SOFTENG 370 Tutorial 4

Timo van Veenendaal

25 August 2020

Today

- Test is next week :)
- This tutorial: going over some past test questions
- Includes Kahoot!
- Slides will be uploaded after tutorial

Kahoot!

[[Show the Kahoot now]]
Thank you to Edward Zhang, last year's 370 tutor, for the Kahoot!

Test 2019 Question 1b

Question

Choose one significant change in the history of operating systems. Explain in approximately three sentences what the change was and why you consider it significant. The change can be either hardware or software.

Test 2019 Question 1b

Question

Choose one significant change in the history of operating systems. Explain in approximately three sentences what the change was and why you consider it significant. The change can be either hardware or software.

- A pretty open-ended question, there's lots you could talk about
- Some ideas: faster CPUs, more memory, addition of kernel/user mode, protected memory
- Plus something reasonable about why that is important

Test 2019 Question 1c

Question

Give a reason why a language such as Python would not be used to implement an operating system.

Test 2019 Question 1c

Question

Give a reason why a language such as Python would not be used to implement an operating system.

- Again, quite open-ended.
- It doesn't give direct access to memory addresses.
- It doesn't map well to machine code (Python is object-oriented).
- It has a large runtime.
- It is slow compared to C.

Test 2018 Question 3a

Question

What problem does the "trap and emulate" approach solve with respect to virtual machines and how does it solve this problem?

Test 2018 Question 3a

Question

What problem does the "trap and emulate" approach solve with respect to virtual machines and how does it solve this problem?

- Deals with the problem of guest operating systems not being able to execute privileged instructions
- VMM catches the exception, determines which VM is responsible, and emulates or runs the privileged instruction before returning to the guest

Test 2019 Question 2b

Question

Virtual memory has traditionally been difficult to implement on virtual machines. This problem has been solved either with shadow page tables or more recently with second level address translation. Explain why virtual memory is a problem in virtual environments.

Test 2019 Question 2b

Question

Virtual memory has traditionally been difficult to implement on virtual machines. This problem has been solved either with shadow page tables or more recently with second level address translation. Explain why virtual memory is a problem in virtual environments.

- Each OS expects full control over the physical memory, including low-level changes as to which pages are where.
- So the physical page tables need to be under the control of the VMM
- Either the page tables the guest uses are not the real ones (shadow page tables)
- Or an extra level of translation needs to be supported by the hardware

Test 2018 Question 4a

Question

Explain the difference between a thread and a process?

Test 2018 Question 4a

Question

Explain the difference between a thread and a process?

- A process has two parts: the resources allocated to it (memory, files, devices...) and the running part which is one or more threads
- A thread is a sequence of instructions running inside a process
- Switching between threads in the same process is generally simpler than between processes since most resources are shared and don't need to be changed

Test 2018 Question 4b/c (adapted)

Question

What is the most important difference between system level and user level threads?
What is a consequence of that difference?

Test 2018 Question 4b/c (adapted)

Question

What is the most important difference between system level and user level threads?
What is a consequence of that difference?

- The OS knows about system level threads and can therefore schedule them individually

Test 2018 Question 4b/c (adapted)

Question

What is the most important difference between system level and user level threads?
What is a consequence of that difference?

- The OS knows about system level threads and can therefore schedule them individually
- Consequence: multiple user level threads in the same process cannot run on different CPUs simultaneously. Also, if one user level thread is blocked, then the other user level threads cannot run since as far as the OS is concerned, that single thread is blocked (unless the user-level thread implementation accounts for this (how?))

That's it

- Any questions?
- See you all next week
- Next week's tutorial: more test questions!
- Good luck with your study