

**THE UNIVERSITY OF BRITISH COLUMBIA**  
**CPSC 310: SAMPLE MIDTERM #2**

**Name:** \_\_\_\_\_

**Student #:** \_\_\_\_\_

**Signature:** \_\_\_\_\_

**Notes about this examination**

1. You have 60 minutes to write this examination.
2. No notes, books, or any type of electronic equipment is allowed including cell phones and calculators.
3. Good luck!

	Marks	Max
Multiple Choice		<b>14</b>
True/False		<b>12</b>
14		<b>2</b>
15		<b>3</b>
16		<b>7</b>
17		<b>2</b>
18		<b>4</b>
19		<b>8</b>
20		<b>8</b>
Total		<b>60</b>

***Rules Governing Formal Examinations***

1. Each candidate must be prepared to produce, upon request, a UBCcard for identification.
2. Candidates are not permitted to ask questions of the invigilators, except in cases of supposed errors or ambiguities in examination questions.
3. No candidate shall be permitted to enter the examination room after the expiration of one-half hour from the scheduled starting time, or to leave during the first half hour of the examination.
4. Candidates suspected of any of the following, or similar, dishonest practices shall be immediately dismissed from the examination and shall be liable to disciplinary action.
  - Having at the place of writing any books, papers or memoranda, calculators, computers, sound or image players/recorders/transmitters (including telephones), or other memory aid devices, other than those authorized by the examiners.
  - Speaking or communicating with other candidates.
  - Purposely exposing written papers to the view of other candidates. The plea of accident or forgetfulness shall not be received.
5. Candidates must not destroy or mutilate any examination material; must hand in all examination papers; and must not take any examination material from the examination room without permission of the invigilator.
6. Candidates must follow any additional examination rules or directions communicated by the instructor or invigilator.

## Multiple Choice

1. [2 marks] Among the following, select all the non-functional requirements
  - a) The system must save a backup copy of the marking report every time the marker edits the marking report.
  - b) The system must be able to import and export marking schemes.
  - c) The software should be usable even if the computer is not connected to a network.
  - d) The system must be able to handle 100 concurrent markers.
  - e) The system must display a progress bar when an assignment is being imported.
  - f) The system must report an error whenever the user tries to modify a marking scheme.
  - g) The marker must be able to save their name and email address as a preference.
  - h) The marker must be able to access the user manual in English and French.
  - i) The system must remain functional for 1 month without necessitating a reboot.
  
2. [2 marks] During a requirement elicitation interview, you just asked the following questions:
  - i) What are the most frequent tasks you perform?
  - ii) Would you like those tasks to be automated?

In which step of a typical requirements interview template are you in:

  - a) Assessing The Problem
  - b) Validation
  - c) Establish Customer and User Profile
  - d) Functional Requirements
  - e) Understand the User Environment
  
3. [2 marks] Which of the following *is not* part of a typical SRS?
  - a) Glossary
  - b) Functional Requirements
  - c) Use Cases
  - d) Non-Functional Requirements
  - e) Class diagrams

4. [2 marks] Which design principle is violated by the following code?

```
// in the class SoccerTeam
setScore(4, 0);
// must call setScore before setGoalScorers
setGoalScorers(goalScorers);
```

- a) Information Hiding
  - b) Law of Demeter
  - c) Weak Coupling
  - d) High Cohesion
  - e) Open/Closed Principle
  - f) Liskov Substitution Principle
  - g) None
5. [2 marks] Which design principle is violated by the following code?

```
// in the class RetailStore
public final int getMaxWage() { return 10; }
```

- a) Information Hiding
  - b) Law of Demeter
  - c) Weak Coupling
  - d) High Cohesion
  - e) Open/Closed Principle
  - f) Liskov Substitution Principle
  - g) None
6. [2 marks] Which design principle is violated by the following code?

```
public class SalaryEmployee extends HourlyEmployee
{
    ...
    public int calcOvertimePay() {
        // Salaried employees don't earn overtime
        return -1;
    }
    ...
}
```

- a) Information Hiding
- b) Law of Demeter
- c) Weak Coupling
- d) High Cohesion
- e) Open/Closed Principle
- f) Liskov Substitution Principle
- g) None

7. [2 marks] Sequence diagrams are used to:
- show the relationships between classes in a system, especially which fields belong in which classes
  - show how each individual class works, including code snippets about how to use each class
  - show how several classes work together to perform a specific function, often a use case
  - describe the type hierarchy and how it effects the system
  - None of the above.

## True/False

For each true/false question below provide a *one sentence* justification of your answer.

8. [2 marks] When you are using the waterfall lifecycle, you cannot modify the requirements once you have started on the design phase.
9. [2 marks] You would use the subsystem waterfall model if you're planning a project in which four development teams will be developing a product with four independent subsystems.
10. [2 marks] A UML class diagram shows the relationships and communication between classes in a software system.
11. [2 marks] Most of Eclipse (except for the core runtime engine) is implemented as plugins.
12. [2 marks] An Eclipse extension point allows you to extend and customize the behaviour of Eclipse.
13. [2 marks] Information hiding, high cohesion and the Liskov Substitution Principle are all design principles that will help you achieve modularity in your software.

### Short Answer

14. **[2 marks]** What are two benefits of following a lifecycle (eg, waterfall) during software development?

15. **[3 marks]** In two or three sentences, explain how you know when requirements elicitation is complete.

**The below space is intentionally blank.**

16.

- a. **[3 marks]** Briefly explain how you would perform requirements validation. What types of activities would you perform?

- b. **[4 marks]** Give an example of a problem you might discover during the requirements validation phase, and describe how you would rectify the problem.

17. **[2 marks]** In two or three sentences, explain why the following the Law of Demeter will help you design modular software.

18. **[4 marks]** Write a casual use-case describing a withdrawal at an ATM (bank machine).

**The below space is intentionally blank.**

19. **[8 marks]** Imagine that you are working as a consultant, and a potential client has called you to request a meeting. Your potential client is an interior designer who is having trouble tracking all of her clients using paper-based files, and is interested in having you build a software system to assist her. You need to meet with her in 10 minutes! Sketch out a plan for the interview.



20. **[8 marks]** Draw a UML class diagram for the following software system for modeling a bank. Make sure to include multiplicities in your diagram.

Each of the bank's customers can access their account(s) through withdrawals, deposits, or balance inquiries at a bank machine. Each transaction (ie, withdrawal, deposit or balance inquiry) must store the date and time that the transaction occurred. Once a month, a statement that contains a list of all of the transactions that were completed over the last month is generated for each account and mailed to the customer. The bank must be able to produce a list of all of its customers as well as a list of transactions that were completed by a particular bank machine.