



THE UNIVERSITY
of LIVERPOOL

JANUARY 2002 EXAMINATIONS

Master of Science : Year 1

SOFTWARE ENGINEERING

TIME ALLOWED : Two Hours

INSTRUCTIONS TO CANDIDATES

Answer **FOUR** questions

If you attempt to answer more than the required number of questions, the mark awarded for the excess question (i.e. the one with the lowest mark) will be discarded.



THE UNIVERSITY
of LIVERPOOL

- 1(a). Outline the symptoms of the so-called 'software crisis' of the 1960s that led to the emergence of software engineering as a discipline and describe what is generally understood by the term 'software engineering' nowadays. **(5 marks)**
- (b). With the help of a diagram, outline the classic 'waterfall' model of software development and list its merits and drawbacks. **(8 marks)**
- (c). Describe how the waterfall model may be adapted to incorporate a prototyping stage and state the advantages and disadvantages of such an approach. **(6 marks)**
- (d). Outline the problems that occurred with the software for the emergency call handling system of the London Ambulance Service that was operational for a short while in the early 1990s, and list the key findings of the inquiry that ensued. **(6 marks)**





THE UNIVERSITY
of LIVERPOOL

- 2(a)(i). Describe the notation used in data flow diagrams (DFDs). **(4 marks)**
- (ii). Describe the notation used in use-case diagrams of the Unified Modeling Language (UML). **(4 marks)**
- (iii). Compare and contrast the use of DFDs and use-case diagrams in the software development process. **(3 marks)**
- (b). Consider the following description concerning a system for an accommodation agency.

An accommodation agency system is to be developed which aims to help clients find suitable rented accommodation in a large metropolitan area. Each client provides the system with personal information (e.g. name, address, occupation, etc.) together with details of the type of accommodation sought (e.g. house, flat, bed-sit). Property owners provide the system with lists of vacant properties and details of properties previously vacant, but now occupied. On payment of a fee by the client, agency staff can initiate a search request and provide the client with a list of appropriate vacant properties.

Internally the system maintains a stored list of clients containing their personal information and details of the type of accommodation sought. This is updated when a new client registers with the agency. The system also maintains a stored list of properties which is updated whenever property owners provide information about newly vacant properties or about the occupation of previously vacant properties. When the client pays the fee this is duly processed and this information is stored as a field of the client's record. When agency staff initiate a search request, the client's payment status is checked and the client's desired accommodation details are used to construct appropriate search criteria. The search criteria are used together with the stored list of vacant properties to produce a list of potential accommodation addresses for the client to visit.

Using the DFD notation, represent the system as described above by drawing:

- (i). a context diagram; and **(4 marks)**
- (ii). a first-level decomposition diagram. **(10 marks)**

■



THE UNIVERSITY
of LIVERPOOL

- 3(a)(i). What do the requirements analysis stage and the software specification stage of software development have in common and what is the distinction between them? **(2 marks)**
- (ii). Outline the problems facing the analyst when eliciting requirements from the stakeholders of some proposed new software system. **(4 marks)**
- (iii). Identify and explain any occurrences of the so-called 'seven sins' in the following natural language fragment from a requirements document. **(4 marks)**

When an order contains different versions of a product or a pack consisting of multiple items, a variable should be used to record the number of distinct products in the order.

- (b). Below is a Z schema that describes the state space of a simplified system for recording the names of employees in a small organisation. The names of employees must be unique within the system and the abstract type NAME can be assumed, so that it need not be considered in any further detail here.

	Organisation	
employees	:	IP NAME
card employees	≤	100

- (i). Explain the components of this schema in your own words. **(3 marks)**
- (ii). Give a Z schema appropriate for specifying an operation `Hire`, which records the addition of a single extra employee. **(4 marks)**
- (iii). Give a Z schema appropriate for specifying an operation `Fire`, which removes details of an employee who has been dismissed. **(4 marks)**
- (iv). Give a Z schema appropriate for specifying an operation `IsAnEmployee`, which determines whether any particular named person is an employee or not. **(4 marks)**

■



THE UNIVERSITY
of LIVERPOOL

- 4(a)(i). List the possible types of module cohesion giving a brief description of each and show how they form a qualitative cohesion spectrum. **(8 marks)**
- (ii). State with reasons the type of cohesion which best fits each of the following two module descriptions. **(4 marks)**

PRINT-RENEWAL(in VEHICLE-REGISTRATION-NUMBER)

Given a vehicle registration number as input, this module retrieves from the database the name and address of the legal owner and then prints as output the appropriate vehicle licence renewal notice for sending to the legal owner.

PRINT-HTML(in SWITCH, in A-STRING)

This module prints the given input string wrapped inside HTML tags, the precise nature of which is dependent on the particular setting of the input switch. For example:

when SWITCH=1, it prints A-STRING as an HTML title;

when SWITCH=2, it prints A-STRING as an HTML heading;

when SWITCH=3, it prints A-STRING as a URL link.

- (b)(i). Transitions in UML state diagrams may be labelled with an *event*, a *guard*, and an *action*. Describe what is meant by these three terms and indicate the usual notation whereby all three can be incorporated in a transition label. **(4 marks)**
- (ii). Consider the following scenario of a car with a cruise control system and then draw a UML state diagram modelling its behaviour. **(9 marks)**

A cruise control mechanism on a car is designed to operate under the control of three buttons: 'on', 'resume' and 'off'. It is to be assumed that the car initially starts in a state with the engine not running. Switching the engine on takes the car into a state where the engine is running and the cruise control is inactive. Whilst in this latter state, if the cruise control 'on' button is pressed, the car enters a state where cruise control is active and the controller records the current speed and maintains the car at this speed. When either the accelerator or the brake is pressed, the cruise controller disengages but stores the speed setting; in effect, the cruise control mechanism is in standby mode. If 'resume' is pressed while in this standby state, the cruise control becomes active again and the car is automatically accelerated or decelerated back to the previously recorded speed. Pressing the cruise 'off' button when in the active or standby states, takes the car into the cruise control inactive state. Switching the engine off when cruise control is inactive, active or in standby state reverts the car back to its initial state. Button presses other than those described, have no effect, i.e. the car remains in whatever its current state is.



THE UNIVERSITY
of LIVERPOOL

5(a)(i). What is meant by a 'test frame' in the category-partition method of testing? (3 marks)

(ii). During an application of the category-partition method of testing three categories A, B, C are selected with choices a1, a2, a3 in category A, choices b1, b2, b3 in category B, and choices c1, c2, c3, c4 in category C. In the absence of any further constraints how many test frames would be generated? You should explain your calculation. (2 marks)

(iii). Suppose now that constraints are added, as represented by the following test specification in the notation of the Test Specification Language (TSL).

```
Category A:
  a1          [single]
  a2
  a3          [property Q]
Category B:
  b1
  b2
  b3          [property R]
Category C:
  c1
  c2          [if Q]
  c3          [if R]
  c4          [if Q and R]
```

Explain the meanings of the annotations in square brackets. (4 marks)

(iv). By itemising the valid test frames, determine how many would be generated in the presence of the constraints in the TSL specification given above. (7 marks)

(b). Let P be the following pseudo-code program, where $e1$, $e2$ and $e3$ represent boolean expressions and $s1$ and $s2$ represent simple statements.

```
1  start;
2  if e1
3  then s1;
4  else
5      while e2
6      begin loop
7          if e3
8          then s2;
9          end if;
10     end loop;
11 end if;
12 stop;
```

(i). Draw the control flow graph for program P . (4 marks)

(ii). Identify the branches of program P and, with the aid of its control flow graph, devise a set of test paths which, if executed, would achieve complete branch coverage of P . (4 marks)

(iii). Why might it not be possible to find test data to execute your test paths? (1 mark)