Two hours

UNIVERSITY OF MANCHESTER
SCHOOL OF COMPUTER SCIENCE

Concurrency

Date: Monday 19th May 2008
Time: 09:45 – 11:45

Please answer any THREE Questions from the FIVE questions provided
This is a CLOSED book examination

The use of electronic calculators is NOT permitted.
1. The Southern Sponge Bank (SSB) is taking over a network of ATM machines from another bank somewhat in need of financial rescue. An FSP model of their system was part of the documentation received at takeover. An abstracted version of it is reproduced below.

```
const N = 1
range Acc = 0..N

const T = 1
const F = 0
range B = F..T

set S = {monchester, leeverpule}

ATM = (cardIn[a:Acc] -> log[a] -> ack[r:B] ->
       ( when (r==1) displayService -> TRANSACTION[a]
         | when (r==0) consumeCard -> ATM ) ),

TRANSACTION[a:Acc] = ( withdrawCash -> TRANSACTION[a]
                         | cardOut -> deLog[a] -> ATM).

LOGGING =
           ( S.log[a:Acc] -> [a].use -> S.ack[T] -> LOGGING
             | S.deLog[a:Acc] -> [a].disUse -> LOGGING ) + {S.ack[B]}.

LOG = LOG[F],
LOG[v:B] = ( when (v==F) use -> LOG[T]
            | when (v==T) disUse -> LOG[F] ).

|| BANK = ( LOGGING || [Acc]:LOG ).
|| NETWORK = ( S:ATM || BANK ).
```

a) Provide informal descriptions of the FSP processes ATM, LOGGING, LOG, BANK and NETWORK. (5 marks)

b) The Board of Governors of the SSB are concerned over reports of their ATM network stopping working (the system analysts used the term “deadlocking”). Indeed, the analysts linked the problem to fraudulent use of duplicate ATM cards. Give a trace of actions that lead to a deadlock situation. (5 marks)

(Question 1 continues on the following page)
(Question 1 continues from the previous page)

c) The SSB system analysts realised that the deadlock problem related to the LOGGING and LOG processes comprising the BANK system. They issued a fix that introduced a new action, inUse, for the LOG process that could be used to determine whether an account is in use at some teller machine. Extend the capability of the current LOG process for this new action, and then make use of it in a revised LOGGING process. (6 marks)

d) Not content with the current level of security on the ATM network, the SSB governors insisted the analysts improve the ATM process to include the use of a user PIN, also stored on the ATM card. Sketch out the modifications necessary for the ATM process to perform such checking. You might assume a new action keypad that can be used to get the PIN from the user. (4 marks)
2.  
   a) Define \textit{trace-equivalence} and \textit{strong bisimilarity} for labelled transition systems and hence define \textit{strong bisimulation} for FSP processes. (4 marks)
   
   b) Define two FSP processes that have equivalent trace sets but are not strongly bisimilar. Justify your answer. Give a third process that can be used to distinguish the first two processes, i.e. its composition with each will yield processes that have different trace sets. (3 marks)
   
   c) Describe an algorithm for computing whether two FSP processes are bisimilar. (4 marks)
   
   d) Given the following FSP definitions,
   \[
   \begin{align*}
   P &= (a \rightarrow Q \mid b \rightarrow R), \\
   Q &= (b \rightarrow S), \\
   R &= (a \rightarrow P \mid b \rightarrow S \mid c \rightarrow T), \\
   S &= (b \rightarrow Q \mid b \rightarrow S), \\
   T &= (a \rightarrow P \mid b \rightarrow S \mid c \rightarrow R). \\
   A &= (a \rightarrow B \mid b \rightarrow C), \\
   B &= (b \rightarrow B), \\
   C &= (a \rightarrow A \mid b \rightarrow B \mid c \rightarrow D), \\
   D &= (a \rightarrow A \mid b \rightarrow E \mid c \rightarrow D), \\
   E &= (b \rightarrow E).
   \end{align*}
   \]
   give labelled transition diagrams for the processes \(P\) and \(A\), then use the algorithm given as answer to part (c) to show whether the process \(P\) is bisimilar to \(A\). (6 marks)
   
   e) How can a minimized version of an FSP process be computed? Give a minimised LTS representation for process \(P\) of part (d). (3 marks)
3. Consider the following FSP process definitions, which abstractly model a railway level crossing barrier and its control.

```
const Up = 1
const Down = 0
range S = Down .. Up

BARRIER = BARRIER[Up],
  BARRIER[s: S] =
    ( when (s == Up) close -> CLOSING
      | when (s == Down) open -> OPENING
      | fault -> STOP ),

  CLOSING = ( closing -> CLOSING
            | down -> BARRIER[Down] ),

  OPENING = ( opening -> OPENING
             | up -> BARRIER[Up] )\{opening,closing}.

CONTROL = INCONTROL,
  INCONTROL = ( onApproach -> close -> down -> OUTCONTROL ),
  OUTCONTROL = ( passedBy -> open -> up -> INCONTROL ).
```

|| SYS = ( BARRIER || CONTROL )\{up, down}.

a) Define the prefix action rule for FSP. Use the rule to provide a formal derivation of the onApproach transition that the CONTROL process can make. (2 marks)

b) Define the choice action rule for FSP. Now provide a detailed derivation of the close transition that the given BARRIER[Up] process can make. (3 marks)

c) Define the three forms of parallel composition rule for FSP. Using one of them, provide a detailed derivation of the close transition that the SYS process can make after the onApproach action. (5 marks)

d) Construct a labelled transition system that corresponds to the composite FSP process SYS. For information, this process contains 10 states. Identify the initial state of the labelled transition system you construct with its corresponding FSP process terms. (10 marks)
4. a) Briefly explain the following concepts:

i) The Java thread lifecycle

ii) Condition synchronisation in Java

iii) Synchronous and asynchronous message passing

iv) Monitors

v) Locks in Java (10 marks - 2 marks per part)

b) A Java class, EventLog, is required to implement a log of events that are sequentially emitted by a program, using a putEvent method, and which are consumed by a monitoring program, using a getEvent method. An EventLog object may be assumed to hold just a finite number of events. The following FSP models an EventLog that hold up to four of three types of events.

```plaintext
const N = 3
range Event = 1..N
set EventList = {[Event],
                 [Event][Event],
                 [Event][Event][Event],
                 [Event][Event][Event][Event]}

EventLog = ( putEvent[e:Event] -> EventLog[e] ),
            EventLog[h:Event] =
            ( putEvent[e:Event] -> EventLog[h][e]
             | getEvent[h] -> EventLog )
            EventLog[h:Event][t:List] =
            ( putEvent[e:Event] -> EventLog[h][t][e]
```

i) Describe what sort of communication is modelled through the process EventLog. (1 mark)

ii) Would you implement the EventLog as a passive or an active object? (1 mark)

iii) Give an outline Java class definition for the FSP EventLog process, assuming just two accessor methods, putEvent and getEvent. (8 marks)
5. a) Why are the phrases “Nothing bad will happen” and “Something good will happen” used, respectively, to describe safety and liveness properties? (2 marks)

b) Explain what is meant by a progress property? (2 marks)

c) What is the difference between the LIGHTS_PROPERTY property and LIGHTS process.

\[
\text{property}
\]
\[
\text{LIGHTS\_PROPERTY = ( red -> amber -> green -> LIGHTS\_PROPERTY ).}
\]
\[
\text{LIGHTS = ( red -> amber -> green -> LIGHTS ).}
\]

(4 marks)

d) Consider the following FSP model of a state machine for a traffic signal.

\[
\text{range } S = 0..3
\]
\[
\text{SIGNAL} = \text{SIGNAL}[0],
\]
\[
\text{SIGNAL}[s:S] =
\]
\[
\left( \begin{array}{l}
\text{when } (s == 0) \text{ red } -> \text{ SIGNAL}[1] \\
\text{when } (s == 1) \text{ amber } -> \text{ SIGNAL}[2] \\
\text{when } (s == 2) \text{ green } -> \text{ SIGNAL}[3] \\
\text{when } (s == 3) \text{ red } -> \text{ SIGNAL}[0] \end{array} \right).
\]

Construct the labelled transition system corresponding to the process

\[
( \text{SIGNAL} || \text{LIGHTS\_PROPERTY} )
\]

Does the SIGNAL process satisfy the LIGHTS\_PROPERTY? Provide an informal justification for your answer, for which, if “no”, you should include a counterexample trace. (10 marks)

e) Fix the SIGNAL FSP process definition so that it satisfies the desired LIGHTS\_PROPERTY property. (2 marks)