Communication and Concurrency
Exercise Sheet 1

The deadline for this coursework is 4.00pm on Monday 21st October. The answers should be submitted to the ITO where the answer to Q4 should also include a text file you have run on the CWB.

1. Consider the following vending machine and a user. *[12 marks]*

$$\begin{array}{lcl} \mathtt{V} & \stackrel{\text{def}}{=} & \mathtt{2p.V_b} + \mathtt{1p.V_l} \\ \mathtt{V_b} & \stackrel{\text{def}}{=} & \mathtt{big.collect_b.V} \\ \mathtt{V_l} & \stackrel{\text{def}}{=} & \mathtt{little.collect_l.V} \\ \mathtt{U} & \stackrel{\text{def}}{=} & \overline{\mathtt{1p}}.\overline{\mathtt{little}}.\mathtt{U} \end{array}$$

Let K be the set of actions $\{\mathtt{1p}, \mathtt{little}\}$ and let L be $\{\mathtt{1p}, \mathtt{little}, \mathtt{2p}\}$.

Recall that the flow graph of a process is a diagram with input arrows to *observable* actions a and output arrows from *observable* co-actions $\overline{a}$: so, actions that are restricted on do not occur in a flow graph.

For each of the following processes draw its flow graph and transition graph (with respect to the thin transitions $\xrightarrow{a}$).

   1. $\mathtt{V} \mid \mathtt{U}$
   2. $(\mathtt{V} \mid (\mathtt{U} \mid \mathtt{U}))\backslash K$
   3. $(\mathtt{V} \mid (\mathtt{U} \mid \mathtt{U}))\backslash L$

2. Assuming one datum value 0, draw the observable transition graph (with respect to the thick $\stackrel{a}{\Longrightarrow}$ and $\stackrel{\varepsilon}{\Longrightarrow}$ transitions) for the following process `Protocol`. *[12 marks]*

$$\begin{array}{lcl} \mathtt{S} & \stackrel{\text{def}}{=} & \mathtt{in}(x).\overline{\mathtt{sm}}(x).\mathtt{S1}(x) \\ \mathtt{S1}(x) & \stackrel{\text{def}}{=} & \mathtt{ms}.\overline{\mathtt{sm}}(x).\mathtt{S1}(x) + \mathtt{ok.S} \\ \mathtt{M} & \stackrel{\text{def}}{=} & \mathtt{sm}(y).\mathtt{M1}(y) \\ \mathtt{M1}(y) & \stackrel{\text{def}}{=} & \overline{\mathtt{mr}}(y).\mathtt{M} + \tau.\overline{\mathtt{ms}}.\mathtt{M} \\ \mathtt{R} & \stackrel{\text{def}}{=} & \mathtt{mr}(x).\overline{\mathtt{out}}(x).\overline{\mathtt{ok}}.\mathtt{R} \\ & & \\ \mathtt{Protocol} & \equiv & (\mathtt{S} \mid \mathtt{M} \mid \mathtt{R})\backslash\{\mathtt{sm}, \mathtt{ms}, \mathtt{mr}, \mathtt{ok}\} \end{array}$$

3. Consider the following CTL^- formulas: *[12 marks]*

   (1) $\texttt{AF}\ (\langle \texttt{b}\rangle \texttt{tt} \wedge [-][\texttt{b}]\texttt{ff})$

   (2) $\texttt{AF}\ ([\texttt{b}]\texttt{ff} \wedge \langle \texttt{c}\rangle \texttt{tt})$

   (3) $\texttt{AG}\ ([\texttt{b}]\texttt{ff} \vee \langle \texttt{c}\rangle \texttt{tt})$

   (4) $\texttt{AG}\ \langle a\rangle \texttt{tt}$

   (5) $\texttt{AG}\ ([\texttt{b}]\texttt{ff} \vee [-][\texttt{c}]\texttt{ff})$

   For each $i = 1, 2, 3, 4, 5$: either give a process E_i such that $E_i \models (1) \wedge (2) \wedge \ldots \wedge (i)$, that is satisfies *every* property from (1) to (i), or give a detailed argument explaining why such a process cannot exist. The processes you give should be as simple as possible. (In particular, if you find a process E_5 and then take $E_1 = E_2 = E_3 = E_4 = E_5$ you will lose points, because E_1, say, will be more complicated than necessary.)

4. Your answer to this question should include a text file that you have checked with the CWB. A very slight variant of the process `Crossing` defined in the lectures is as follows: *[14 marks]*

$$\begin{array}{lcl}
\texttt{Road} & \stackrel{\text{def}}{=} & \tau.\texttt{car}.\texttt{up}.\overline{\texttt{ccross}}.\overline{\texttt{down}}.\texttt{Road} \\
\texttt{Rail} & \stackrel{\text{def}}{=} & \tau.\texttt{train}.\texttt{green}.\overline{\texttt{tcross}}.\overline{\texttt{red}}.\texttt{Rail} \\
\texttt{Signal} & \stackrel{\text{def}}{=} & \overline{\texttt{green}}.\texttt{red}.\texttt{Signal} + \overline{\texttt{up}}.\texttt{down}.\texttt{Signal} \\
 & & \\
\texttt{Crossing} & \stackrel{\text{def}}{=} & (\texttt{Road} \mid \texttt{Rail} \mid \texttt{Signal})\backslash\{\texttt{green}, \texttt{red}, \texttt{up}, \texttt{down}\}
\end{array}$$

   Show that `Crossing` satisfies the following properties

   - No deadlocks: $\texttt{AG}\ (\langle - \rangle \texttt{tt})$
   - Safety: $\texttt{AG}\ ([\overline{\texttt{ccross}}]\texttt{ff} \vee [\overline{\texttt{tcross}}]\texttt{ff})$
   - It is possible that `car` never happens: $\texttt{EG}\ [\texttt{car}]\texttt{ff}$
   - It is possible that `train` never happens: $\texttt{EG}\ [\texttt{train}]\texttt{ff}$

   However, it can be the case that cars or trains are delayed forever: a train can be delayed forever because of cars crossing continuously, and a car can be delayed forever because of trains crossing continuously.

Modify `Crossing` in such a way that:

- It still has the same *observable* actions
- All the properties above still hold; show that they do hold for your modified solution
- Trains cannot be delayed forever: If a train is waiting at the crossing, then the train will eventually cross. This is captured by the property:
  - Trains will eventually cross: $\texttt{AG}\ [\texttt{train}]\ \texttt{AF}\ \langle\overline{\texttt{tcross}}\rangle\texttt{tt}$.
    (Actually, this property states that if a train is waiting at the crossing, then it will be eventually given the oportunity to cross, but for this system this is good enough.)

  Show that your modified solution does indeed have this property.