P1(a) 15 pts

Significant speed difference between main memory and disk makes it desirable to context-switch out the page faulting process which blocks on disk I/O to complete and bring in the missing page. Hardware busy waiting on disk I/O to complete would waste CPU resources.

5 pts

If there is a free frame, perform disk read to bring in the missing frame. In tandem, context-switch out the page faulting process and context-switch in a ready process. If there is no free frame, evict a page followed by the steps above.

4 pts

The page that will be accessed by the process farthest in the future.

2 pts

LRU

2 pts

Global clock is simpler and less costly to implement with hardware support than LRU.

2 pts

P1(b) 15 pts

Most files are small, a minority are very large ("mice and elephants" or 90/10 rule).

4 pts

Direct points (e.g., 10) allow small files to be indexed with constant overhead. Indirect pointer (single, double, triple) allow very large files to be indexed with logarithmic overhead.

3 pts

Typical: constant overhead since most files are small.

2 pts

Worst-case: logarithmic overhead since a minority are very large.

2 pts

For application environments such as memory sticks and other portable memory devices, the typical operation is copying entire files from one memory device to another as such. Therefore constant/logarithmic random access overhead facilitated by indexing in traditional file systems may not be practically relevant.

4 pts

P1(c) 15 pts

One, linear overhead.

5 pts

Two, by isolation/protection the damage from processes belonging to an app deadlocking is limited to the application without significant impact on the rest of the system.

5 pts

The primitives wait() and signal() themselves utilized hardware support (i.e., interrupt disabling) to carry out their operations without disruption/atomically.

5 pts

P2(a) 15 pts

Most processes consume little memory whereas a minority consume a lot
Since for most processes only a few entries in their page table are used this affords an opportunity to compress.

5 pts

12 bits represent offset within a page, hence are not needed in address translation. The last (most significant) 10 bits code for the level-2 (outer) page table containing 1024 (= 2^10) entries. Each of the 1024 entries may point to a level-1 (inner) page table each containing 1024 entries. Each entry in the level-1 page table contains 20 bits that specify the 20-bit physical address that the 10+10 bits virtual address is translated to. [Some of the 12 offset bits are used for bookkeeping purpose such as noting whether the page resides in main memory.]

7 pts

Since typical processes consume little memory, most entries in the level-2 page table do not point to level-1 page tables. The level-2 page tables has 1024 entries. Since only a few level-1 page tables are needed (each containing 1024 entries), the total memory (in terms of page table entries) consumed by level-2 and level-1 page tables is significantly smaller than a million.

3 pts

P2(b) 15 pts

When events are sparse, responding to system timer ticks every 1 msec consumes battery power which is a scarce resource for mobile devices.

5 pts

If events are dense then a tickless kernel would generate too frequently which can bog down the system. [In a tickful kernel system timers are generated at a fixed rate with dense events handled as a batch.]

4 pts

Suppose the first (i.e., earliest) event in the sleep queue implemented as a delta list has time value x msec. Program the interval timer to generate a system timer interrupt after x msec. When a system timer interrupt occurs, dequeue the first element from the sleep queue and program the interval timer to generate a clock interrupt after y msec where y is the time value of the second (now the first) event in the delta list.

6 pts

P3 25 pts

The lower half responds to interrupts by borrowing the context of the current process which copies the content from device buffer to the shared kernel buffer. Hence the current process which may not be the recipient of a message is the writer/producer of the shared kernel buffer.

4 pts

The upper half executes when a system call is made by the recipient process. The upper half copies the content of the shared kernel buffer to the user buffer. Hence the recipient process is the reader/consumer of the shared kernel buffer.

4 pts

In both cases, a process writes to the shared buffer and a process reads from the shared buffer which falls under IPC producer/consumer queue coordination.

3 pts

The lower half cannot be allowed to block since the process whose context is being borrowed may be the idle/null process which must always be ready to execute when there are no ready processes in the system.

3 pts

The top half of the lower half runs with interrupts disabled to perform lower overhead tasks. The bottom half of the lower half runs with interrupts enabled since it performs interrupt handling tasks that may incur more overhead. Running with interrupts enabled allows new interrupts
to preempt the bottom half so that interrupts are not missed.
4 pts

In the absence of DMA, the top half copies data from the device buffer
to kernel memory (in addition to signaling the device controller).
2 pts

With DMA support, the top half instructs the DMA to copy how much data
where in main memory.
2 pts

The bottom half may be implemented as a kernel process which allows it
to make blocking function calls.
3 pts

Bonus 10 pts

Memory thrashing is the phenomenon where memory demand significantly
exceeds physical memory so that locality of reference is unable to
prevent pages needed in the near future from being evicted.
3 pts

Chief symptoms are high page fault rate and low CPU utilization.
3 pts

Increasing CPU speed does not alleviate the problem since disk I/O
is the bottleneck where CPU remains mostly idle.
2 pts

For page fault rate to keep increasing a process has to page fault.
To page fault, a process has to execute. However, a page faulting
process blocks on disk I/O to complete. Hence a page faulting
process eventually self limits how high page fault rate can increase.
2 pts