

Some of the answers to these questions can be found using the lecture slides, the recommended textbooks or other sources (question marked with \*). Some questions may have more than one possible answer, or be more or less open for discussion. Note that no answers (solutions) will be given to these questions, but if help is needed the assistants will be available to answer questions. The concepts marked with yellow are important and should be fully understood.

## Inter-process Communication (IPC)

1. What is IPC and why do we need it?
2. Fundamental IPC models
  - (a) Discuss the two fundamental IPC models presented in the lecture. What are the differences? Where is the kernel involved?
  - (b) Give a concrete application scenario for each model. Discuss why your scenarios fit to the respective model (and not the other one).
3. Message passing.
  - (a) Discuss the different synchronization options for the send/receive primitives. Where do you see pros and cons?
  - (b) In the lecture, three different buffering approaches were presented. Describe the approaches and discuss how the buffering approach interplays with the synchronization options.
  - (c) Discuss the pros and cons of variable vs. fixed size messages.
4. Client-Server Communication
  - (a) Client and server processes can reside on the same machine or on different machines that are connected by a network. Discuss the differences between both scenarios. Which of the fundamental IPC models is suitable for each scenario and why?
  - (b) What is a socket? Describe the concept and state which of the fundamental IPC models sockets implement. Do you know different kinds of sockets?
  - (c) Describe the RPC concept and state which of the fundamental IPC models it uses. Which synchronization approach is typically used?
  - (d) Explain the *at most/least once* RPC calling semantics. Describe the undesirable consequences that could arise from not enforcing them. Describe possible uses for a mechanism that has neither of these guarantees.
- 5.\* What implies “distribution transparency” (see Andrew S. Tanenbaum and Maarten van Steen, *Distributed Systems – Principles and Paradigms*)?

## Hierarchical Microkernel

6. Describe the hierarchical microkernel architecture. What is the difference to a common microkernel architecture?
7. How does inter-module communication work?
  - (a) Describe the inter-module communication approach. Use an example system with at least *three* modules and *two* hierarchy levels, excluding the kernel at level 0.
  - (b) Is there a difference between inter-module and inter-process communication? Consider module communication across multiple hierarchy levels.
  - (c) Which of the fundamental IPC models is used in the hierarchical microkernel architecture?