PMR5230 - Sistemas Computacionais para Automação

1a. Lista de Exercícios - Versão 2017

Capítulo 1

[Ex. 1] What are the three main purposes of an operating system ?

[Ex. 2] In a multiprogramming and time-sharing environment, several users share the system simultaneously. This situation can result in various security problems.

(a) What are two such problems ?

(b) Can we ensure the same degree of security in a time-shared machine as we have in a dedicated machine ?

[Ex. 3] Define the essential properties of the following types of operating systems: batch, interative, time-sharing, real time, network, parallel, distributed, clustered, handheld.

Capítulo 3

[Ex. 1] What are the five major activities of an operating system in regard to process management ?

[Ex. 2] What are the three major activities of an operating system in regard to memory management ?

[Ex. 3] What is the main advantage of the microkernel approach to system design ?

[Ex. 4] What is the main advantage of using a virtual machine architecture ?

Capítulo 4

[Ex. 1] MS-DOS provided no means of concurrent processing. Discuss three major complications that concurrent processing adds to an operating system.

[Ex. 2] Describe the actions taken by a kernel to switch context between processes.

[Ex. 3] The correct producer-consumer algorithm presented in Section 4.4 allows only $n - 1$ buffers to be full at any one time. Modify the algorithm to allow all buffers to be utilized fully.

[Ex. 4] Consider the interprocess-communication scheme where mailboxes are used.

(a) Suppose a process P wants to wait for two messages, one from mailbox A and one from mailbox B. What sequence of send and receive should it execute ?

(b) What sequence of send and receive should P execute if P wants to wait for one message from mailbox A or from mailbox B (or from both) ?

Capítulo 5

[Ex. 1] What are two diffrences between user-level threads and kernel level threads ? Under what circumstances is one type better than the other ?

[Ex. 2] Describe the actions taken by a kernel to context switch between kernel level threads.

[Ex. 3] What resources are used when a thread is created ? How do they differ from those used when a process is created ?

Capítulo 6

[Ex. 1] Define the difference between preemptive and nonpreemptive scheduling. State why strict nonpreemptive scheduling is unlikely to used in a computer center ?

[Ex. 2] Consider the following set of process, with the length of the CPU burst time given in milliseconds:

Process	Burts Time	Priority
P_1	10	3
P_2	1	1
P_3	2	3
P_4	1	4
P_5	5	2

The process are assumed to have arrived in the order P_1, P_2, P_3, P_4, P_5 all at time 0.

(a) Draw four Gantt charts illustrating the execution of these processes using FCFS, SJF a nonpremptive priority (a smaller priority number implies a higher priority) scheduling.

(b) What is the turnaround time of each process for each of the algorithms above ?

(c) What is the waiting time of each process for each of the algorithms above ?

(d) Which of the schedules above results in minimal average waiting time ?

[Ex. 3] Suppose that the following processes arrive for execution at the times indicated. Each process will run the listed amount of time. In answering the questions, use nonpreemptive scheduling and base all decisions on the information you have at the time the decision must be made.

Process	Arrival Time	Burst Time
P_1	0.0	8
P_2	0.4	4
P_3	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 P_1 at time 0. because we did not know that two shorter process would arrive soon. Compute what the average turnaround time will be if the CPU is left idle for the first 1 unit and then SJF scheduling is used. Remember that processes P_1 a P_2 are waiting during this idle time, so their waiting time may increase. This algorithm could be known as future-knowledge scheduling.