Consider two queues, $Q_1$ and $Q_2$, of capacity one that are put in sequence (see Section 5.5 p.60 in [Fok00]). Let $\Delta$ be a finite set of data elements. Queue $Q_1$ reads a datum $d \in \Delta$ from a channel 1 and sends this datum into channel 3. Queue $Q_2$ reads a datum $d \in \Delta$ from a channel 3 and sends this datum into channel 2 (the system is depicted on p.60 in [Fok00]).

The two queues are defined by recursive specifications:

$$Q_1 = \sum_{d \in \Delta} r_1(d) \cdot s_3(d) \cdot Q_1$$

$$Q_2 = \sum_{d \in \Delta} r_3(d) \cdot s_2(d) \cdot Q_2$$

where action $r_i(d)$ represents reading a datum $d \in \Delta$ from channel $i$, action $s_i(d)$ represents sending datum $d \in \Delta$ into channel $i$, and $\sum_{d \in \Delta} t(d)$ denotes the alternative composition of process terms $t(d)$, for all elements $d \in \Delta$.

The communication function $\gamma$ is defined by:

$$\gamma(s_3(d), r_3(d)) = c_3(d)$$

where action $c_3(d)$ represents communication of datum $d$ via channel 3 (all other communications between atomic actions result to $\delta$).

The overall behavior of the system is described as the term

$$\tau_{\{c_3(d)\mid d \in \Delta\}}(\partial_{\{s_3(d), r_3(d)\mid d \in \Delta\}}(Q_2\parallel Q_1)).$$

1. Give the specification of the two queues with $\Delta = \{d1, d2\}$ as a process declaration in $\mu$CRL language.

2. Use $\mu$CRL tool set to produce the process graph that belongs to the process declaration from part 1.

3. Use $\mu$CRL tool set to show that the system does not contain any deadlocks.