IA Operating Systems: Scheduling Algorithms, Memory Management

 $\label{eq:Lent 2024} Lent 2024$ Last Updated: Tuesday 18th June, 2024 22:24

- 1 (a) Describe how the CPU is allocated to processes if static priority scheduling is used. Be sure to consider the various possibilities available in the case of a tie.
 - (b) "All scheduling algorithms are essentially priority scheduling algorithms."

Discuss this statement with reference to the first-come first-served (FCFS), shortest job first (SJF), shortest remaining time first (SRTF) and round-robin (RR) scheduling algorithms.

- (c) What is the major problem with static priority scheduling and how may it be addressed?
- (d) Why do many CPU scheduling algorithms try to favour I/O intensive jobs?
- (e) What information do the shortest job first (SJF) and shortest remaining time first (SRTF) algorithms require about each job or process? How can this information be obtained?

- **2** An operating system uses a single queue round-robin scheduling algorithm for all processes. You are told that a *quantum* of three time units is used.
 - (a) What can you infer about the scheduling algorithm?
 - (b) Why is this sort of algorithm suitable for a multi-user operating system?
 - (c) The following processes are to be scheduled by the operating system.

Process	Creation Time	Required Computing Time
P_1	0	9
P_2	1	4
P_3	7	2

None of the processes ever blocks. New processes are added to the tail of the queue and do not disrupt the currently running process. Assuming context switches are instantaneous, determine the *response time* for each process.

(d) Give one advantage and one disadvantage of using a small quantum.

3 For each of the following, indicate if the statement is true or false, and explain why:

- (a) A paged virtual memory is smaller than a segmented one.
- (b) Shortest job first (SJF) is an optimal scheduling algorithm.
- (c) Round-robin scheduling can suffer from the so-called 'convoy effect'.
- 4 (a) What is the *address binding* problem?
 - (b) The address binding problem can be solved at compile time, load time or run time. For *each* case, explain what form the solution takes, and give one advantage and one disadvantage.

- **5** Most operating systems provide each process with its own *address space* by providing a level of indirection between virtual and physical addresses.
 - (a) Give three benefits of this approach.
 - (b) Are there any drawbacks? Justify your answer.
- 6 A processor may support a *paged* or a *segmented* virtual address space.
 - (a) Sketch the format of a virtual address in each of these cases, and explain using a diagram how this address is translated to a physical one.
 - (b) In which case is physical memory allocation easier? Justify your answer.
 - (c) Give two benefits of the segmented approach.
- 7 (a) In the context of memory management, under which circumstances do *external* and *internal* fragmentation occur? How can each be handled?
 - (b) What is the purpose of a page table? What sort of information might it contain? How does it interact with a TLB?
 - (c) Describe with the aid of a diagram a two-level page table. Explain the motivation behind the structure and how it operates.
- 8 Describe *how* CPU scheduling algorithms favour I/O intensive jobs in the Unix operating systems.

- 9 Past paper questions
 - y2015p2q4 [not the last part of (c) about segment faults]
 - y2013p2q4
 - y2009p2q3 [not (b)]
 - y2009p2q4
 - y2011p2q4 (a)