Dr. Annette Bieniusa
Albert Schimpf, M. Sc.

TU Kaiserslautern
Fachbereich Informatik
AG Programmiersprachen

Exercise 7: Programming Distributed Systems (Summer 2020)

Submission

- You need a team and a Gitlab repository for this exercise sheet.
- In your Git repository, create a branch for this exercise sheet, for example with

  ```
  git checkout -b ex7
  ```

- Create a folder named "ex7" in your repository and add your solutions to this folder.
- Create a merge request in Gitlab and assign Albert Schimpf as assignee. If you do not want to get feedback on your solution, you can merge it by yourself.

1 TLA+ specifications: AB Protocol

Install the TLA+ toolbox and specify the Alternating Bit Protocol (AB Protocol) as presented by Leslie Lamport in his video course in the linked videos on part 6.3. of this learning paths.

- Specification of the AB Protocol `http://lamport.azurewebsites.net/video/video9a.html` (description starts at "What the protocol should accomplish" at time 05:41)
- Safety and Liveness Properties of the AB Protocol `http://lamport.azurewebsites.net/video/video9b.html`

Hint for Part 2: The Protocol (17:28) The Advanced Options page has been eliminated. The State Constraint is now specified on the Spec Options page, which is reached by clicking Additional Spec Options on the model's Model Overview page.

2 Temporal properties

Consider the following property for the boradcast protocols:

> Every message should be delivered exactly once.

- Explain why this is neither a liveness nor a safety property!
- Give a definition of this property as conjunction of some liveness and some safety property!

3 TLA+ specifications: Broadcast Protocols

On the website you find a TLA+ specification for the Best-effort Broadcast algorithm.

- Explain in your own words how the specification works.
  - What are the state variables and what do they represent?
  - What are the state transitions?
  - How is the network layer modeled?
- Identify the following liveness properties and decide which one should hold for the best-effort broadcast!

```
Property1 ==
     [](\A i \in Process, j \in Process:
          i \in correctProcess /\ j \in correctProcess =>
             \A msg \in Message: msg \in broadcast[i]
                => <>(msg \in delivered[j]))


 Property2 ==
     [](\A msg \in Message: \A i \in Process: i \in correctProcess =>
         (msg \in delivered[i]
              => <>(\A j \in Process : j \in correctProcess =>
                        msg \in delivered[j])))

 Property3 ==
     [](\A msg \in Message: \A i \in Process:
         (msg \in delivered[i]
              => <>(\A j \in Process : j \in correctProcess =>
                        msg \in delivered[j])))

 Property4 ==
     [](\A m \in Message: \A p \in Process: \A m2 \in Message:
         m \in delivered[p] /\ m2 \in happensBefore[m] => m2 \in delivered[p])
```

- Use TLC to model check the properties!
  - Why is it useful to consider a model with 3 processes?
  - Why is it useful to consider a model with 3 messages?