11/10/15 Pre Class Work

* Required

**Harvard email address:** *

Please indicate which course you are taking. *

- CS61 (College)
- CSCIE-61 (Extension)

**Semaphores using Mutexes and CVs**

If you were given semaphores, could you build mutexes on top of them? *

- Yes, the mutex would simply be a binary semaphore
- Yes, the mutex would simply be a counting semaphore
- No, semaphores do not provide sufficient functionality

What guarantee does a semaphore provide about the order in which incoming threads acquire the semaphore? *

- Threads acquire the semaphore in FIFO order.
- Threads acquire the semaphore in LIFO order (the last thread to block gets the semaphore)
- The thread that has waited the most times gets the semaphore.
- There is no guarantee provided.

If you had only mutexes, how would you build a semaphore implementation? *

The mutex would protect the counter. However, without a CV, you do not have atomic check & sleep, so you need some form of busy waiting.

Solving Pingpong using Semaphores

The pingpong solution uses: *

- Counting semaphores
Thinking back to A2 ...

Please describe your approach to defusing the bomb. *
Be honest! We are thinking about how the assignment can be made more useful for future years.

Which of the following best matches your response to the previous question? *
Please answer the previous question first. This question will make our preliminary analysis easier, but we still want to read your description!

I reverse-engineered the C source of each phase
I tried random inputs until something worked
I read through the assembly and figured out what each phase does, but I did not translate into C
I stepped through execution with gdb to figure out what each phase does
I tried different inputs and looked at what was compared to the solution at the end of each phase to determine how each phase changes its input
Other: 

Did you work with a partner on A2? *
Yes
No

Practicum

Which command in GDB shows you the set of callframes that are currently active? *
Up
Down
Where
Print

Have you ever used a conditional breakpoint? *
(If not, quick go find out how -- they are quite useful!)
Yes
Food for thought

These are questions designed to make you think.

Let’s say that you have a highly threaded program and you want to run tests on it to see if you can trigger any race conditions. *
Would you expect to be more likely to trigger race conditions on a really fast processor or a really slow one?

- Fast
- Slow

Discuss

Submit

Never submit passwords through Google Forms.