

--	--	--

**Semester Two 2017
Examination Period****Faculty of Information Technology**

EXAM CODES: FIT2107

TITLE OF PAPER: SOFTWARE QUALITY AND TESTING - PAPER 1

EXAM DURATION: 2 hours writing time

READING TIME: 10 minutes

THIS PAPER IS FOR STUDENTS STUDYING AT: (tick where applicable)

- | | | | | |
|------------------------------------|---|--|--|--|
| <input type="checkbox"/> Berwick | <input checked="" type="checkbox"/> Clayton | <input checked="" type="checkbox"/> Malaysia | <input type="checkbox"/> Off Campus Learning | <input type="checkbox"/> Open Learning |
| <input type="checkbox"/> Caulfield | <input type="checkbox"/> Gippsland | <input type="checkbox"/> Peninsula | <input type="checkbox"/> Monash Extension | <input type="checkbox"/> Sth Africa |
| <input type="checkbox"/> Parkville | <input type="checkbox"/> Other (specify) | | | |

During an exam, you must not have in your possession any item/material that has not been authorised for your exam. This includes books, notes, paper, electronic device/s, mobile phone, smart watch/device, calculator, pencil case, or writing on any part of your body. Any authorised items are listed below. Items/materials on your desk, chair, in your clothing or otherwise on your person will be deemed to be in your possession.

No examination materials are to be removed from the room. This includes retaining, copying, memorising or noting down content of exam material for personal use or to share with any other person by any means following your exam.

Failure to comply with the above instructions, or attempting to cheat or cheating in an exam is a discipline offence under Part 7 of the Monash University (Council) Regulations.

AUTHORISED MATERIALS

OPEN BOOK	<input checked="" type="checkbox"/> YES	<input type="checkbox"/> NO
CALCULATORS	<input type="checkbox"/> YES	<input checked="" type="checkbox"/> NO
SPECIFICALLY PERMITTED ITEMS if yes, items permitted are:	<input type="checkbox"/> YES	<input checked="" type="checkbox"/> NO

Candidates must complete this section if required to write answers within this paper

STUDENT ID: _____

DESK NUMBER: _____

INSTRUCTIONS TO STUDENTS

Please answer the questions in the provided script book. If you run out of room, ask for another.

Please do not use red pen.

There are a total of 80 marks available in this exam. The number of marks allocated to a question is a *rough* guide to how long you should spend on it.

Good luck!

Question 1: Aspects of quality (6+6 = 12 marks)

A new system, MYDAM, has been introduced at a University very much like Monash, for lecturers and tutors to mark assignments. Unfortunately, there have been a lot of problems with the system. You hear a lecturer say:

“This new MYDAM thingamy is a complete disaster! It takes over a minute to save a student’s assignment marks after you hit the submit button – and that’s if you can figure out the ridiculous marking interface – none of my tutors were able to use the system without help. Even when you enter the marks correctly, when you ask for total marks for calculating the final subject mark it rounds up rather than to the nearest integer, as university policy says we should. And, just yesterday, the server crashed and lost all the marks that were entered that morning!”

- a. There are multiple software quality problems described in this lecturer’s comments. Identify the distinct quality problems, and classify them using the ISO/IEC 9126 classification scheme. Briefly justify your answer. **(6 marks)**

For your information, the top level ISO/IEC 9126 quality properties are:

- Functionality
- Reliability
- Usability
- Efficiency
- Maintainability
- Portability

“It takes over a minute to save a student’s assignment marks after you hit the submit button” – efficiency (would also consider usability with a justification), because it relates to time efficiency.

“the ridiculous marking interface” – usability, because it was hard to learn and learnability is part of usability.

“mark it rounds up rather than to the nearest integer” – functional correctness

“server crashed and lost marks” – reliability (but would accept functional correctness with reasonable justification), because it relates to fault tolerance/recoverability.

If a good case is made for something else, accept.

1 mark for each correct classification, 0.5 marks for justification.

- b. For each problem, identify two quality assurance methods that could have been applied to reduce the risk of this problem occurring in production use. For each method you identify, briefly explain HOW it could be used to reduce this risk. **(6 marks)**

Would accept design/code review for any of them.

Other plausible answers for the ones identified above:

- i) Performance testing/benchmarking
- ii) Usability testing/paper prototyping
- iii) Unit testing/system testing

iv) Unit testing/ acceptance testing/system recovery testing etc.

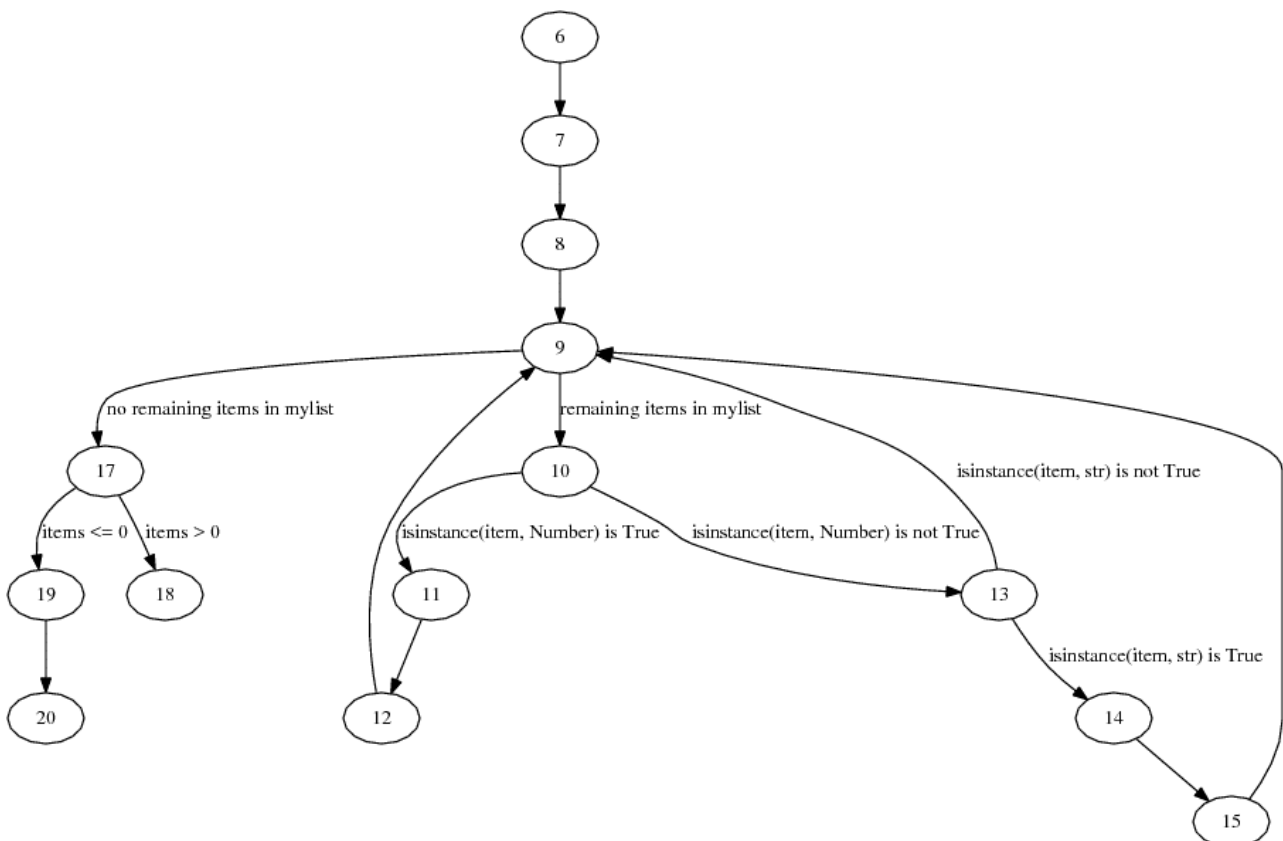
0.5 mark for each plausible method, 0.25 for brief explanation.

Question 2: White Box Testing (6+5+5+4 = 20 marks)

```
1  from numbers import Number
2
3  class NoValidMeanException(Exception):
4      pass
5
6  def listmean(mylist):
7      total = 0.0
8      items = 0
9      for item in mylist:
10         if isinstance(item, Number):
11             total += item
12             items += 1
13         elif isinstance(item, str):
14             total += len(item)
15             items += 1
16
17     if items > 0:
18         return float(total) / float(items)
19     else:
20         raise NoValidMeanException
21
```

- a. Draw a control flow graph for the listmean function above. Label each branch with constraints. **(6 marks)**

Sample solution



2 marks if they have something that looks like boxes and lines.

4 marks: something that has boxes and lines that resembles the control flow of the actual program, but with substantial errors, or major mistakes in constraint labelling.

6 marks: clear syntax, correct flow of control, constraints are correct.

If you're not sure where it fits, split the difference.

Note that there are potential correct variations – for instance, if they jump to line 15 or 16 before jumping back through the loop, accept, also accept if they combine sequential lines into composite nodes.

- b. Come up with the *smallest possible* set of test inputs that achieves *100% branch coverage* for the listmean function. Explain how your test inputs achieve the required coverage. **(5 marks)**

2 tests required - 1 where items >0 and 1 where items =

for loop, if, and elif can be covered in 1 test.

2 marks if it achieves 100% branch coverage (not necessarily minimal).

1 mark for minimal set.

2 marks for clear explanation demonstrating understanding of what branch coverage is and making clear how the tests achieve it.

- c. Is the set of tests you came up with in part b) a good set of tests for conducting unit testing? If so, explain why. If not, explain why not and describe a better set of tests. Take a few sentences to explain your answer. **(5 marks)**

0 marks: "yes" without further explanation.

0.5 marks: "no" without further explanation.

A "yes" answer that says simply "because it achieves coverage and coverage is good" or words to that effect is worth around 3 marks. Pay more if it's a really convincing explanation, but it would have to be good.

A "no" explanation that clearly explains that splitting up tests aids debuggability is worth 3 marks, 2 marks for a set of tests that does that. Pay partial marks on both criteria.

- d. Would achieving MC/DC coverage *for the entire function* require any additional tests? Explain why, or why not. Your answer should address all the requirements of MC/DC testing. **(4 marks)**

Assuming previous parts answered correctly, answer is "no".

MC/DC involves: branch coverage + condition coverage + all entry/exit points.

Branch coverage achieved. Condition coverage achieved because no compound conditions in function. Only one entry point in method, all exit points covered already.

4 marks: clear description that addresses all three criteria.

3 marks: clear, but addresses 2 out of 3 correctly, or average description of all three.

2 marks: gets at least 2 out of 3, may be badly written but sufficient to tell what answer is about.

1 mark: stating relevant criteria, or really poor attempt at applying.

0: equivalent to no answer.

Question 3: Black box testing (8 + 4 + 5 +5 = 22 marks)

Imagine that you have been assigned to develop an extension to a learning management system very similar to Moodle.

The extension “Python Script Assignment”, allows students to submit a Python script. When the Python script is uploaded, it is run with command-line arguments and other inputs specified by the subject lecturer.

Your current job is to test **the extended “create assignment” page** for creating a “Python Script Assignment”, as depicted below. Note that the date selectors make it impossible to select a non-existent date (such as the 50th of the month, or where the month is 13).

The screenshot shows a web browser window titled "Create Python Script Assignment". The address bar displays the URL `https://moodle.vle.monash.edu/create_python_script_assignment?`. The page content is as follows:

- Left Column (Irrelevant Content):**
 - Box 1: "Irrelevant to this question stuff", "More stuff", "Yet more stuff..."
 - Box 2: "More irrelevant stuff", a list of items with asterisks: "'Twas brillig", "Slightly Toves", "Gimbling Wabe", "Mimsied Borogroves", "Mome Raths", "Jubjub Bird", "Frumious", "Vorpai Blade", "Maxome Foes", "Jabberwocks".
- Right Column (Form Fields):**
 - Title: "Create Python Script Assignment"
 - "Assignment name:" followed by a text input field.
 - "Available from:" followed by a date selector (//) and a calendar icon.
 - "Due date:" followed by a date selector (//) and a calendar icon.
 - "Maximum size (LOC):" followed by a text input field.
 - "Execution timeout (seconds):" followed by a text input field.
 - Buttons: "Create Assignment" and "Cancel".

Question continues on next page

- a) Describe a set of categories, and associated partitions (choices), for testing the “create assignment” page. You must explain these clearly. **(8 marks)**

Things to look for:

- * are the categories and choices clearly AND EXPLICITLY named and described?
- * are the partitions for each category disjoint?
- * are the categories “natural” (typically, we’d have a category per input field, but accept other sensible answers?)
- * do the categories cover all aspects of the form input?
- * do the choices cover sensible groups of valid and invalid data (eg strings, negative numbers)
- * do the choices cover the interaction between the date fields?

8 marks: an explicit, easy to understand, listing and description of reasonably-chosen categories and choices that would form a good basis for testing without significant modification.

6 marks: a set of categories and choices that would require minor modification for use for testing, either with one or two flaws (eg lumping all invalid inputs together in all categories) or a flawed but still understandable description of them.

4: a set of categories and choices that has substantial flaws (eg not addressing invalid inputs and oddly-chosen names), or is poorly described, and would need major revision before use, but is still on the right track.

2: something that shows some knowledge of categories and choices, and some appreciation of the problem domain, but nothing more.

If you’re trying to decide between two descriptions, split the difference.

- b) How many *test frames* would result from your categories and choices? Are there any *invalid test frames* (that is, test frames for which there are no possible inputs)? Explain your answer. **(4 marks)**

Answer: product of number of choices in each category. (2 marks for answer with brief explanation)

Validity of test frames depends on answer for part a). 2 marks for a correct and explained answer.

- c) For any *two* of your test frames, write down *test inputs* for them. Make clear how your inputs relate to a specific test frame. **(5 marks)**

1.25 marks each: concrete test inputs that come from the test frame.

2 marks: clearly explained how inputs are related to specific test frames.

- d) Name a technique that can be used to reduce the number of tests, if the number of test frames from category-partition testing is too large. In a paragraph or so, explain how it works *using your answer to the previous parts of the question as an example*. Note: you are not required to show the complete resulting set of test frames, though you may do so if it helps illustrate your explanation. **(5 marks)**

2 marks: combinatoric or pairwise testing.

3 marks: quality of explanation.

3 marks: clear and technically correct explanation that shows how it works on the categories and choices here

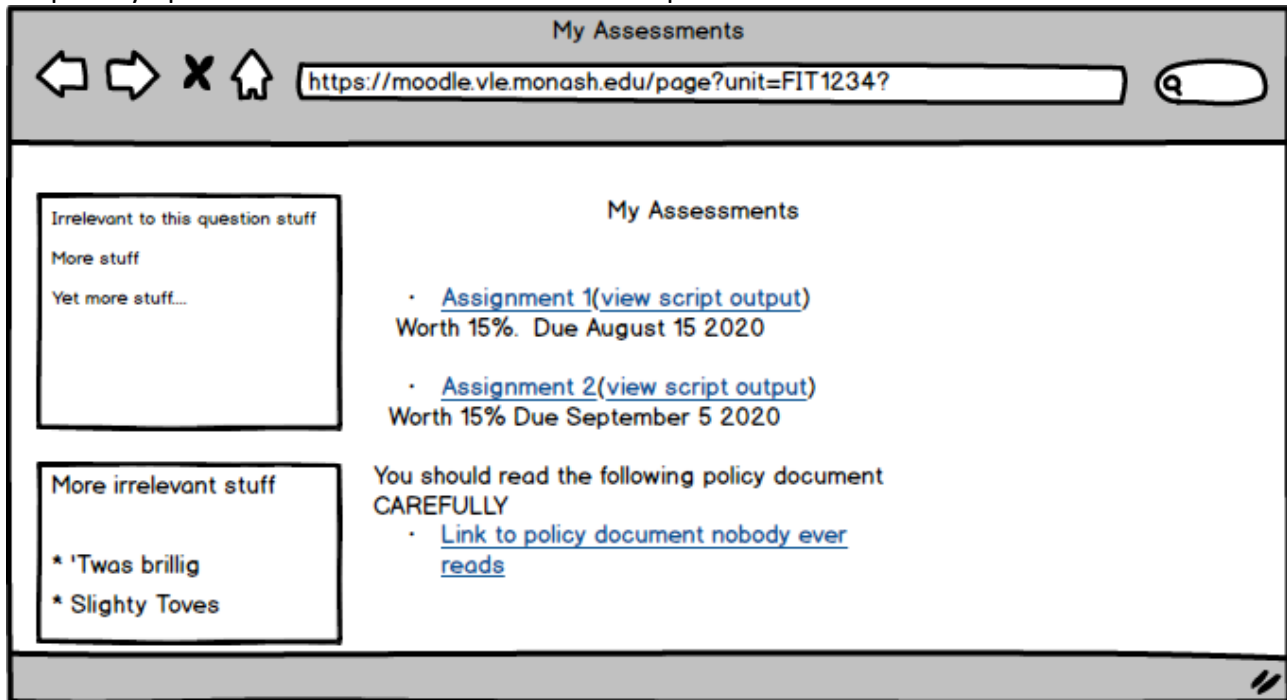
2 marks: some minor errors or not related to this example.

1 mark: some relevant points getting towards an answer made.

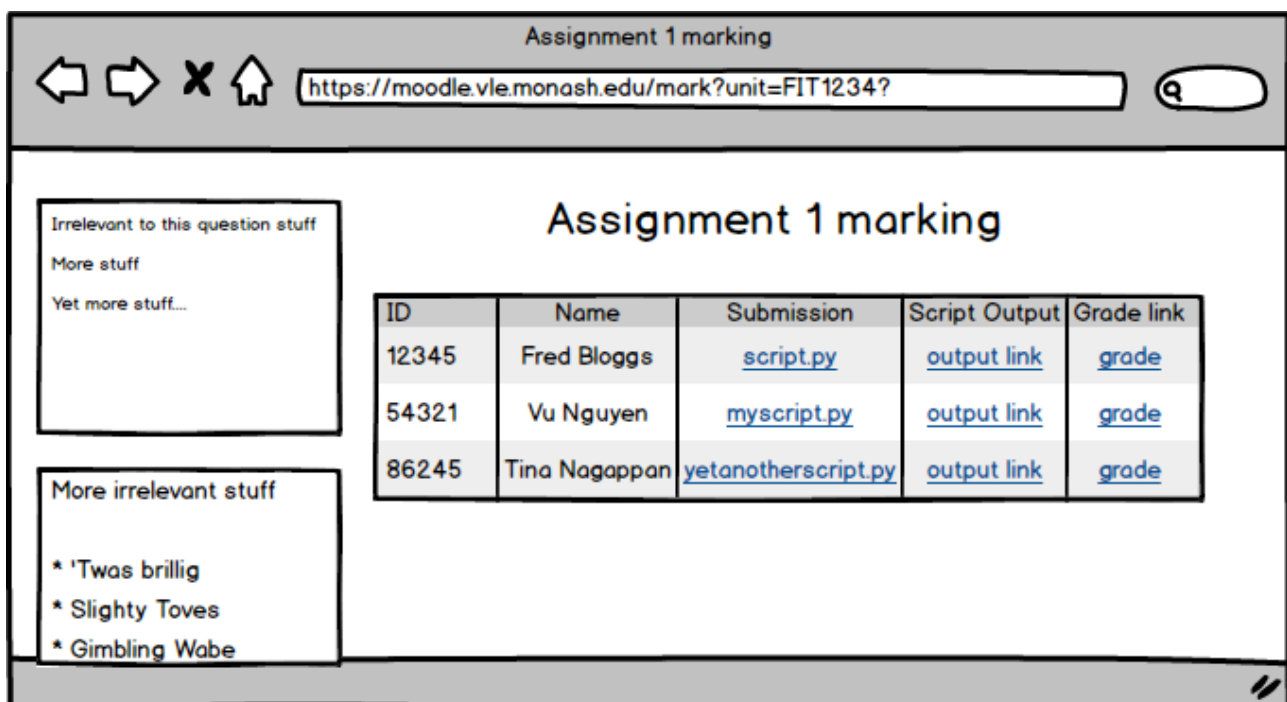
Question 4: Security (5 + 5 + 4 = 14 marks)

As part of your development of the “Python Script Assignment” extension to the Moodle-like system (as discussed in Question 3), you are reviewing the security of the extension.

On the subject page, each Python Script Assignment has two links: as well as the usual submission link to submit the assignment files, a second link entitled “view script output”. Clicking on it takes the student to a page which shows the output of the python script they uploaded with the lecturer’s selected inputs.

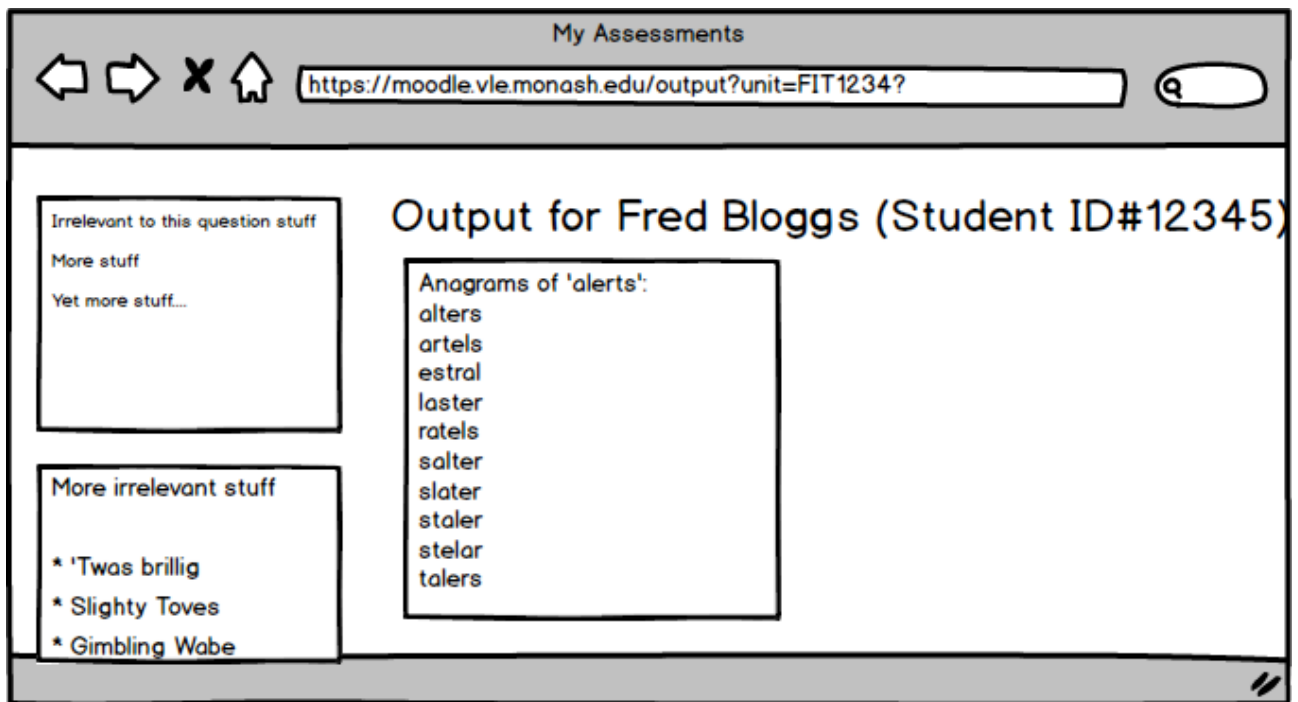


On the lecturer’s Assignment marking page, a list of students, links to download their Python scripts, and links to view the execution results are visible.



Question continues on next page

For both students and lecturers, clicking on the “view output” or “output link” takes them to a page that shows the output for the relevant student’s Python script:



- a) Describe a potential security problem with this extension that could result in data *stored on the server running the Moodle-like application* (i.e. student’s personal information) being stolen or tampered with. You should include a description of a *proof-of-concept* attack against *this specific extension* illustrating the potential problem. Your answer can include an example Python script, if you wish. **(5 marks)**

Suitable answers: command injection (injection not really necessary). Would accept other techniques including shellshock or mysql injection if well explained.

2 marks for identifying a suitable technique.

3 marks for description of proof-of-concept exploit (eg Python script reads contents of a file and emails back to a random address)

3 marks: clear description that provides enough detail to explain what the exploit could be used for, and how.

2 marks: acceptable description, but skips important steps or would need adaptation to apply here.

1 mark: non-zero knowledge about how to exploit demonstrated.

b)

3 marks: clear description that provides enough detail to explain what the exploit could be used for, and how. For 3 marks, MUST include printing cookie-stealing Javascript (would accept other workable solutions if clearly described).

2 marks: acceptable description, but skips important steps or would need adaptation to apply here.

1 mark: non-zero knowledge about how to exploit demonstrated.

- c) Describe a technique that could be used to reduce or eliminate the risk of the attack you have described in part b). Your answer must make clear how the technique works to prevent your proposed attack. **(4 marks)**

1 mark: naming a valid technique (eg HTML-only cookies or HTML escaping, or tag filtering).

3 marks for explanation:

3 marks: clear explanation of how the technique works relevant to problem (eg, for HTML escaping, describe how the HTML javascript tags would be escaped and therefore the script wouldn't run)

2 marks: vague, only partly correct, or otherwise significantly flawed description.

1 mark: non-zero amount of knowledge about the technique beyond name demonstrated.

Question 5: Metrics (6 + 6 = 12 marks)

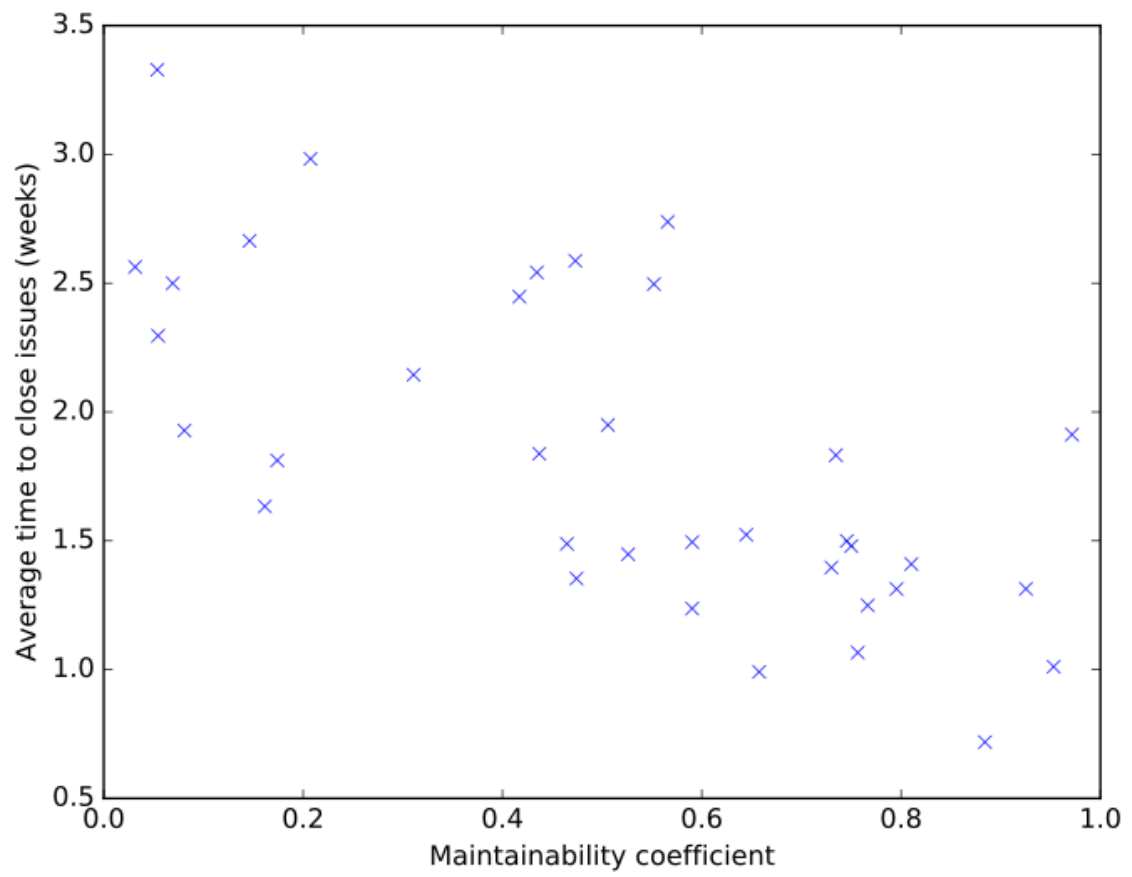
You are the Vice-President for New Software Development for the Moderately Big Corporation of Australia (MBCA). MBCA, as a moderately big IT-focused company, conducts many software development projects.

You are evaluating a new software engineering product called DesignRater that you are considering purchasing for your development teams. An advertisement for DesignRater reads as follows:

With our exclusive HugeData Analytics Technology (HAT), DesignRater can tell you in advance whether your new project is maintainable or not. All you need is a UML class diagram showing the design, and DesignRater gives you a Maintainability Coefficient from 0 (hard to maintain) to 1 (easy to maintain).

At MBCA, there is a policy that class diagrams are required to be maintained for all projects, and your bug tracker provides statistics on the time it takes for a bug to be closed after it has been initially reported.

Your assistant produces a graph of projects, where their position on the x axis indicates the maintainability coefficient, as calculated by DesignRater based on the project class diagrams, and the position on the y axis indicates the mean time (in weeks) to fix issues for the project:



Question continues next page

- a) Does the evidence summarised in the graph support the idea that the maintainability coefficient could be used to assess the quality of a design for a software product, before it is implemented? Explain why or why not. Take about half a page. (6 marks)

Relevant points in answer : Maintainability is generally an important quality property (though not the only quality property of importance). An important aspect of maintainability is the difficulty/cost of fixing bugs. Average time to close issue is a relevant metric for difficulty/cost of fixing bugs, and it appears there is a relationship between the maintainability coefficient and the time to fix bugs. Therefore, it could be used to *help* assess the quality of a design, but would not be a complete assessment of design quality.

6 marks: a clearly written answer that demonstrates understanding of:

- * quality
- * maintainability
- * what the metric purports to measure
- * what the graph shows
- * the limitations of the data.

Does NOT have to come to the same conclusions as mine if the arguments are of sufficient quality!

4 marks: an answer that makes relevant points and constructs some kind of argument but is weak on a couple of the following:

- * understanding of key terms
- * understanding of limitation of data.
- * quality of writing.
- * has some other substantial flaw.

2 marks: shows some understanding of the topic area but has major flaws such that the arguments are obviously wrong or are so poorly expressed that they cannot be clearly understood.

0: no answer, or no understanding of question demonstrated in answer.

If you can't decide between 2 categories, split the difference.

- b) The Microsoft maintainability index is an example of another metric that is used to assess the maintainability of software. In your own words, compare the maintainability index and the new Maintainability Coefficient calculated by DesignRater. Identify key differences, advantages and disadvantages of the maintainability coefficient as compared to the maintainability index. Take no more than half a page. (6 marks)

Key similarities: both are supposed to measure maintainability.

Key differences: Maintainability index requires source code, maintainability index can be done with a class diagram, so can be used at design stage. Validation of maintainability index done by polling managers, whereas maintainability coefficient based on bug-fixing times (which is more objective, but also more limited sense of maintainability). Maintainability index formula is public, so you know why your code is considered non-maintainable and so can work to avoid it, maintainability coefficient is a black box so only way to achieve low ones is rate design then try modifying.

Accept other reasonable points!

6 marks: an answer that coherently and clearly addresses many of the key points made above in a structured way.

4 marks: an answer that has aspects that are vaguely or poorly written, or misses some major points, but still summarises some of the key points about both methods.

2 marks: an answer that demonstrates non-zero understanding of the methods.

0 marks: equivalent to no answer.

If you're not sure, split the difference.

END OF EXAM PAPER