

SOEN 6461 Fall 2018 — Assignment 1 Marking

Mark each section 0, 1, 2 marks; no fractions.

Problem Description

This should contain

- ▶ Task Description
Develop a class `StudentClassSchedule` to represent/store the schedule of classes for a student for a semester
- ▶ Inputs and Outputs
Inputs: student, semester, classes with times and locations
Outputs: a display/print out of the schedule
The only specified task is to display the schedule
- ▶ Assumptions on Hardware and Software Context
There are none specified, other than OO for the design.
- ▶ Indication of Non-Functional Qualities of Interest
Besides correctness, the students were asked to consider resource usage for memory and cpu steps.

For two marks, must include task clearly described, something on inputs/outputs, and something about qualities required.

Give zero marks, if the task is not clearly described. It is a description for Assignment 1 and not for the whole system.

Design Description

The web page gave them lots of guidance as what to include here.

It is a description of a class `StudentClassSchedule`.

Simply a description; not a discussion of choices or motivation, etc.

This should contain

- ▶ Name
- ▶ Purpose / Responsibility
- ▶ List of services / operations / methods
CRUD is not critical, but display/print is!
- ▶ signature for each service
- ▶ contract for each service
These are trivial.
Post-condition for `display` should show they realize that the order of displaying classes should follow days of the week, and hours in the day.

- ▶ Attributes and the selected data structure
Must have student, semester, collection of classes.
Data structure should be high-level like List, Map, Set, etc; okay to be List<Class> but not List<Course>; okay to be Map from TimeSlots in the week to Class, or similar
- ▶ Description of major algorithm: display
Ideally a high-level pseudocode that recognizes order of display is important:
eg

```
for each Class c in Schedule ordered by day-of-week and time-of-day{
    c.display()
}
```

There is no need for a UML class diagram. But if they include one, then it must have a number, title, be clear and readable, focus on **StudentClassSchedule**. A diagram does not replace the need for text descriptions of the above items.

Do not worry about exceptions in the signatures.

For two marks, they must include purpose/responsibility, contract for display, data structure for the schedule; and their descriptions need to be clear. Also, any diagram must be clear, and consistent with the text.

Give zero marks, if they do not describe the data structure for the schedule, or do not have a display/print service/method/operation.

Major Design Decisions

This should contain a short list of decisions with a description

- ▶ What was the issue
- ▶ What was decided
- ▶ What was the impact of the decision
- ▶ Maybe alternatives

The single major decision is the choice of data structure for the schedule.

They might include the choice of algorithm for the display/print as well.

For two marks, they need to talk about the choice of data structure, the issue, and the impact of their choice. No need to talk about alternatives.

Give zero marks, if they do not mention the choice of data structure as a major decision.

Design Review

This should contain

- ▶ Brief Description of their review steps
 - Scenario-driven review
 - Which qualities: correctness, data usage, computation time
 - Some details of concrete scenario, system state, and inputs
- ▶ Justification of correctness
 - Logical reasoning showing that the schedule is correctly displayed
 - This is straightforward, as it only requires that they recognize that schedule should be displayed in an appropriate order following the days of the week and the hours of the day.
- ▶ Formulas for data and computation
 - This should be done in terms of the number C of classes in the schedule; the memory $m(\text{Class})$ used to store one Class object; and the steps $c(\text{Class})$ required to display one Class object.
 - Basically $C*m(\text{Class})$ and $C*c(\text{Class})$

For two marks, they need some specifics on the scenarios, reasoning about correctness, and formulas for memory and time usage.

Give zero marks, if they simply say “scenario-driven” review, without details.

Glossary

This should contain

- ▶ List of Dictionary Entries
 - Short definition, like a dictionary, of important terms in design. Should have schedule, class, course (with a clear distinction between course and class). Might have student, semester, time, location, etc.
 - Not terms from OO or SE. So not class in OO sense, or object, state, design, etc

For two marks, need at least 3 terms with good to-the-point definitions, and no OO/SE terms.

If only OO/SE terms, then give zero marks.

Marks Spreadsheet

For me, create a spreadsheet which gives the five marks against each students name and ID. Have a column with the total out of 10.

Also indicate number of days late in a column, and page length in a column.

Do not count days late until after 9am following the deadline; do not count Declaration Form in the pages, and do not count a “title page” if they have a separate title page (ie has no design information).