Comp 47480 Written Paper 2025

Exam Advice

Here is some general advice for doing the COMP 47480 written exam paper.

Most of these are issues are things you should do in order to get the best grade possible on the paper.

Others are issues that make grading easier -- don't worry if you forget them in the pressure of doing the exam; I'll do my best with whatever you write.

Header Material

Time Allowed: 120 minutes

Instructions to Students:

Attempt all questions. Questions 1 and 2 carry 25 marks each, and all parts of these questions, (a), (b) and (c) carry equal marks. Question 3 carries 50 marks.

Note also:

No need to provide filenames or import statements in your answers.

Questions 1 and 2

Each question has three parts, (a) to (c). Answer all parts. Total 25 marks per question.

Advice: Attempt all parts and do not spend too long on a part that you find challenging.

Similar in style to parts of Q1 from previous years.

Question 3

Question 3 carries **50** marks. It has two parts, (a) and (b).

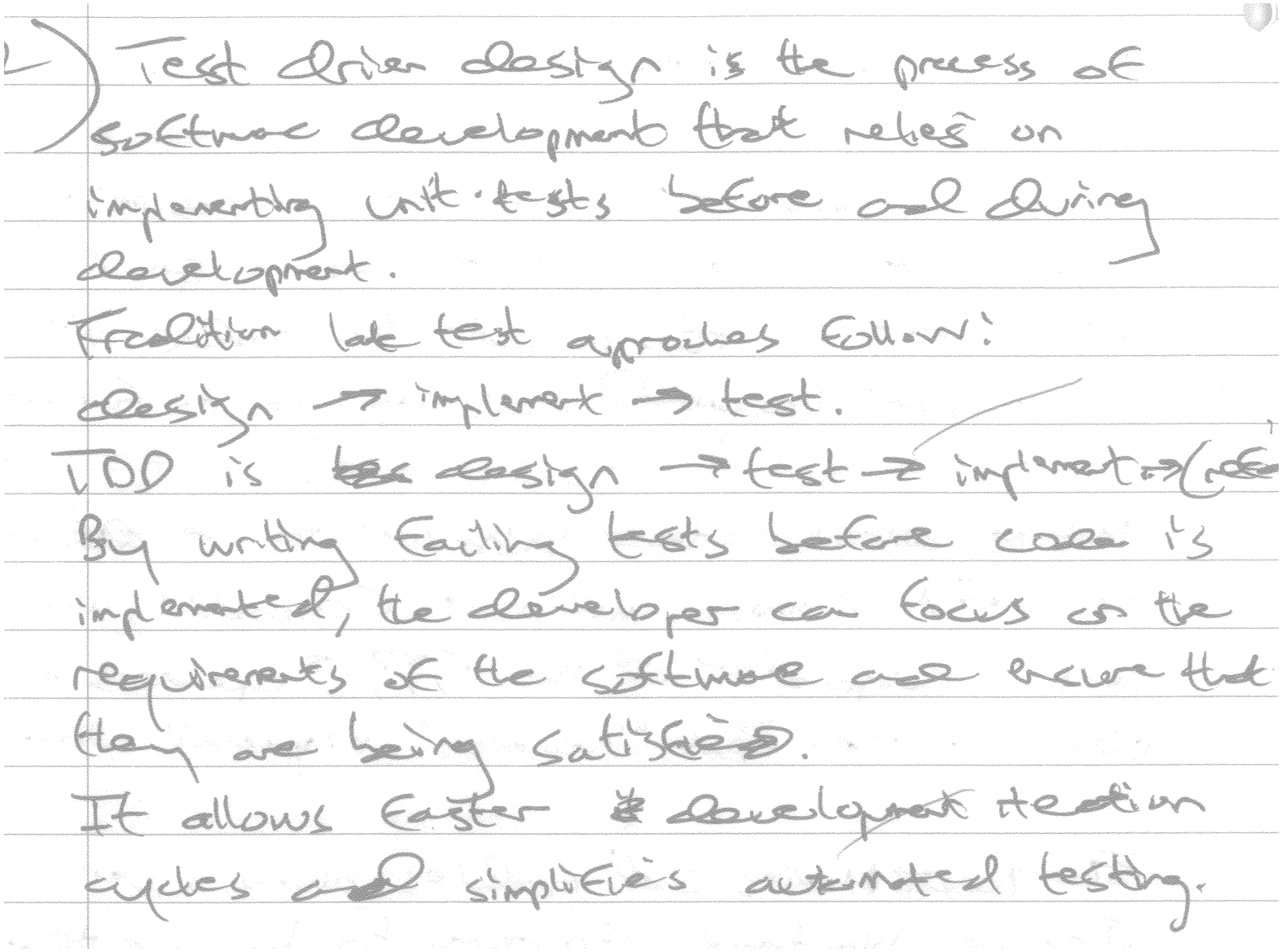
You are given a 3-page Java application to study.

In Part (a) you are asked to identify quality issues with the code, and for each identified issue you are asked to: **(i)** state what the issue is, **(ii)** explain the problems this issue may cause, and **(iii)** explain briefly how you might fix the issue.
Your answers should be tied to the application as much as possible.

Part (b) is related to design patterns.

Write Legibly

A fairly typical half-page in an exam answer book:



Write Legibly

… and that half-page typed:

“Test driven development is the process of software development that relies on implementing unit tests before and during development.

Traditional late test approaches follow:
design -> implement -> test

TDD is design -> test -> implement

By writing failing tests before code is implemented, the developer can focus on the requirements of the software and ensure that they are being satisfied. It allows faster development iteration cycles and simplifies automated testing.”

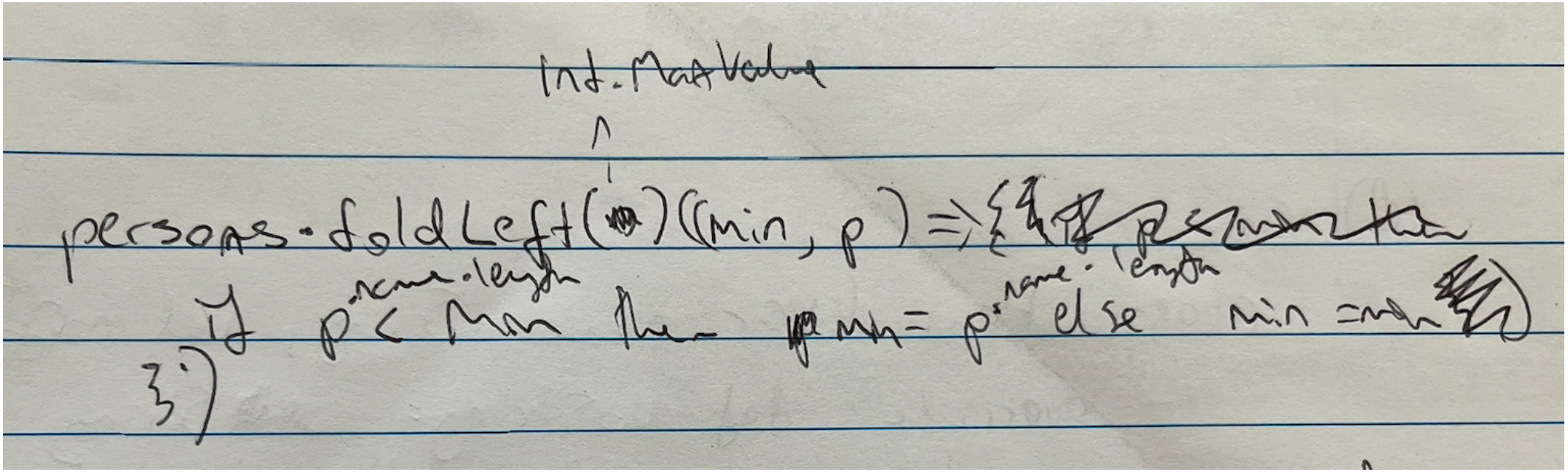
The answer is actually excellent, but it's obscured by the the hard-to-read handwriting.

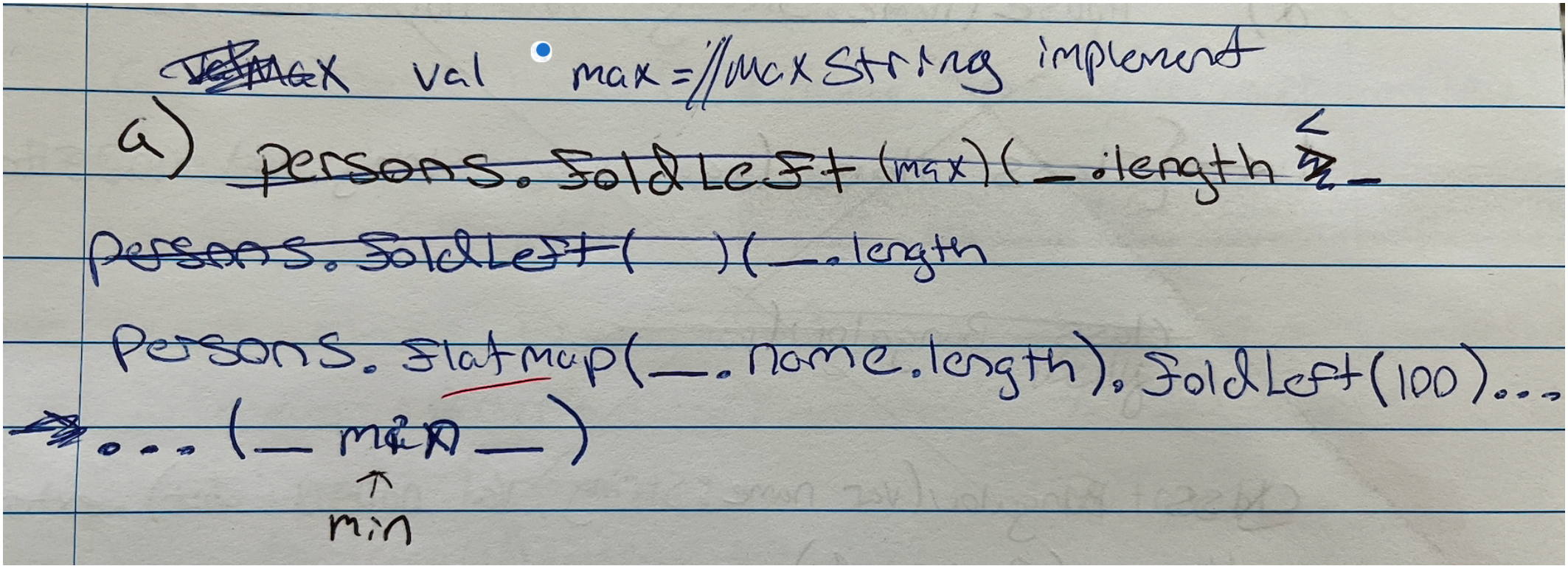
You are not asked to write a lot, so please take the time to write clearly.

My sample answers ran to four (typed, well-spaced) pages.

You have a 12-page answer book, which should be ample space.

Take extra care with code





The code on the previous slide fails to convey its meaning.

Suggestion: sketch any code in *Rough Work* first, then transcribe it to your answer book

You are only expected to write short snippets of code (if at all).

Sample Question from COMP30950

(h) (i) What is the *fragile base class problem*? **(ii)** Explain how the use of the `override` keyword in Scala helps in developing an inheritance hierarchy.

Firstly, it's in two parts **so make sure your answer is clearly in two parts as well**:

(h)

(i)

... ...

... ...

... ...

(ii)

... ...

... ...

... ...

And sample answer

i- fragile base class: when changes to the base class affect or override existing code in the subclasses that extend it. This is why we explicitly use the override modifier.

ii - The override modifier:

1. prevents us from accidentally overriding something in the superclass.
2. prevents us from accidentally NOT overriding something in the superclass (ie, method overloading instead)
3. prevents changes in the base class being accidentally overridden (fragile base class problem).

... and thus maintains inheritance hierarchies.

Avoid interspersing your answers to questions

For example, don't lay out your answers in this order:
Question 1(a)
Question 2(c)
Question 3(a)
Question 1(b)

Do it this way instead:
Question 1(a)
Question 1(b)
Question 1(c)
Question 2(a)
Question 2(b)
Question 2(c)
Question 3

1 page for each part of Questions 1 and 2 should be fine.

If you want to leave answering a question until later, just leave some blank space for the answer.

Questions during the exam

If you have a question about the exam brief itself, an invigilator will contact me, but please note:

I don't provide help/clarification to individual students.

When I'm asked a question, I'm thinking: "Should I interrupt every student to clarify/answer this?"

If I think not, then my answer will be the unhelpful "please work with the paper as it stands." Apologies in advance.

If there is a genuine error/ambiguity in the exam I'll adjust the grading of course.

So by all means ask questions! But appreciate they may not be answered.

Use your own examples

Use examples different from those seen in lectures.

If you are asked to e.g. to explain the Strategy pattern and you use the Commuter example from the lecture notes, you can get a "good" mark.

You can get an excellent mark if you use another example that shows that you have thought about the problem yourself, and perhaps read outside the course material.

The same applies to any question. **Use your own examples** to show that you really understand the material.

Grade-Aware Study Advice

In order of importance, drive your study using:

1. Past Exam Papers. This is material that you know I thought was important. Use (2) and (3) to help.

2. Lecture Slides. The lecture slides summarise the material we covered in the course. Use (3) to help.

3. Recommended Textbook. This is an excellent source of further reading for this module.

4. Internet/Books. There's plenty of information about software development available online and in textbooks.

Finally

Good luck on the exam!

If you have any questions about the module material, drop me an email. I'll try to answer promptly.