

University of St Andrews



DECEMBER 2016 EXAMINATION DIET

SCHOOL OF COMPUTER SCIENCE

MODULE CODE: CS5030

MODULE TITLE: Software Engineering Principles

EXAM DURATION: 2 hours

EXAM INSTRUCTIONS

- (a) Answer **three** questions.
- (b) Each question carries 20 marks.
- (c) Answer questions in the script book.

YOU MUST HAND IN THIS EXAM PAPER AT THE END OF THE EXAM.

**DO NOT TURN OVER THIS EXAM PAPER UNTIL
YOU ARE INSTRUCTED TO DO SO.**

1. Software Process and Testing.

- (a) Imagine you are a software manager responsible for the two projects listed below. For each project, suggest the most appropriate development process, and justify your choice. In addition, comment on the advantages and disadvantages of your chosen process.

- (i) Project I: develop the critical control software for a radiation therapy machine that delivers controlled doses of radiation to tumor sites.
- (ii) Project II: develop a mobile application for a wearable device to track users' diet and exercise data.

[4 marks]

- (b) List and describe any four testing activities for the mobile application developed in Question 1(a)(ii), based on the paper that we have covered in the lecture: *"Mobile Application Testing: a Tutorial, by Jerry Gao et al, published in IEEE Computer Magazine 2014 February, 46-55"*. [4 marks]

- (c) What is test-driven development? Describe its fundamental steps/activities with the aid of a diagram. [4 marks]

- (d) One principle of agile methods is close customer involvement. Discuss the advantages of such a principle, and describe possible practical complications.

[2 marks]

- (e) A fundamental principle of traditional software engineering is *design for change*. What does it mean? Why is it discarded by extreme programming? Discuss the reason and how extreme programming manages changes. [3 marks]

- (f) What is partitioning testing? Design partitioning test cases for a program which evaluates whether a string is a *palindrome* or not. Recall that a palindrome is a word, sentence or sequence that reads the same forwards as backwards. For example, "madam" and "noon" are palindromes, but "fish" is not. [3 marks]

[Total marks 20]

2. Requirement Engineering and Dependability Engineering.

- (a) Using your knowledge and experience of the MMS (Module Management System) of the University of St Andrews, write a use-case diagram to capture the main functional requirements of how students manage their modules.

[4 marks]

- (b) Write a requirement scenario for submitting a piece of coursework for a certain module on MMS. The scenario should cover both normal and exception procedures. Also list the functional requirements that can be elicited from the scenario.

[6 marks]

- (c) Discuss the advantages and pitfalls of defining quantitative reliability specifications. Use the following reliability requirement as an example to illustrate your answer:

The fire detection and extinguishing system on an aircraft should be reliable.

You need to analyse this requirement, identify appropriate metrics, discuss the validation strategies, and finally rephrase it so that it is quantifiably measurable.

[10 marks]

[Total marks 20]

3. Advanced Software Engineering.

- (a) What are the principal factors that affect the cost of system reengineering?
[5 marks]
- (b) Under what circumstances might an organisation decide to scrap a system when the system assessment suggests that it is of high quality and of high business value? Identify four factors and expand the discussion.
[4 marks]
- (c) Define software metrics in software quality management. Introduce different types of system metrics and an example of each. Discuss why software metrics are, by themselves, insufficient to predict software quality.
[6 marks]
- (d) What process is most commonly used by extreme programming practitioners in program inspection? Discuss its strengths and limitations.
[5 marks]

[Total marks 20]

4. Software Management.

Consider Table 1 describing the tasks and milestones associated with a software project.

Task/Milestone	Duration (Days)	Dependency	Task/Milestone	Duration (Days)	Dependency
T1	4	-	T11	12	T6
T2	1	-	T12	6	T11
T3	6	T2	T13	13	M3
T4	3	T1, T3	T14	5	M3
T5	10	-	T15	8	M2, T12
T6	3	M1	T16	9	T13
T7	7	M1	M1	-	T5
T8	9	T4	M2	-	T8, T9
T9	3	T4	M3	-	T7
T10	1	T4	M4	-	T12, T14

Table 1. A software project is composed of a list of tasks and milestones.

- Draw a PERT chart showing the project schedule. [6 marks]
- According to the PERT chart you obtained in Question 4(a), identify the critical path of the project. [2 marks]
- Calculate the slack time for the tasks that lead to T12 and T15. [3 marks]
- Briefly describe the four submodels of the COCOMO II model. [4 marks]
- Consider the following statement: "Software reuse is common in current software industry, as the development cost of reusing the existing components or code is zero." Do you agree with this statement? Justify your answer. [3 marks]
- Explain why keeping all members of a group informed about progress and technical decisions in a project can improve group cohesiveness. [2 marks]

[Total marks 20]

***** END OF PAPER *****