Univ. Grenoble Alpes (Ensimag & UFR IM2AG)
M1 MoSIG
Academic year: 2025-2026

Principles of Operating Systems — Midterm #1 — Some answers

Problem 1

1.1) a, b, d
1.2) a, c
1.3) a, b, c
1.4) a, b, c
1.5) b

Problem 2

2.1

Running: N
Ready: More than N
Blocked: More than N

2.2

Note: To simplify the discussion, in the answers below, we assume that the threads are managed by the kernel scheduler.

(a) Transition from *Ready* to *Running*:
A decision of the scheduler, itself invoked (while another thread/process was running) due to one of the following events: (1) the occurrence of a hardware interrupt of (2) the invocation of a system call by the thread currently running at that time.

(b) Transition from *Running* to *Ready*:
Preemption by the scheduler caused by (1) the occurrence of a hardware interrupt (typically from the periodic timer or possibly from another device such as a disk) or (2) the invocation of a system call (whose code may lead to the invocation of the scheduler) by the running thread.

(c) Transition from *Running* to *Blocked*:
The invocation of a blocking system call by the running thread, for example, in order to perform a disk I/O operation, a network I/O operation, or a synchronization operation (e.g., requesting a mutex lock that is not currently available).

(d) Transition from *Blocked* to *Ready*:
The occurrence of the logical event that the blocked thread was awaiting. For example, the completion of a disk/network I/O operation, or the obtention of a mutex lock.

2.3

(a) Yes (but only under certain conditions discussed in the next answer below).

(b) At a given point in time during the lifespan of P_i, it is possible to have all its threads simultaneously in the *"Running"* only if **all** of the following conditions hold:

- The total number of threads of P_i is not greater than the total number of CPU cores on the machine. In other words: $X \leq N$
- The threads used by P_i are implemented as kernel-level (as opposed to user-level) **and** preemptive (as opposed to cooperative) threads.

Problem 3

3.1

(a) Potential drawback of the new design: increased (and potentially significant) internal fragmentation. For example, a requested block size of 600 bytes will result in the allocation of more than 400 extra bytes (used only for padding).

(b) Potential advantages of the new design: The set of possible sizes for an allocated block will be reduced. As a consequence, this may reduce the time needed to allocate a block (if the chosen free block is larger than the requested size, splitting it may not be systematically required). Another consequence of these more homogeneous block sizes is that this may help reducing (although not fully eliminating) external fragmentation.