Operating systems 2018

Operating systems (1DT044)
Operating systems and process oriented programming (1DT096)
Uppsala university

Seminar questions

Module 1

Fundamental concepts

Document last updated
2018-01-16

The grand picture

1. What is the overall purpose of an operating system?

2. On a single (core) CPU, how can the operating system make several programs execute seemingly at the same time?

3. What do we mean with operating system kernel?

Processes

4. What is the difference between a program, executable and process?

5. What is the purpose of the call stack?

6. Can processes share a single stack? Justify your answer.

Dual mode operation

7. What are the names of the two modes?

8. What is the purpose of dual mode operation?

9. How does dual mode achieve its purpose?

Exceptions and interrupts

10. In general, what does it mean for something to be synchronous or asynchronous?

11. In relation to the CPU, what does it mean for an event to be synchronous or asynchronous?

12. What is the purpose of exceptions and interrupts?

13. What are the differences between an exception and an interrupt?

14. What is meant by CPU context?

15. What steps are taken when handling an exception or interrupt?

System calls

16. What is the overall purpose of the system call concept?

17. When requesting service from the operating system, why can a user process simply not use an ordinary function call to the kernel? Why must this be done using a system call and how is this different from an ordinary function call?

18. Why are exceptions important when implementing system calls?

Multiprogramming

19. What is meant by I/O?

20. What is special with I/O compared to normal execution?

21. In multiprogramming a job can be in three states, name and explain the purpose of each state.

22. In multiprogramming, what happens when a job makes a request for I/O.

System call design

23. Explain step-by-step how the `getc` system call that allows a program to read a single character typed by a human user on the keyboard can be implemented.

24. Compared to `getc`, what steps must be added to implement the `gets` system call that allows a program to read a string of characters typed by a human user on the keyboard.