

# Seminar questions

## Module 2

### The process concept and inter process communication

File descriptors, standard streams, sockets, TCP vs UDP, and I/O redirection

Computer systems with project work (1DT003) Uppsala university spring 2020 - document updated 2020-01-29

## The process concept

1. What is meant by a process?
2. Name and describe the various memory segments used by a process.
3. Draw a diagram showing how the process states are related using directed arrows showing possible state transitions

## Communicating processes

4. What is the main difference between a) process-oriented and b) service-oriented communication?
5. What is the main difference between the design of an a) client/server and b) peer-to-peer, system?
6. What is *jitter* and how is it relevant for processes that are communicating with each other?
7. What information is used for process multiplexing between communicating processes at different hosts?

## The process control block (PCB)

8. What is the purpose of the PCB?
9. Give examples of data stored in the PCB.
10. Explain how PCBs can be used to construct various process queues.

## Fork

11. What is the purpose of the fork system call?
12. What do we mean with parent and child?
13. What are the possible return values of fork and how are they interpreted?

## Exit

14. What is the purpose of the exit system call?

## Exec

15. What is the purpose of the exec family of system calls?
16. When calling a function or invoking a system call, normally execution will return back to the caller, possibly with a return value. Is this true for the exec family of system calls? Justify your answer.

## Wait

17. What is the purpose of the wait system call?

## Zombie processes

18. What is the purpose of the zombie process state? When does a process become a zombie?

## Signals

19. What is the purpose of signals?
20. What happens when a process receives a signal?

## File descriptors and Sockets

21. Explain the file descriptor concept.
22. How are file descriptors and sockets related?
23. If you write a TCP server that forks a new child for every connecting client, what is important to remember about how the socket descriptors are handled?

## Pipes

24. What is a pipe?
25. How are file descriptors used together with pipes?
26. How do we create a pipe? What is the result of creating a pipe?
27. What happens if we read from an empty pipe and there are a) open write descriptors attached to the pipe, or b) no open write descriptors attached to the pipe?
28. What happens if we write to a) a full pipe if there are open read descriptors attached to the pipe, or b) a pipe with no open read descriptors attached to the pipe?

## TCP and UDP

29. How does TCP and UDP differ with respect to a) speed, b) reliability, c) adaptiveness, d) communication service?
30. Outline how TCP congestion control typically works in an AIMD manner.
31. Give examples of applications that would prefer the communication service provided by a) TCP, and b) UDP. Justify your answer.

## The dup2 system call

32. What is the dup2 system call doing to file descriptors and how can this be useful?