

**CT210 Software Quality**

Answer Sheet for Multiple Choice Questions (1-20)

Candidate name \_\_\_\_\_

Degree Course: \_\_\_\_\_

To indicate your answer, circle the appropriate letter for each question. If you make a mistake, cross out the letter with a cross (X) and write the letter you want at the end of the row, for example:

Question 1	a	b	c	d	b
------------	---	---	---	---	---

Question 1	a	b	c	d	
Question 2	a	b	c	d	
Question 3	a	b	c	d	
Question 4	a	b	c	d	
Question 5	a	b	c	d	
Question 6	a	b	c	d	
Question 7	a	b	c	d	
Question 8	a	b	c	d	
Question 9	a	b	c	d	
Question 10	a	b	c	d	
Question 11	a	b	c	d	
Question 12	a	b	c	d	
Question 13	a	b	c	d	
Question 14	a	b	c	d	
Question 15	a	b	c	d	
Question 16	a	b	c	d	
Question 17	a	b	c	d	
Question 18	a	b	c	d	
Question 19	a	b	c	d	
Question 20	a	b	c	d	

**Answer sheet for Section A to be provided**

**Two Hours**

UNIVERSITY OF MANCHESTER  
INSTITUTE OF SCIENCE AND TECHNOLOGY

**CT210 Software Quality**

For candidates taking:

BSc IN COMPUTATION  
SECOND YEAR SESSIONAL

BSc IN COMPUTING SCIENCE  
SECOND YEAR SESSIONAL

BSc IN INFORMATION SYSTEMS ENGINEERING  
SECOND YEAR SESSIONAL

MEng, BEng IN SOFTWARE ENGINEERING  
SECOND YEAR SESSIONAL

MEng, BEng IN COMPUTER SYSTEMS ENGINEERING  
SECOND YEAR SESSIONAL

Wednesday 31 May 2000

9.30-11.30

Answer **all** the **multiple choice** questions in **Section A**  
and **one** question from **Section B**  
*20 marks is allocated to Section A and 25 marks to Section B*

The **answer sheet** for use with **Section A** should be handed in at the end of the examination

The use of electronic calculators is NOT permitted

**Note: Do not answer more than the required number of questions. Clearly**  
cross out anything you do not wish to be marked.

PTO

UMIST, 2000  
(2)

**SECTION A: You must answer ALL questions in this section. Select the correct answer and mark on the accompanying answer sheet.**

- 1) Which type of risk factor is most likely to cause problems for a software project developing commercial software?
  - (a) Inadequate user documentation
  - (b) Litigation expense
  - (c) Low productivity
  - (d) Cancellation of project
  
- 2) *Defect prevention* is defined as:
  - (a) Finding and fixing errors after insertion
  - (b) Finding and fixing errors before release but after insertion
  - (c) Finding and fixing errors after release
  - (d) Avoiding defect insertion
  
- 3) *Product quality* is defined as:
  - (a) Delivering a product with correct requirements
  - (b) Delivering a product using correct development procedures
  - (c) Delivering a product which is developed iteratively
  - (d) Delivering a product using high quality procedures
  
- 4) Which of the following is NOT a main reason to undertake software quality assurance activities:

- (a) Reduce software personnel turnover
- (b) Legal liability
- (c) Insistence by the user on a satisfactory software quality assurance programme
- (d) Marketing reasons

5) Which type of risk factor is most likely to cause problems for a software project which develops military software?

- (a) Unused or unusable software
- (b) Legal expenses
- (c) Excessive paperwork
- (d) High maintenance costs

(3)

- 6) The main goal of quality assurance is:
- (a) Set coding standards.
  - (b) Improve software project management
  - (c) Reduce the technical and programmatic risks in developing the software
  - (d) Specify corrective actions.
- 7) Software interoperability is:
- (a) The ability of a software system to work on different hardware platforms.
  - (b) The ability of a software system to work under different operating systems.
  - (c) The ability of a software system to exchange information with other software systems and to use the exchanged information.
  - (d) The ability to replace a software system with another software system that has similar functionality
- 8) With respect to software metrics, which statement is NOT true:
- (a) A *indirect measure* focuses on attributes of a project which can be measured by examining a process, product or resource
  - (b) A *direct measure* focuses on attributes of a project which can be measured by examining a process, product or resource
  - (c) *External attributes* are always measured indirectly
  - (d) *Lines of code* is a *direct* measurement
- 9) Which of the following statements is NOT true.
- (a) Coding standards address naming of constants.
  - (b) Coding standards address the number of errors encountered per 1000 lines of code.
  - (c) Coding standards address layout of code text.
  - (d) Coding standards address the use of program comments.

- 10) Measures for a project are given as:

Effort: 12  
 Cost: £24,000  
 Thousand lines of code: 600k  
 Defects: 120

What is the *productivity* of the project?

- (a) 0.1
- (b) 2000
- (c) 5
- (d) 50

(4)

- 11) Which of the following statements is NOT true:
- (a) A good design methodology should provide a clear division of design from implementation
  - (b) A good design methodology should not promote a top-down decomposition strategy.
  - (c) A good design methodology should encourage phased development of the software
  - (d) A good design methodology should help to minimise future maintenance.

12) *Formal Reviews* seek to:

- (a) Identify system faults, but not to attribute blame or seek solutions
- (b) Identify system faults, attribute the source of errors, but not seek solutions
- (c) Identify system faults, attribute the source of errors and seek solutions
- (d) Identify system faults, seek solutions, but not to attribute blame

13) Using the following table for function point weightings:

Factors	Weights		
	Simple	Average	Complex
Number of user inputs	3	4	6
Number of user outputs	4	5	7
Number of user inquiries	3	4	6
Number of files	7	10	15
Number of external interfaces	5	7	10

A system being developed has the following characteristics:

- Number of user inputs 10 (simple)
- Number of user outputs 7 (simple)
- Number of user inquiries 3 (average)
- Number of files 6 (average)
- Number of external interfaces 1 (complex)

The function point count for the system is:

- (a) 27
  - (b) 31
  - (c) 58
  - (d) 140
- 14) Which form of software development model is most suited to a system where all the requirements are known at the start of a project and remain stable throughout the project.
- (a) Waterfall model
  - (b) Incremental model
  - (c) Evolutionary model
  - (d) Spiral model

15) What type of software development model is shown in the following diagram:

- (a) Waterfall model
- (b) Incremental model
- (c) Evolutionary model
- (d) Spiral model



16) Which of the following statements is NOT true:

- (a) Requirements must be testable
- (b) Requirements must be concerned with system functionality only
- (c) Requirements must be complete
- (d) Requirements must be unambiguously stated

17) The following software configuration diagram states that:

- (a) Version 1.3 is replacing version 2.1
- (b) Version 1.3 is evolving in parallel with the versions of the new release 2.1
- (c) Version 1.3 is the combination of versions 2.1 and 1.4
- (d) Only one of versions 2.1 and 1.4 should be developed.

18) Which of the following is NOT part of a software quality assurance plan:

- (a) Reference documents
- (b) Configuration Action
- (c) Supplier Control
- (d) Customer Control

(6)

19) Who of the following is NOT usually present in a technical review

- (a) User
- (b) Quality Engineer
- (c) The programming tools supplier
- (d) Specialist with knowledge of the application

20) The following diagram shows that:

- (a) Specification is completed before delivery
- (b) Specification is not completed until delivery
- (c) Specification is part of delivery.
- (d) Specification is an ongoing activity.





**SECTION B: You must answer 1 question in this section**

21) Answer all parts

- (a) Explain the main differences between software review and software inspection or walkthrough. (6 marks)
- (b) *Lines of code (LOC)* and *function point counts (FPC)* are two measures of the size of a system. Explain the advantages and disadvantages of using these two metrics for measuring systems. (6 marks)
- (c) Produce a critical path network, showing the *earliest start times* and *latest start times* for each task, using the data in the table below.

Task code	Task name	Duration	Starts after completion of task(s)
PLAN	Plan project	3	
REQ	Capture requirements	8	PLAN
AGREE	Agree requirements with customer	2	REQ
DESIGN	Design system	10	AGREE
CODE	Code system	12	DESIGN
ID	Identify subcontractors	3	DESIGN
BUY	Buy-in subcontractor code	5	ID
INTEG	Integrate code and buy-in code	6	CODE, BUY
TRAIN	Train staff	5	DESIGN
REL	Release system	4	INTEG, TRAIN

(13 marks)

(8)

22) Answer all parts

- (a) What are the main *risk factors* which may be encountered in the development of software?  
(8 marks)
- (b) Give a suitable definition of *software quality* and briefly describe the rationale for your definition.  
(6 marks)
- (c) Why should an organisation be concerned about *software risk factors* and *software quality*?  
(11 marks)

23) Answer all parts

- (a) How does development of software differ from that of hardware from a quality viewpoint?  
(9 marks)
- (b) What are the main objectives of configuration management and version control?  
(6 marks)
- (c) What are 'throwaway' prototypes and 'evolutionary' prototypes? How can production of a prototype be controlled and why is this important?  
(10 marks)

**END OF PAPER**

-----  
1.5

1.4

2.2

2.1

1.3

1.1

1.0