UNIVERSITY OF LONDON

IMPERIAL COLLEGE OF SCIENCE, TECHNOLOGY AND MEDICINE

EXAMINATIONS 2000

BEng Honours Degree in Computing Part II
MEng Honours Degrees in Computing Part II
BEng Honours Degree in Mathematics and Computer Science Part II
MEng Honours Degree in Mathematics and Computer Science Part II
for Internal Students of the Imperial College of Science, Technology and Medicine

This paper is also taken for the relevant examinations for the Associateship of the City and Guilds of London Institute
This paper is also taken for the relevant examinations for the Associateship of the Royal College of Science

PAPER C220=MC220

SOFTWARE ENGINEERING - DESIGN I

Thursday 18 May 2000, 14:00
Duration: 90 minutes
(Reading time 5 minutes)

Answer THREE questions

Paper contains 4 questions

Section A *(Use a separate answer book for this Section)*

1a Briefly distinguish Validation and Verification (V&V), and explain why the V&V process would benefit from making requirements measurable.

b Reformulate the requirement "the system shall be secure" into a form that is more measurable. Explain your approach.

c Distinguish inheritance and aggregation. Give an example to support your distinction.

d The following is the specification of Jack's working week. If Jack's bedside alarm does not ring, then he continues to sleep. If the alarm rings and it is not the weekend, then Jack wakes up and goes to work. If the alarm rings and Jack is feeling ill, then he continues to sleep.

i) Draw a decision table describing Jack's working week as specified above.

ii) Is the specification consistent? Complete? Expand the decision table in part (i) to explain your answer.

iii) In what way do decision tables complement data flow diagrams? How might data flow diagrams be extended to encompass that complementary information.

The four parts carry, respectively, 20%, 10%, 20%, 50% of the marks.

© University of London 2000

2a i) Briefly describe the kind of information that state transition diagrams model.

ii) Explain one problem with using state transition diagrams, and explain how statecharts address this problem.

b A real-time lecturer evaluation system behaves as follows. Students rate lecturers as either *poor*, *OK* or *excellent* (by pressing the appropriate button). By default, a lecturer's rating is *OK*. If a lecturer gives a good lecture, then the students rate the lecture as *OK*. If the lecturer gives a bad lecture, the students rate him as *poor*. If the lecturer gives the students exam hints in his lecture, they rate him as *excellent*, but only if the students have not been given the same hints before. While the lecturer's rating is *excellent*, he smiles, and while his ratings are *poor*, he frowns.

i) Draw a state-transition diagram that models the behaviour of the lecturer above.

ii) A lecturer is either depressed or happy, depending on his rating by students. When he gives a lecture and students rate him as *excellent* or *OK*, he is happy, but when they rate him as *poor*, he is depressed. By default, he is a happy lecturer. Extend the state transition diagram in part (i) into a statechart, taking into account the lecturer's mood.

c State transition diagrams are well suited to describing use cases involving one object. Suggest and briefly explain an alternative notation for describing use cases involving multiple objects.

The three parts carry, respectively, 30%, 50%, 20% of the marks.

Section B *(Use a separate answer book for this Section)*

3a What is a model of a schema? (Assume that the schema does not have any base sort, i.e. no symbol in square brackets after the schema name.)

For each of the following schemas describe what would constitute a model. For cases (i) and (ii) state how many models the schema has.

i)

```
 ____ Even ____
| n: N
|______________
| ∃ h:N. (n = 2*h)
|______________
```

ii)

```
 ____ Factorial ____
| f: N → N
|__________________
| ∀x:N. f(x) = x!
|__________________
```

iii)

```
 ____ UG[X] ____
| f: X → N
|________________
| ∀x:X. 18 ≤ f(x) ≤ 21
|________________
```

iv)

```
 ____ Firstyear ____
| UG[X]
| S: FX
|___________________
|
|___________________
```

v)

```
 ____ NewUG ____
| ΔFirstyear
| ΞUG[X]
| in?: X
|_______________
| in?∈ X ∧ in?∉S
| S'= S ∪ {in?}
|_______________
```

3b

i) Write out the schema NewUG in full without inclusions, Δ or Ξ.

ii) Given the following schema Employee write a finite bag schema Bag(Employee)[X] and explain what its models are.

```
 ____ Employee ____
| name: seqchar
| age: N
| salary: R
|__________________
| ∃ h: R. salary = h * age
|__________________
```

The two parts carry, respectively, 60%, 40% of the marks.

4a An e-commerce system enables people to buy a certain product over the Internet. The system allows a customer to inquire about the price of the product, and to buy one or more items. To buy an item a customer has to have an ID number of type **seqchar**. The system provides a customer with a new ID number during a registration operation and keeps a record of the IDs already allocated. A button "Cancel" can be clicked at any moment to cancel the purchasing process. At each purchase, the system updates the bank balance of the shop. The following schema *WebShop* specifies the basic parameters of the system, where **nat** denotes the type of natural numbers.

WebShop

totnumberitems: **nat**	(number of items when in full stock)
price: **nat**	(price per product)

i) Write a schema *SellingState*, which includes *WebShop*, and uses the variables *accountbalance* (the balance of the shop's bank account), *stock* (number of items left), *members* (record of IDs allocated). Include relevant constraints on the variables.

ii) Write operation schemas for the following operations. They all have to include the output *itemssold!*. Marks will be awarded for correct use of Ξ.

Register gives the current customer a new ID number and keeps record of it.

Cancel gives no item to the current customer and removes his/her ID.

iii) Write operation schemas for the following operations. They all include the input *ID?* and *numberitems?*, and the outputs *itemssold!* and *report!*. Marks will be awarded for correct use of Ξ.

BoughtOk sells the requested number of items and updates the shop's bank balance.

OutofStock gives no item because the shop does not have the number of items requested in stock.

NoMember gives no item because the customer is not registered.

4b

i) Consider the operation *Buying = BoughtOk* $\vee$ *OutofStock* $\vee$ *Nomember.* Is this operation total? Explain your answer.

ii) Show that *accountbalance = (totnumberitems – stock) * price* is an invariant of each operation schema in 4a (iii).

The two parts carry, respectively, 70% and 30% of the marks.