Tutorial 8
Virtual Memory part II (Chapter 10) and Process Scheduling (Chapter 5)
Operating Systems CS 3SH3 Term 2, Winter 2022
Prof. Neerja Mhaskar

Tutorials are not mandatory. They are simply a tool for you to understand the course concepts better.

Tutorial Format: The questions will be posted a day before or on the day of the tutorial on the course website. You can choose to solve these problems before hand and come in with your solutions. I or one of the TAs helping me will check your solutions. If you have all of the questions correct you can choose to leave. If you have any of them incorrect, it is recommended that you stay and understand the solutions.

Solutions to the tutorial will not be posted online.

1. Consider the following page reference string:
   7, 2, 3, 1, 2, 5, 3, 4, 6, 7, 7, 1, 0, 5, 4, 6, 2, 3, 0, 1.

   Assuming demand paging with three frames, how many page faults would occur for the following replacement algorithms?
   a. LRU replacement
   c. Optimal replacement

2. The VAX/VMS system uses a FIFO replacement algorithm for resident pages and a free-frame pool of recently used pages. Assume that the free-frame pool is managed using the least recently used replacement policy. Answer the following questions:

   a) If a page fault occurs and if the page does not exist in the free-frame pool, how is free space generated for the newly requested page?
   b) If a page fault occurs and if the page exists in the free-frame pool how is the resident page set and the free-frame pool managed to make space for the requested page?
   c) What does the system degenerate to if the number of resident pages is set to one?
   d) What does the system degenerate to if the number of pages in the free-frame pool is zero?

3. What is the cause of thrashing? How does the system detect thrashing? Once it detects thrashing, what can the system do to eliminate this problem?

4. Consider the parameter Δ used to define the working-set window in the working-set model. What is the effect of setting Δ to a small value on the page fault frequency and the number of active (non-suspended) processes currently executing in the system? What is the effect when Δ is set to a very high value?

5. Assume there is an initial 16 KB segment where memory is allocated using the Buddy system. As shown in lecture notes on the Buddy system, draw the tree illustrating how the following memory requests are allocated:
1. request 3.6 KB
2. request 1.5 KB
3. request 1.2 KB
4. request 1.9 KB
5. request 2.7 KB

Next, modify the tree for the following releases of memory. Perform coalescing whenever possible and list the available segment sizes:
1. release 1.2 KB
2. release 1.9 KB
3. release 1.5 KB

6. Suppose that the following processes arrive for execution at the times indicated. Each process will run for the amount of time listed. In answering the questions, use non-preemptive scheduling, and base all decisions on the information you have at the time the decision must be made.

Process	Arrival Time	Burst Time
P1	0.0	8
P2	0.4	4
P3	1.0	1

a) What is the average turnaround time for these processes with the FCFS scheduling algorithm?
b) What is the average turnaround time for these processes with the SJF scheduling algorithm?
c) The SJF algorithm is supposed to improve performance but notice that we chose to run process P1 at time 0 because we did not know that two shorter processes would arrive soon. Compute what the average turnaround time will be if the CPU is left idle for the first unit and then SJF scheduling is used. Remember that processes P1 and P2 are waiting during this idle time, so their waiting time may increase. This algorithm could be known as future-knowledge scheduling.

7. Consider the following set of processes, with the length of the CPU burst time given in milliseconds:

Process	Burst Time	Priority
P1	2	2
P2	1	1
P3	8	4
P4	4	2
P5	5	3

The processes are assumed to have arrived in the order P1, P2, P3, P4, P5, all at time 0.

a) Draw four Gantt charts that illustrate the execution of these processes using the following scheduling algorithms: FCFS, SJF, non-preemptive priority (a larger priority number implies a higher priority), and RR (quantum = 2).
b) What is the turnaround time of each process for each of the scheduling algorithms in part a?
c) What is the waiting time of each process for each of these scheduling algorithms?
d) Which of the algorithms results in the minimum average waiting time (over all processes)?

8. Which of the following scheduling algorithms could result in starvation?
a. First-come, first-served
b. Shortest job first
c. Round robin
d. Priority

Answer:
Shortest job first and priority-based scheduling algorithms could result in starvation.