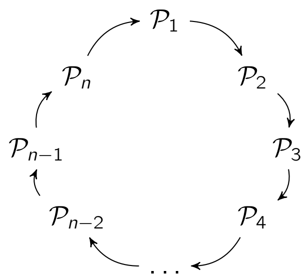
Exercise 1 (Channel Systems): (3 points)

Consider the following leader election algorithm: For $n \in \mathbb{N}$, n processes $\mathcal{P}_1, \ldots, \mathcal{P}_n$ are located in a ring topology where each process is connected by an unidirectional, asynchronous channel to its neighbour as outlined below.



To distinguish the processes, each process i is assigned a unique identifier $id(\mathcal{P}_i) \in \{1, \ldots, n\}$ that is written to private variable id_i. The aim of the algorithm is to elect the process with the highest identifier as the (unique) leader within the ring. Therefore each process executes the following algorithm using another private variable m_i (which is initially 0):

```
send(id_i); // send own id to next process.
while (true) do {
  receive (m_i);
  if (m_i == id_i) then stop; // process i is the leader
  if (m_i > id_i) then send(m_i); // forward other identifier
}
```

a) Model the leader election protocol for n processes as a channel system.

b) Give an initial execution fragment of $TS([\mathcal{P}_1 \mid P_2 \mid \mathcal{P}_3])$ such that at least one process has executed its send-statement within the body of the while-loop. Assume for $1 \leq i \leq 3$, that process P_i has identifier $id_i = i$.

Exercise 2 (Extending Channel Systems): (2 points)

According to the lecture, the transition relation $\hookrightarrow$ of a program graph $\mathcal{P} = (\text{Loc}, \text{Act}, \text{Effect}, \hookrightarrow, \text{Loc}_0, g_0)$ in a channel system over the set of variables Var and set of channels Chan consists of transitions of the form

- $\ell \overset{g:\alpha}{\hookrightarrow} \ell'$ with $\ell, \ell' \in \text{Loc}$, $g \in \text{Cond}(\text{Var})$, $\alpha \in Act$, or
- $\ell \overset{c!v}{\hookrightarrow} \ell'$ with $\ell, \ell' \in \text{Loc}$, $c \in \text{Chan}$, $v \in \text{Dom}(c)$, or
- $\ell \overset{c?x}{\hookrightarrow} \ell'$ with $\ell, \ell' \in \text{Loc}$, $c \in \text{Chan}$, $x \in \text{Var}$.

For both subtasks, consider a channel system consisting of the program graphs $\mathcal{P}_1, \ldots, \mathcal{P}_n$ that are extended or modified as described in the subtasks.

a) Let $env \notin \text{Chan}$ be a distinguished channel. Reading from this channel is supposed to model reading a value from an (unknown) environment. Give SOS rules for the transition system semantics for transitions of the form $\ell \overset{env?x}{\hookrightarrow} \ell'$.

b) For channels c with $cap(c) = 0$, we change the semantics of a transition $\ell \overset{c!v}{\hookrightarrow} \ell'$ to that of a broadcasting mechanism: if a value is sent via such a channel, any number of processes (including none) can decide to participate in the handshaking and receive the value. Give SOS rules that formalize the transition system semantics of this operation.

Exercise 3 (Parallel Composition): (3 points)

In the following, whenever transition systems are compared via $=$ or $\neq$, this means (in)equality **up to renaming of states** (i.e. isomorphism).

a) Show that, the handshaking $\|_H$ operator **is not** associative, i.e. that in general

$$(\mathrm{TS}_1 \|_H TS_2) \|_{H'} \mathrm{TS}_3 \neq \mathrm{TS}_1 \|_H (\mathrm{TS}_2 \|_{H'} \mathrm{TS}_3)$$

b) The handshaking operator $\|$ that forces transition systems to synchronize over their common actions **is** associative. Show that

$$\underbrace{(\mathrm{TS}_1 \| \mathrm{TS}_2) \| \mathrm{TS}_3}_{L} = \underbrace{\mathrm{TS}_1 \| (\mathrm{TS}_2 \| \mathrm{TS}_3)}_{R}$$

where $\mathrm{TS}_1, \mathrm{TS}_2, \mathrm{TS}_3$ are arbitrary (finite) transition systems. To this end, show that the bijective function $f_\approx : (S_1 \times S_2) \times S_3 \to S_1 \times (S_2 \times S_3)$ given by $f_\approx(\langle\langle s_1, s_2\rangle, s_3\rangle) = \langle s_1, \langle s_2, s_3\rangle\rangle$ preserves the transition relation in the sense that

$$l \xrightarrow{\alpha}_L l' \iff f_\approx(l) \xrightarrow{\alpha}_R f_\approx(l') \tag{1}$$

where $l, l' \in S_L$, S_L is the state space of transition system L and $\longrightarrow_L$, $\longrightarrow_R$ are the transition relations of L and R, respectively.

Hint: When considering an action α, you need only distinguish the cases

(i) $\alpha \in \mathrm{Act}_1 \setminus (Act_2 \cup Act_3)$

(ii) $\alpha \in (\mathrm{Act}_1 \cap \mathrm{Act}_2) \setminus Act_3$

(iii) $\alpha \in \mathrm{Act}_1 \cap Act_2 \cap Act_3$

(Act_i is the action set of TS_i) as all other cases are symmetric. Also, for simplicity, it suffices to show the direction "$\Longrightarrow$" of condition (1). However, keep in mind that L and R are not necessarily action-deterministic (see exercise sheet 1).

Exercise 4 (LT properties): (2 points)

Let's do some rocket science. Imagine a rocket whose possible (observable) behaviors are infinite traces over the atomic propositions

$$\mathrm{AP} = \{parachute, ground, space\}$$

All signals are issued by two detectors on board the rocket: a parachute detector and a ground/space detector. Assume that the detectors are working perfectly unless otherwise specified, meaning that *ground* and *space* are never detected in a contradictory way. Note that a rocket can only go from the ground to space and back (and repeat this). It is not possible to go from space into the atmosphere and back to space again without coming back to the ground. I mean c'mon, seriously, what kind of rocket could do that?!
Express the following informally stated properties as LT-properties over the given atomic propositions AP and indicate which of them are invariants. Give (very) brief explanations for your answers.

(i) Initially, the rocket is on the ground.

(ii) Assuming that the detector actually *can* break:
The ground/space detector is never broken, meaning that it never yields contradictory results.

(iii) Whenever the rocket leaves the ground, it eventually returns to the ground.

(iv) Whenever the rocket descends back to earth, the parachute is open at most two time steps after space has been left.

(v) The rocket is only finitely many time steps in space.