Two hours

UNIVERSITY OF MANCHESTER
SCHOOL OF COMPUTER SCIENCE

Concurrency

Date: Wednesday 27th May 2009
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. Consider the diagrammatic view below which depicts a storage cell with six connections denoted by in, out, put, get, amEmpty, and isEmpty.

The idea is that a cell may receive a value via an in action, and if empty, store the value. A cell that is storing a value may get rid of its value to the left (in the picture) through an out action, in which case it attempts to get a value from the right via a get action. A cell that is storing a value may also receive a value from the left, via another in action, and then pass it directly to the right via the put action.

An FSP definition CELL, listed below, is an attempt to describe such a cell.

\[
\begin{align*}
\text{range } \text{Val} & = 0..2 \\
\text{CELL}(I=2) & = \\
& ( \text{in}[v:\text{Val}] \rightarrow \\
& \quad \text{if} \ (I \neq 0) \ \text{then} \ \text{CELL}[v] \ \text{else} \ \text{END}[v] \\
& \quad \mid \text{amEmpty} \rightarrow \ \text{CELL} ), \\
\text{CELL}[v:\text{Val}] & = \\
& ( \text{out}[v] \rightarrow ( \text{get}[u:\text{Val}] \rightarrow \text{CELL}[u] \\
& \quad \mid \text{isEmpty} \rightarrow \ \text{CELL} ) \\
& \quad \mid \text{in}[u:\text{Val}] \rightarrow \text{put}[u] \rightarrow \text{CELL}[v] ), \\
\text{END}[v:\text{Val}] & = ( \text{out}[v] \rightarrow \ \text{CELL} ).
\end{align*}
\]

Such cells may be composed in parallel to form a chain of cells, effectively connecting the in and put actions, out and get actions, and amEmpty and isEmpty actions, of adjacent cells, pictorially shown as below.

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

a) The composite process \textit{CHAIN} below is a first attempt to model such a chain of connected processes. Explain how the FSP definition of \textit{CELL} is used to define the last cell in the chain, with only three actions. (3 marks)

\[ \text{CHAIN}(M=2) = \left( \forall [i:0..M-1] \ \text{c}[i]:\text{CELL}(M-i-1) \right) \]

b) Unfortunately, the given definition does not make any connections between the adjacent cells. Modify the definition of \textit{CHAIN}, using appropriate action relabelling, to achieve the intended connections. (4 marks)

c) Suppose a \textit{CHAIN} is created with 4 cells, simulate its behaviour for an input trace:-

\begin{align*}
c[0].\text{in[4]} \\
c[0].\text{in[2]} \\
c[0].\text{in[3]} \\
c[0].\text{in[1]} \\
\end{align*}

What sequence of values would be obtained through repeated use of the \textit{out} action of head cell, i.e. that labelled \text{c}[0]. (3 marks)

d) Make a simple modification to the definition of \textit{CELL} so that the composite \textit{CHAIN} process would act as an ascending sorter. For example, input of the sequence 3, 2, 1, 4 to the head of the chain would yield output (from the head of the chain) 1, 2, 3, 4. (5 marks)

e) There is an undesirable deadlock behaviour of the original \textit{CHAIN} process. Exemplify the situation in which it arises. (3 marks)

f) Outline what modifications to the \textit{CELL} process are necessary to eliminate the deadlock you’ve identified. (2 marks)
2. a) Define strong bisimilarity for labelled transition systems, and hence define strong bisimulation for FSP processes. (4 marks)

b) Define trace-equivalence for labelled transition systems. Give an example of two labelled transition systems that are trace equivalent but are not strongly bisimilar. (3 marks)

c) Outline and explain an algorithm for establishing whether two labelled transition systems are strongly bisimilar. (5 marks)

d) Consider the following two FSP processes $\text{DAY1}$ and $\text{DAY2}$ where

$\text{DAY1} = (\text{up } \rightarrow (\text{tea } \rightarrow W \mid \text{coffee } \rightarrow W)),
\text{W } = (\text{work } \rightarrow \text{sleep } \rightarrow \text{DAY1}).$

$\text{DAY2} = (\text{up } \rightarrow B),
\text{B } = (\text{tea } \rightarrow W1 \mid \text{coffee } \rightarrow \text{work } \rightarrow W2),
\text{W1 } = (\text{work } \rightarrow \text{sleep } \rightarrow \text{DAY2}),
\text{W2 } = (\text{sleep } \rightarrow \text{up } \rightarrow
\text{tea } \rightarrow \text{work } \rightarrow W2 | \text{coffee } \rightarrow W1)).$

After translating $\text{DAY1}$ and $\text{DAY2}$ to be labelled transition systems, use the algorithm you gave in answer to part c) to determine whether the process $\text{DAY1}$ is strongly bisimilar to $\text{DAY2}$. (6 marks)

e) Hence, or otherwise, define a minimized version of process $\text{DAY2}$. (2 marks)
3. Consider the following FSP process definitions which model a simple communication protocol.

\[
\begin{align*}
\text{SENDER} &= (\text{in} \rightarrow \text{snd} \rightarrow \text{ack} \rightarrow \text{SENDER}), \\
\text{RECEIVER} &= (\text{snd} \rightarrow \text{ack} \rightarrow \text{out} \rightarrow \text{RECEIVER}). \\
\text{COMMUNICATION} &= (\text{SENDER} \parallel \text{RECEIVER})\{\text{snd, ack}\}.
\end{align*}
\]

a) Using transition rules for FSP, provide a detailed derivation of the \text{snd} transition that the given \text{RECEIVER} process can make. Explain the semantic rules you use.  
(2 marks)

b) Now provide a detailed derivation of the \text{snd} transition that the \text{SENDER} process can make.  
(3 marks)

c) Give three semantic rules for parallel composition of FSP processes, and explain them. Hence, construct a detailed derivation of the \text{snd} transition that the process composition \text{SENDER} \parallel \text{RECEIVER} can make.  
(5 marks)

d) Give the semantic rule for hiding and then use it to derive the transition of the \text{COMMUNICATION} composite process that corresponds to the transition derived in c) above.  
(4 marks)

e) Using the transition rules for FSP, now build a labelled transition system that corresponds to the composite FSP process \text{COMMUNICATION}. Identify the states of the labelled transition system you construct with corresponding FSP process terms.  
(6 marks)
4. a) Briefly explain the following concepts:

i) An interleaving model of concurrency
ii) Mutual exclusion
iii) Synchronous and asynchronous message passing
iv) Java threads

(8 marks - 2 marks per part)

b) Consider the following FSP definitions that model a stack with operations push, top and pop.

range Item = 1 .. 3
set ItemList = { [Item],
                [Item][Item],
                [Item][Item][Item] }

STACK = ( push[i:Item] -> STACK[i] ),
STACK[i:Item] =
    ( push[j:Item] -> STACK[j][i] |
      top[i] -> STACK[i] |
      pop -> STACK ),
STACK[t:Item][r:ItemList] =
    ( push[j:Item] -> STACK[j][t][r] |
      top[t] -> STACK[t][r] |
      pop -> STACK[r] ).

The stack is to be shared between concurrent processes, as in, for example, the following FSP.

USERB = (top[i:Item] -> pop -> top[j:Item] -> STOP) +{push[i:Item],pop,top[j:Item]}.
||SYSTEM =
    (a:USERA || (a,b)::STACK || b:USERB).

A Java class, Stack, is required to implement the above model. There should be methods corresponding to the actions push, top and pop. The Stack object may be assumed to hold just a finite number of items.

Would you implement the STACK as a passive or active object? (1 mark)

Give an outline Java class definition for the FSP STACK process, with a constructor and three synchronised accessor methods corresponding to push, pop and top. You may assume the constructor is passed the maximum number of items to be stored in the stack. (11 marks)
5. a) Explain the difference between a blocked process, a deadlocked system and a livelocked system? (3 marks)

b) Explain what is meant by a safety property. Is freedom from deadlock a safety property? (2 marks)

c) A safety property is defined in FSP as a process definition prefixed by the keyword property as in the example below.

```
property
CYCLE_PROPERTY =
   ( wash -> rinse -> spin -> CYCLE_PROPERTY ).
```

Explain the difference between CYCLE_PROPERTY and the process CYCLE defined below, giving the labelled transition systems for both.

```
CYCLE = ( wash -> rinse -> spin -> CYCLE ).
```

(4 marks)

d) Consider the following FSP model of a washing machine.

```
range S = 0..3
MACHINE = ( on -> STATE[0] ),
STATE[s:S] =
   ( when (s==0) closeDoor -> STATE[s+1]
   | when (s==0) openDoor -> beep -> STATE[s]
   | when (s==1) wash -> STATE[s+1]
   | when (s==2) pause -> STATE[s]
   | when (s==2) rinse -> STATE[s+1]
   | when (s==3) pause -> PAUSE[s]
   | when (s==3) spin -> off -> MACHINE),
```

Draw the labelled transition system corresponding to the MACHINE process. (3 marks)

Now construct the labelled transition system corresponding to the parallel composition

```
MACHINE || CYCLE_PROPERTY
```

(3 marks)

Provide an informal justification, and counter example trace, of why the machine fails the wash cycle property, CYCLE_PROPERTY. (2 marks)

e) Fix the MACHINE FSP process definition so that it satisfies the desired wash cycle property. (3 marks)

END OF EXAMINATION