Thread Throttling

- *admission control* is a gatekeeping mechanism to protect resources from overuse by denying access to some potential users
- *thread throttling* limits the number of executing threads to prevent system overload or other performance impacts
  - e.g. limit concurrent I/O operations
  - e.g. limit the number of simultaneous server requests handled
  - e.g. limit the number of threads executing a memory-intensive section of code
- use a semaphore
  - initialize semaphore's value to the maximum number of threads allowed in the region at once
  - put `sem_wait()` and `sem_post()` around the region

Summary

- we've seen three synchronization primitives
  - *locks* provide mutual exclusion
  - *condition variables* provide an alternative to busy waiting until a condition is satisfied
  - *semaphores* are more general purpose
- it's worthwhile to be familiar with all three
  - each provides a convenient solution for certain kinds of problems
  - concurrency bugs are common enough – you don't need unnecessary complexity on top of that!