

SENG301 2022 - Final exam (export from Learn quiz)

Copyright - Fabian Gilson & Neville Churcher

Opened: Friday, 17 June 2022, 2:25 PM

Closed: Friday, 17 June 2022, 5:30 PM

This is the Final examination of SENG301 and **it is considered as invigilated**. It counts for **50%** of the course grade. **You need to achieve a grade of 45%** to this examination **to pass the course**.

This is an open book exam: you may use all (electronic) resources supplied during the course (i.e. lecture notes, additional readings, preparation work, lab material).

You have 180 minutes starting from the starting time. The quiz will be open 5 minutes prior the official time. There is a **grace period of 5 minutes for you to finish your current answer** at the end of the examination.

Instructions to Students

Answer all questions. Please write in a readable and intelligible manner, **unreadable answers will be ignored**.

Read the entire question before answering any part of it. **A few questions have associated figures** or notes that are crucial to the question.

Some questions require you to describe or explain some concepts. You are expected to write **complete sentences**. **No answers made of bullet points will be accepted**.

Check carefully the **number of marks allotted to each question**. This suggests the degree of detail required in each answer, and the amount of time you should spend on the question.

No form of collaboration is permitted:

You **may not communicate** with anyone during the exam, and you **may not post material** relating to the exam online. This is simply a matter of ethical behaviour and **academic integrity**. Your online activity leaves a **footprint we eventually can trace**. Stay ethical and keep your integrity as you sit the exam by making sure everything you do is your own work.

In the unlikely event you would decide to collaborate in any form and we would have any form of suspicion, **we may contact you for an additional oral exam**. If we have sufficient doubts about the intellectual integrity of your exam answers, **we may open a case to the Registrar** with the potentially detrimental academic consequence this may lead to (e.g., failing the course, exclusion of the University).

Academic integrity

Academic integrity is a principle at the University of Canterbury whereby both staff and students agree to act honestly, fairly, ethically and with respect for each other in teaching and learning.

For some students, there may be increased temptation to cheat and engage in dishonest academic practices such as:

- **Plagiarism** using someone's ideas and information without acknowledging them as the source;
- **Self-plagiarism** where someone attempts to submit their own writing to two different assessments to gain credit twice;
- **Collusion** copying the work of someone else or allowing someone else to copy your work without disclosing this with the intent to deceive;
- **Impersonating/Ghost writing** having another person or commercial organisation impersonate you and complete an assessment item on your behalf;
- **Fabrication** "inventing" facts or data for example in a lab report or from a publication.

Declaration


By **taking part to and completing** this examination,

1. I confirm that I have **read and understood the rules** of the exam and the **Academic integrity** principle.
2. I understand that the **exam is considered confidential** and I must not discuss the exam materials until the exam is over for all students.
3. I understand that **failure to comply** with these requirements may mean that **the matter will be referred to** the Head of Department, Dean or **Proctor** as appropriate for disciplinary action.

Attempts allowed: 1

Settings overrides exist ([Groups: 1](#))

Summary of your previous attempts

State	Review
In progress	

This quiz is currently not available.

For support please contact the [IT Service Desk](#) on 0508 UC IT HELP (0508 824 843) or on 03 369 5000.

Alternatively, visit us at Te Pātaka (Level 2, Central Library).

For support hours, please visit the [IT Services](#) page.

Unless otherwise stated, you may not sell, alter, further reproduce or distribute any part of this material on LEARN to any other person, without permission from the copyright owner. See [UC copyright policy](#).



You can preview this quiz, but if this were a real attempt, you would be blocked because:

This quiz is currently not available.

Question 1

Not yet answered

Marked out of 10.00

Describe how a sprint planning typically works. Specifically, describe:

- who is involved in the event (1 mark)
- what are the various "phases" of this event (1 mark)
- what are the roles of each participant in the different phases of the event (2 marks)
- how these phases work (3 marks)
- what are the outputs of each phase (3 marks)

A ▼

B

I

T ▼

HP

U

x₂

x²

—

I

G

</>

#

Question 2

Not yet answered

Marked out of 5.00

What is a "Persona" and what is it used for (3 marks)? Give an example of a Persona (2 marks).

A ▼

B

I

T ▼

H-P

U

x₂

x²

G

</>

#

Question 3

Not yet answered

Marked out of 10.00

Below is an excerpt from a transcript of an interview at *PlucheMobile* where they explain part of their software development strategy.

Begin of transcript:

We usually work with a Kanban board where we put all our tasks linked to the user stories we are working on (one story typically has 5 to 10 tasks). All stories are also documented with acceptance criteria.

We have daily standups where we discuss the progress of the project and we have weekly releases with all finished stories being pushed to the development branch to be deployed later. For every task ready on the Kanban board, developers pull the latest development branch into a new branch. They start working on the task either by themselves, in pairs or the whole team together. When a task is finished, unless it has been worked on by more than one developer, the code is reviewed to ensure coding style is followed and then merged into the development branch if the code passes this syntactical check. If unit tests have been written, then the tests will be executed during the nightly build. The developer in charge will be notified of subsequent errors and will be responsible to roll their changes back from the development branch before fixing the issues.

The deployment team works on the release over the weekend. They pull the development branch from the Thursday's nightly build to update the master branch and then follow a manual script that has been updated during the week. If the deployment goes well, then the members of the development team run some smoke tests to confirm the success. If anything goes wrong, the deployment team rolls back to the prior version and document all unresolved issues for the developers to fix over the next week.

End of transcript

This development process has at least **5** major flaws that are either technical (i.e. the way the software product is managed, including its source code) or methodological (i.e. the way the team works). Identify them (**0.5 marks** per flaw) and for each flaw, give actual advice on how to improve the current strategy (**1.5 marks** per improvement). **Do not suggest to completely change their development process** (e.g., work with Scrum), but improve their current method instead.

A ▼

B

I

T ▼

H-P

U

x₂

x²

—

I

G

</>

#

Question 4

Not yet answered

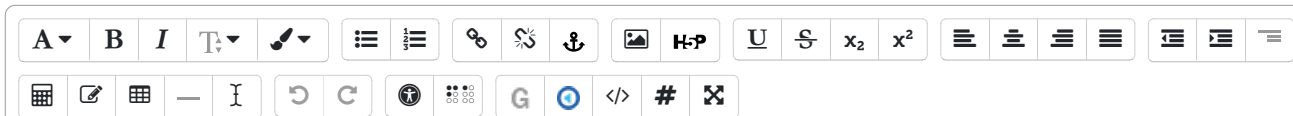
Marked out of 5.00

Specify the unit tests you would write for this method. You can express the tests either in plain English or in a mathematical form.

Your unit tests must specify **both the input and expected output**. Discuss any issues you may find with the code.

```
/**
 * This method returns the longest string from given string list
 * @param strings a list of strings
 * @return the longest string from given list, "None" if none is bigger.
 */
public String longestString(String[] strings) {
    String candidate = "";
    if (strings.length == 0) {
        return "None";
    }

    for (int i = 0; i < strings.length; i++) {
        if (strings[i].length() > candidate.length()) {
            candidate = strings[i];
        }
    }
    return candidate;
}
```



Question 5

Not yet answered

Marked out of 5.00

Write a review for this piece of code **as you would do when working on a merge request**. Consider that the unit tests you wrote in the previous question are part of the merge request.

Commit message:

implemented the method to find the longest string from a list of strings
added javadoc and tests

```
/**
 * This method returns the longest string from given string list
 * @param strings a list of strings
 * @return the longest string from given list, "None" if none is bigger.
 */
public String longestString(String[] strings) {
    String candidate = "";
    if (strings.length == 0) {
        return "None";
    }

    for (int i = 0; i < strings.length; i++) {
        if (strings[i].length() > candidate.length()) {
            candidate = strings[i];
        }
    }
    return candidate;
}
```

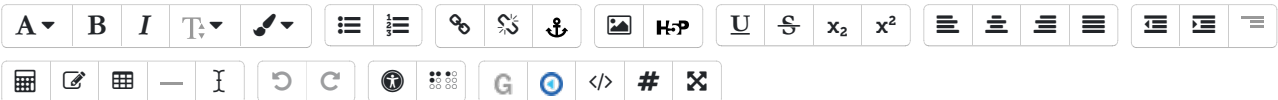


Question 6

Not yet answered

Marked out of 5.00

Explain what is testability (1 mark) and why it is important for TDDers (3 marks). You can make links between GeePaw Hill's and Kent Beck's article used in class (1 mark).



Question 7

Not yet answered

Marked out of 5.00

Explain what is the "4+1" views of software (2 marks). Explain why the "+1" is important for software development (2 marks) and how it relates to agile software development (1 mark).

A ▼

B

I

T ▼

H-P

U

x₂

x²

—

I

G

</>

#

Question 8

Not yet answered

Marked out of 5.00

In the ACM code of ethics, one of the principle requires to "Give a fair hearing to the opinions, concerns, or complaints of a colleague."

- What Scrum values does this relate to? Two values expected (1 mark).
- Which scrum event is particularly reliant on these two values (1 mark)?
- What are the main outcomes of that Scrum event (1 mark)?
- What framework is used to assess these outcomes? Give its full name and describe it in a few sentences (2 marks).

A ▼

B

I

T ▼

H-P

U

x₂

x²

—

I

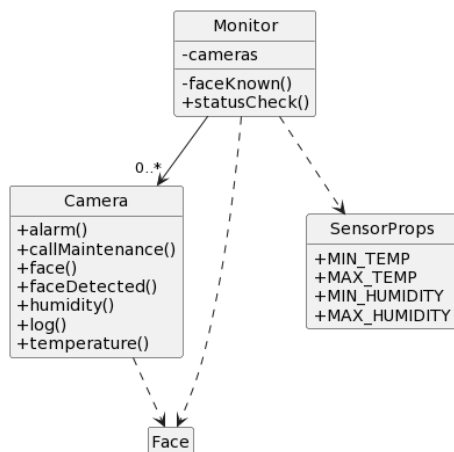
G

</>

#

Marked out of 7.00

ACME Home Security is developing a range of smart devices controlled by Java software. A camera unit contains a number of sensors. It can detect motion, and take a photo if it decides it can see a human face. It can also sound an alarm to scare off intruders. Other sensors record temperature and humidity. A camera can also contact ACME's maintenance division to request an engineer check it for faults. ACME's clients will have a number of cameras and a monitor unit that coordinates them (e.g. to follow an intruder moving around the premises. The software is still under development, and is incomplete, but preliminary versions of some classes (Camera, Face, Monitor, SensorProps) are available, via the links below, together with this initial class diagram.



Face.java

- A. [4 marks] What is your opinion of the design — in particular, the Camera and Monitor classes? If you identify any specific issues, then name them and explain their significance.
- B. [3 marks] Would you suggest any changes to the design? If so, clearly and concisely explain what they are, and what improvements they would provide. You may include modified Java code and/or a class diagram in your answer. Your code and/or diagram need not be perfect in every detail, but should indicate clearly the approach you are proposing.

If you wish, you may attach PDF documents and/or image files as part of your answer to this question. Please ensure that your name or userid appears on any files you attach and that file names clearly indicate which part of the question they refer to.

[illegible]

Maximum size for new files: 512 MB

Files



You can drag and drop files here to add them.

Accepted file types

Image (GIF) .gif

Image (JPEG) .jpg

Image (PNG) .png

PDF document .pdf

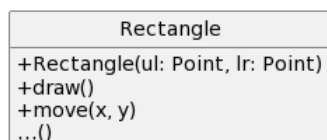
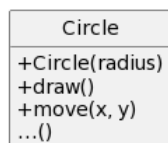
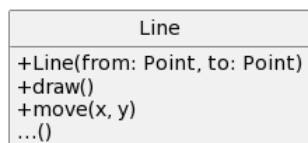
Text file .java

Question 10

Not yet answered

Marked out of 24.00

Dave is designing a drawing app, which is to be implemented in Java, and has partially completed the design. His design currently includes a number of classes and he has ideas for more. A drawing will consist of some number of shapes of various kinds. Although the details will differ, the corresponding objects will have some methods in common — as indicated in the diagram.



As well as primitive shapes, such as circles and rectangles, the application will have compound objects formed by grouping shapes together. A compound shape consisting of two circles and a rectangle is shown below. Although it is made up of a number of shapes, which may themselves be primitive or compound, a compound shape behaves — draw(), move(), select() etc. — like a single shape.



A. [12 marks] Dave is sure that using GoF design patterns will help him develop his design and asks you for advice.

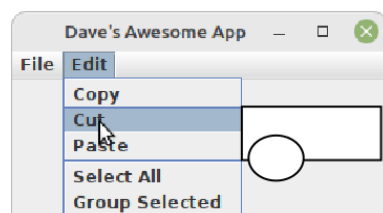
1. [2 marks] Which of the GoF patterns we have studied is the most appropriate here?
2. [2 marks] Explain how the elements of the pattern correspond to those of Dave's design.
3. [4 marks] What, if any, issues arise when implementing this pattern? For any you identify, discuss suitable options and explain which you prefer.
4. [4 marks] Use PlantUML to draw the corresponding class diagram. Include stereotypes and/or notes to highlight elements that have roles in the GoF pattern. **Attach your answer. The file name(s) should clearly indicate which part of the question it refers to.**

B. [10 marks] Dave is so impressed with your advice and knowledge of GoF design patterns that he asks you to help him with the next stage of his design. Although the shapes enable users to make nice drawings, he would like them to be able to add fancy embellishments, such as shadows and borders. A shape (primitive or compound) can have zero or more embellishments as shown here.



1. [2 marks] Which of the GoF patterns we have studied would be most appropriate for this?
2. [2 marks] Explain how the elements of the pattern correspond to those of Dave's design.
3. [2 marks] What, if any, issues arise when implementing this pattern? For any you identify, discuss suitable options and explain which you prefer.
4. [4 marks] Make a copy of your class diagram and use PlantUML to update it to include the new features. Include stereotypes and/or notes to highlight elements that have roles in the GoF pattern. **Attach your answer. The file name(s) should clearly indicate which part of the question it refers to.**

C. [2 marks] Dave plans to implement an edit menu. He shows you a rough prototype (see below) but there is currently no code to implement the various menu items. Suggest a GoF pattern that would be suitable for this. Briefly justify your choice and indicate how the pattern would be used. You need not document fully the correspondences between the pattern elements and the design, but should include sufficient detail to support your answer.



If you wish, you may attach PDF documents and/or image files as part of your answer to this question. Please ensure that your name or userid appears on any files you attach and that file names clearly indicate which part of the question they refer to.

A ▼ B I T ▼

x₂ x²

G </> #

Accepted file types

Image (GIF) .gif

Image (JPEG) .jpg

Image (PNG) .png

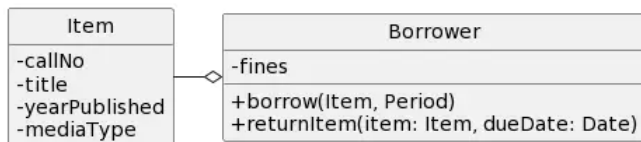
PDF document .pdf

Text file .java

Not yet answered

Big City Library's software team is designing a new software system, to be implemented in Java, and have identified some of the main elements they think they will need. The following design notes outline some of these:

- Borrowers are library members. For now they are all considered the same but later there may be different categories.
- The library has many items. Each item has a number of properties including title, (unique) call number, year of publication, and media type (book, DVD, magazine, ...)
- Borrowers can borrow items, and return, subject to some conditions. A standard loan is for a two week period. If an item is not returned within this period then the borrower will be charged a fine of \$1 per day until it is returned. A borrower can not borrow any further items if they have unpaid fines. A borrower who already has 10 items on loan can not borrow any further items.



A. [2 marks] In your own words, clearly and concisely explain what is meant by the contract of a method or class, and how a contract is expressed

B. [6 marks] Give suitable contracts for the `borrow()` and `returnItem()` methods of class `Borrower`.

C. [2 marks] In your own words, clearly and concisely explain what the relationship should be between the contracts of a method or class, and the corresponding contracts of its subclass(es).

D. [6 marks] In the next phase of the library's software system development, some additional categories of borrowers will be added—as subclasses of the original Borrower class—in order to better meet the needs of the community. The borrow() methods of the new categories are the same as that of Borrower with the following variations:

- StaffBorrowers are people who are employed by the library. They can borrow as many items as they want and don't accrue any fines if they return items late.
- JuniorBorrowers are at least 10 years old but are converted to Borrowers on their 18th birthday. They are only allowed to borrow 2 items at a time. Each time they borrow an item they receive a discount voucher for the local cinema.
- NonResidentBorrowers live outside the city. If they return items late, then they are fined at a rate double that of regular Borrowers.

Consider the contracts of Borrower and each of its subclasses. Are there any problems or issues with the way contracts have been implemented? Clearly and concisely explain your answer.

E. [3 marks] Can you suggest a better way to model the different categories of borrowers? Clearly and concisely explain.


If you wish, you may attach PDF documents and/or image files as part of your answer to this question. Please ensure that your name or userid appears on any files you attach and that file names clearly indicate which part of the question they refer to.


A ▼


B


I


T ▼






















U


~~S~~


x_2


x^2
































































Files





You can drag and drop files here to add them.

Accepted file types

- Image (GIF) .gif
- Image (JPEG) .jpg
- Image (PNG) .png
- PDF document .pdf
- Text file .java