40
Missing program	-40
Missing report	-10
Missing names	-4
Missing partner rating	-2
Missing screenshots	-2
Missing contribution description and/or percentage	-5
Too late to pair (if attended)	-4
Absent	-10
Insufficient / No commenting (if required)	-5
Improper format (e.g. NetBeans ZIP archive instead of a Java file)	-5

Note:

If your partner is not responding to your emails and/or not collaborating, don't hesitate to reach out to the lab TA aide and/or the primary instructor.