UNIVERSITY OF LONDON
IMPERIAL COLLEGE OF SCIENCE, TECHNOLOGY AND MEDICINE

EXAMINATIONS 2001

BEng Honours Degree in Computing Part II
MEng Honours Degrees in Computing Part II
BSc Honours Degree in Mathematics and Computer Science Part II
MSci 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 C223=MC223

CONCURRENT PROGRAMMING

Thursday 17 May 2001, 14:00
Duration: 90 minutes
(Reading time 5 minutes)

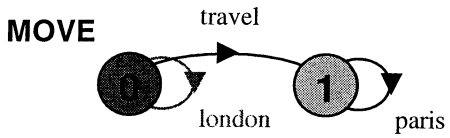
Answer THREE questions

Paper contains 4 questions
Calculators not required

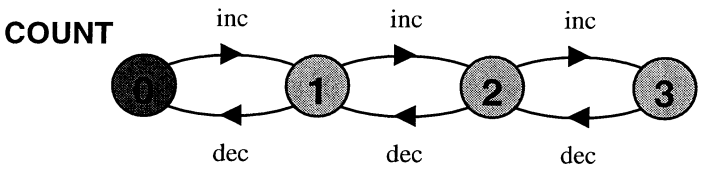
1a Explain how *non-deterministic choice* is expressed in the Finite State Processes (FSP) modelling notation. Briefly explain why it is useful in modelling systems.

b For each of the following Labelled Transition Systems (*LTS*), give an equivalent *FSP* specification.

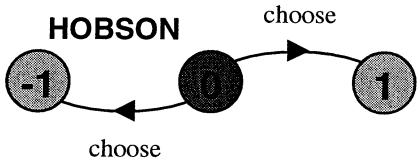
i)



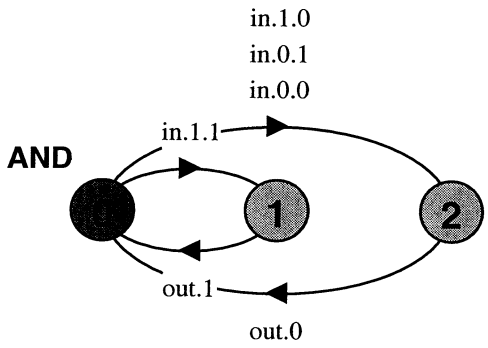
ii)



iii)



iv)



c For each of the following *FSP* specifications, give an equivalent *LTS*.

i) `DICE = (throw[i:1..6] -> (when (i==6) win -> DICE)).`

ii)
```
CLOCK(N=4)     = CLOCK[0],
CLOCK[i:0..N] = (when(i<N) tick[i] -> CLOCK[i+1]).
```

iii)
```
DOG     = (bark -> attack -> DOG).
||DOGS = (fido:DOG || rover:DOG)
          /{bark/{fido,rover}.bark}. // draw LTS for DOGS
```

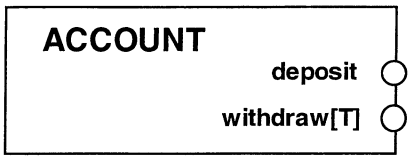
iv)
```
BROKEN  = STOP + {tick[2]}.
||ENDOFTIME = (CLOCK(20) || BROKEN).
```
//draw LTS for ENDOFTIME, CLOCK is as defined in ii) above.

The three parts carry, respectively, 20%, 40% , 40% of the marks.

2a Briefly explain how a *guarded action* in an FSP specification is translated into part of a Java program that implements that specification.

b A Special Savings Building Society Account is permitted to have a maximum balance of **M** hundred pounds. Savers may deposit one hundred pounds at a time into the account up to the maximum. They may withdraw money in multiple units of a hundred pounds so long as the account is not overdrawn.

The alphabet of the process that models the savings account is depicted below, together with a definition of the meaning of each action.



range T = 1..M
deposit - deposit one hundred pounds.
This action is blocked if the balance would exceed **M**.
withdraw[T] - withdraw an amount in the range **T** hundred pounds.
This action is blocked if sufficient funds are not available.

Specify the behaviour of ACCOUNT in FSP.

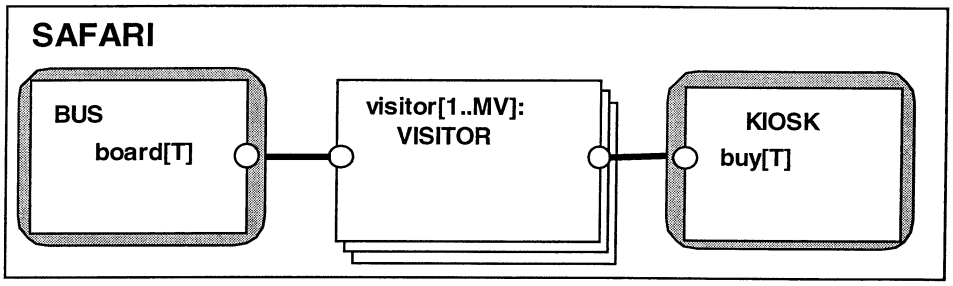
c Implement the ACCOUNT specification from part b with the two actions as monitor methods programmed in Java.

The three parts carry, respectively, 20%, 35%, 45% of the marks.

turn over .....

3a Briefly outline the two different ways of creating a new thread in Java.

b Visitors to a Safari Park buy a numbered ticket from a kiosk. Tickets are numbered in the range **T = 1..MT**. When ticket **MT** has been issued, the next ticket to be issued will be ticket numbered 1, i.e. the ticket vendor installs a new ticket roll. After buying a ticket, the visitor proceeds to the bus stop to board the bus that tours the park. The visitor may only board a bus when his/her ticket number appears on a large display. The structure diagram for a model of this system with **MV** visitors is shown below:



Given that the behaviour of VISITOR is defined by:

```
VISITOR = (buy[t:T] -> board[t] -> STOP).
```

specify the behaviour of each of the processes (BUS, KIOSK) and the composite process SAFARI in FSP.

c Implement the specifications for each of the model entities (BUS, KIOSK, VISITOR) in Java. Include the definition of a method `void build(int MV)` which creates the objects required for SAFARI.

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

4a Explain what is meant by *safety* and *liveness* properties with respect to concurrent programs.

b A university department of computing consists of a systems research group and a theory research group. Due to constraints on space, these groups share a meeting room. Due to mutual antagonism between the groups, to avoid conflict, the head of the department has decreed that the meeting room can contain only systems group members or theory group members but not both at the same time. Given the following definitions:

```
const M    = 3        //maximum number of people allowed in meeting room
const Max = 7
set Hackers    = {hack[1..Max]}  //members of the systems group
set Eggheads   = {egg [1..Max]}  //members of the theory group

ACADEMIC =      (enter -> meet -> exit -> ACADEMIC).
||DEPARTMENT = (Hackers:ACADEMIC || Eggheads:ACADEMIC).
```

Specify a process **MEETINGROOM** in *FSP* that ensures that a maximum of *M* people are allowed into the meeting room at any one time and that the meeting room cannot be occupied by both hackers and eggheads at the same time.

c Specify the follow safety properties in *FSP*:

i) **NOCONFLICT** checks that the meeting room is occupied either by hackers or eggheads, but not both simultaneously.

ii) **OVERFLOW** checks that more than *M* people do not occupy the meeting room at the same time.

Give the *FSP* composition for the system that combines the department, meeting room and the safety properties.

d Specify two progress properties in *FSP* that check, respectively, that hackers eventually get to use the meeting room and that the eggheads eventually get to use the meeting room. Give the specification for a system that models the situation in which there is a heavy demand for the meeting room. Would your progress properties be violated in this system?

The four parts carry, respectively, 10%, 30% , 40%, 20% of the marks