

**THE UNIVERSITY OF BRITISH COLUMBIA**  
**CPSC 310: SAMPLE FINAL EXAM**

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

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

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

**Notes about this examination**

1. You have **150 minutes** to write this examination.
2. There are **120 marks** available. You may wish to start with the questions that you consider to be the easiest.
3. No notes, books, or any type of electronic equipment is allowed, including cell phones and calculators.
4. Good luck!

	Marks	Max
Multiple Choice		<b>20</b>
True/False		<b>20</b>
21		<b>6</b>
22		<b>8</b>
23		<b>8</b>
24		<b>12</b>
25		<b>6</b>
26		<b>6</b>
27		<b>8</b>
28		<b>8</b>
29		<b>6</b>
30		<b>6</b>
31		<b>6</b>
Total		<b>120</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.

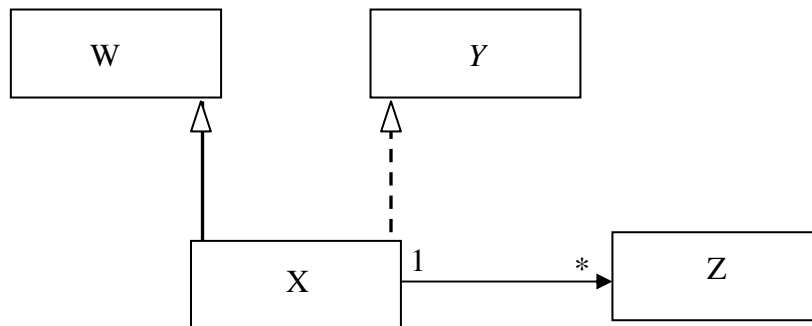
1. [2 marks] Select the best functional requirement from the list of requirements below.
  - a) A warning dialog should pop up if the student's assignment does not compile.
  - b) The system needs to manage the student grades.
  - c) The marker should be able to use this tool to aid the assignment evaluation process.
  - d) The marker must be able to edit comments in the marking report.

d

2. [2 marks] You should use this design pattern when extension by subclassing is impractical and you need to add responsibilities to individual objects dynamically.
  - a) Composite
  - b) Mediator
  - c) Decorator
  - d) Singleton
  - e) None of the above

c

3. [2 marks] Which names make the most sense for the UML diagram shown below?



- a. W: Person, X: Employee, Y: ComparableInterface, Z: Responsibility
- b. W: Cat, X: Dog, Y: Animal, Z: Leg
- c. W: Inventory, X: RetailInventory, Y: AbstractInventory, Z: Item
- d. W: Person, X: Parent, Y: Employee, Z: Child

a

[2 marks] Which design principle is violated in the following code snippet?

```
public class FurnitureTable {

import java.awt.Color;
import java.awt.event.MouseAdapter;
import java.awt.event.MouseEvent;
import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;

import com.sample.model.Content;
import com.sample.model.FurnitureEvent;
import com.sample.model.FurnitureListener;
import com.sample.model.ContentManager;
import com.sample.model.Home;
import com.sample.model.HomePieceOfFurniture;
import com.sample.model.SelectionEvent;
import com.sample.model.SelectionListener;
import com.sample.model.UserPreferences;
import com.sample.model.HomePieceOfFurniture.SortableProperty;
import com.sample.model.FurnitureCatalog;
...
}
```

- a. Open/Closed Principle
- b. Law of Demeter
- c. Liskov Substitution Principle
- d. High Cohesion
- e. Low Coupling
- f. None of the above

e

4. [2 marks] Conflict between members of a software development team:

- a. Can lead to a better product
- b. Can be handled poorly and cause some team members to feel uncomfortable
- c. Can arise if one or more team members have different ideas about the best way to solve a problem
- d. a and b
- e. All of the above

e

5. [2 marks] You have to test a *factorial* function, but you don't have access to the code. Which of the following is the best set of test inputs?

- a) 1, 2, 5, 200
- b) -5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5
- c) -4, 1, 5, 250
- d) All the positive integers until the program crashes

c

**[2 marks]** What type of coverage would you get if you tested the following code with the values

$x = 3, y = 4$  and  $x = 2, y = 1$ .

```
public void calc(int x, int y) {
    if (x < y)
    {
        if (x > 2)
        {
            y++;
        }
    }
    else
    {
        y--;
    }
}
```

- Statement coverage but not branch or path coverage
- Statement and branch coverage but not path coverage
- Statement, branch and path coverage
- None of the above

a

6. **[2 marks]** When you are part of a test-driven development team, in which order should complete the following tasks?

- Write the code, Make the code compile, Write the tests, Make the tests pass
- Write the tests, Write the code, Make the code compile, Make the tests pass
- Write the code, Write the tests, Make the code compile, Make the tests pass
- None of the above

b

7. **[2 marks]** You are working on a poster to advertise around campus for your new tech start-up. You'd like to include a picture of a puppy on your poster to soften your audience. You go to Google Images, do a search for "puppy" and find the perfect image on an artist's website. There is no copyright notice anywhere on the image or on the website. Are you free to include this image in your presentation?

- Yes - since you are using only one of their images, and it won't be the main focus of your poster, this qualifies as fair dealing.
- Yes - since there is no copyright notice anywhere on the image or website, the image is not protected by copyright and you may use it freely.
- Yes - copyright only applies to printed materials, not to electronic media that you find online, so you are free to use the image.
- No - this is copyright infringement and is illegal.
- No - you must wait until 20 years after the image was published before you may use it freely.

d

[2 marks] What's the worst code smell in the following code snippet?

```
public class CapitalStrategy {  
  
    // other code not shown ...  
  
    public double capital(Loan loan){  
        if (loan.isExpired() && !loan.isMature())  
        {  
            return loan.getCommitment() * loan.duration() * loan.riskFactor();  
        }  
        if (!loan.isExpired() && loan.isMature())  
        {  
            if (loan.getUnusedPercentage() != 1.0)  
                return loan.getCommitment() * loan.getUnusedPercentage() *  
                    loan.duration() * loan.riskFactor();  
            else  
                return (loan.amount() * loan.duration() * loan.riskFactor())  
                    + (loan.unusedAmount() * loan.duration() * loan.unusedFactor());  
        }  
        return 0.0;  
    }  
}
```

- a. Data Class
- b. Duplicated Code
- c. Decompose Conditional
- d. Feature Envy
- e. Long Method

d

The space below is intentionally blank.

---

## True/False

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

8. [2 marks] The EPL (Eclipse Public License) is more business-friendly than the Gnu GPL (General Public License).

T (not covered this term)

9. [2 marks] You cannot use XP if you cannot create automated tests for your software.

T (You can use some aspects of XP, but you must have automated tests to fully use the XP process)

10. [2 marks] A use case diagram is a complete graphical representation of a use case.

F, a use case diagram shows all the use cases for a system.

11. [2 marks] Testing is used to demonstrate that software is free of errors.

F, testing can demonstrate the presence of errors but not the absence.

12. [2 marks] It is expected that a team of experienced software developers using an agile development process will produce smelly code.

T, agile teams follow the “just barely good enough” approach which will lead to smelly code that needs to be refactored as the system evolves.

13. [2 marks] Design patterns encapsulate existing, well-proven re-usable blocks of design and therefore should be applied without modification.

F, sometimes the design patterns will need to be modified to suit the particular problem you are trying to solve.

14. [2 marks] When using a version control system (eg. Jazz), a new version of the software (aka, a baseline) should only be created if you are planning on publically releasing the version.

F, a new version can be useful for many other reasons (eg, an internal milestone or a client demo)

15. [2 marks] Reusing previously tested and deployed software components always leads to higher software quality.

F, not always. If the previously tested and deployed components aren't suitable for the particular problem you're working on, they won't improve your software quality.



16. **[2 marks]** Clear and specific requirements can be very useful when planning acceptance testing.

T, acceptance tests are supposed to test whether the software meets the client's requirements. Clear and specific requirements can help you plan this kind of testing.

17. **[2 marks]** Professional software engineers do not need to worry about intellectual property issues unless they own their own company.

F, all software engineers need to be careful that they are following intellectual property laws. For example, a software engineer who uses an open source library needs to be sure that he/she understands the license the library uses and how it affects the software that he/she is writing.

### Short Answer

18. **[6 marks]** You decided to start a new software company with some friends. Unfortunately, most of your friends who are starting the company with you are inexperienced programmers and don't have great programming style. They tend to leave out comments, format code in a hard to understand way, and generally deliver code with poor style. What could you do to improve this situation and why would it work?