Seminar questions

Module 8
Erlang

Operating systems and process oriented programming (1DT096)

Uppsala university spring 2020 - document updated 2019-12-16

Functional programming

1. In short, how is the functional programming paradigm different from the imperative programming paradigm?
2. What is meant by the head and tail of a list?
3. What is meant by a predicate?
4. What is meant by arity?
5. What is meant by a higher order function?
6. What is meant by an anonymous function?
7. How does anonymous functions relate to higher order functions?
8. Four common operations on lists are: map, filter, zip, and fold. In brief, describe these operations.

Recursion

9. In computer science, what is meant by recursion?
10. Why is recursion important in functional programming languages?
11. What is meant by tail recursion?
12. How does tail call optimisation work?
13. Why is tail recursion important?

Message passing

14. What is meant by synchronous message passing? What is meant by asynchronous message passing?

The actor model

15. Describe the actor model for concurrency?
16. How is the actor model different from processes and threads?

Erlang

The questions about Erlang has been divided into separate sections.

Lightweight processes

17. What makes an Erlang process lightweight compared to threads and processes in operating systems?

18. In Erlang, how can you create a new process?

19. Do Erlang processes share any memory?

20. How can a process get to know its own process id (PID)?

Message passing

21. Explain the syntax for sending a message from one process to another process.

22. Explain the syntax for receiving a message.

23. What can be sent in a message?

24. Is sending a message blocking the sender?

25. Is receiving a message blocking the receiver?

26. Is message passing in Erlang synchronous or asynchronous?

27. If process A sends a message to process B process and wants that process to send some sort of result back, how can this be accomplished?

Stateful process

28. How is it possible for process to maintain and change state?

Process supervision

29. What is the effect of linking two processes?

30. What is the purpose of trap exit?

Hot code swapping

31. Explain what is meant with hot code swapping (aka hot swapping or code replacement).

32. In brief, explain how hot code swapping works in Erlang.