Day 1 – Commit Protocols

- Write down the decentralized 2PC protocol in full detail.
- Assume that the maximum delay for a message to go from a process to its neighbours is the same for all process.
  - Write down the linear 2PC protocol in detail, indicating the timeout period that each process should set when waiting for a message.
  - Explain carefully how timeouts are determined by the position in the chain and the kind of expected message.
- In the linear 2PC protocol it is possible to speed up the abort when some process votes NO (passing the abort message immediately to the left).
  - Develop this variation of 2PC.
Day 2 – Petri Nets

- Consider the net below:

![Petri Net Diagram]

- find two different processes that correspond to the same computation proof (in the algebra that we have seen)
- write down part of the unfolding showing three pairs of events that are related respectively by ≤, # and co

Day 2 – Zero Safe Nets

- Consider the Zero Safe net:

![Zero Safe Net Diagram]

- write the corresponding CTPh / ITPh abstract nets
- Write the Zero Safe nets corresponding to processes
  - a!(a?0|a!0)
  - (a?0|a!0|a!b?0|b!0)\a\b
  - (a?0+b?0)|(b!0+c!0)\a\b\c
- and then find the associated abstract nets
Day 2 – Basic Shapes

Show that given any Zero Safe net $B$, we can always find a corresponding Zero Safe net $B_{\text{base}}$ that:

- yields the same abstract net as $B$,
- contains only transitions of the form below:

```
  open
  fork
  join
  close
  drop
```

Day 3 – Linda

Draw the LTS associated with the process

\[
\text{out}(a).0 \mid \text{rdp}(a)?\text{out}(b).0:\text{out}(c).0 \mid \text{in}(a)?\text{out}(c).0:\text{out}(b).0
\]

under both the ordered and unordered semantics

then draw

- the corresponding Petri nets
- their unfoldings
Day 3 – Net Flavors

- It has been proved that reachability is decidable for any finite contextual net $N$ that has exactly one inhibitor arc.
  - Exploiting this result, show that reachability is decidable for any finite zero safe net.

(Day 3 – TraLinda)

- Write a TraLinda process $P$ for modeling the meeting schedule problem:
  - the goal is to select a date for a meeting which is ok for every member of the team;
  - the decision is constrained by a finite set of possible dates based on the availability of the meeting room;
  - every member of the team suggests possible dates until an agreement is reached.
  - State any additional assumptions you may need.
  - Illustrate $zsnet(P)$ and the corresponding abstract net.
Day 4 – Join Calculus

- Define the type system $\Delta_2$ that characterizes those terms of the join-calculus that correspond to reconfigurable nets.
- Find the time and message complexity of D2PC.

Day 5 – Join Calculus

- Define the Zero Safe net and the corresponding join-calculus implementation for the instance of the “Apartment Problem” defined by
  - Persons: P, Q, R, S, T
  - Apartments: A (free), B, C, D (free), E
  - Rented: P in B, Q in C, S in E
  - Wanted: P wants A or C, Q wants A or B, R wants A or C or E, S wants C or D, T wants D or E
- What is the abstract net?
Day 5 – Compensations

- Propose an extension of join calculus (or committed join) where each transaction has two kinds of compensations:
  - the first to be activated on abort
  - the second to be activated on commit
- Describe the syntax, heating/cooling rules and reactions.
- Illustrate the main features of the new calculus by means of a few examples.

Day 5 – Committed Join

- Write a Committed Join process P for modeling the order fulfillment problem:
  - A company takes orders from its customers;
  - when a new order is received the company:
    - asks the warehouse to prepare the order for shipment
    - books a courier to collect the order
    - performs a credit check on the customer to verify that the goods can be paid
  - State any additional assumptions you may need.