CS 5523: Operating Systems
Homework 3
!!!!! No late HW will be accepted !!!!!!
(Check <u>BlackBoard Learn</u> for due date and submission)

Objective

- Solve various problems from [SGG] ch 1 to ch 9

Problems

----

1. (10pt) **4.14** Consider a multiprocessor system and a multithreaded program written using the many-to-many threading model. Let the number of user-level threads in the program be more than the number of processors in the system. Discuss the performance implications of the following scenarios.
a. The number of kernel threads allocated to the program is less than the number of processors.
b. The number of kernel threads allocated to the program is equal to the number of processors.
c. The number of kernel threads allocated to the program is greater than the number of processors but less than the number of user level threads.

----

2. (10pt) **5.12** Consider the following set of processes, with the length of the CPU burst time given in milliseconds:

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

The processes 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 that illustrate the execution of these processes using the following scheduling algorithms: FCFS, SJF, non-preemptive priority (a smaller priority number implies a higher priority), and RR (quantum = 1).
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)?

----

3. (10pt) **5.15** Consider a system running ten I/O-bound tasks and one CPU-bound task. Assume that the I/O-bound tasks issue an I/O operation once for every millisecond of CPU computing and that each I/O operation takes 10 milliseconds to complete. Also assume that the context-switching overhead is 0.1 millisecond and that all processes are long-running tasks. Describe the CPU utilization for a round-robin scheduler when:
a. The time quantum is 1 millisecond
b. The time quantum is 10 milliseconds

----

4. (10pt) **6.8** Race conditions are possible in many computer systems. Consider a banking system with two methods: deposit(amount) and withdraw(amount). These two methods are passed the amount that is to be deposited or withdrawn from a bank account. Assume that a husband and wife share a bank account and that concurrently the husband calls the withdraw() method and the wife

calls deposit(). Describe how a race condition is possible and what might be done to prevent the race condition from occurring.

----

5. (10pt) **6.26** Consider a system consisting of processes P_1, P_2, ..., P_n, each of which has a unique priority number. Write a monitor that allocates three identical line printers to these processes, using the priority numbers for deciding the order of allocation. (pseudo code is enough)

----

6. (10pt) **7.13** In a real computer system, neither the resources available nor the demands of processes for resources are consistent over long periods (months).
Resources break or are replaced, new processes come and go, new resources
are bought and added to the system. If deadlock is controlled
by the banker's algorithm, which of the following changes can be made
safely (without introducing the possibility of deadlock), and under what
circumstances?
a. Increase *Available* (new resources added)
b. Decrease *Available* (resource permanently removed from system)
c. Increase *Max* for one process (the process needs or wants more resources than allowed).
d. Decrease *Max* for one process (the process decides it does not need
that many resources)
e. Increase the number of processes
f. Decrease the number of processes

----

7. (10pt) **7.17** Consider the following snapshot of a system:

	Allocation	Max	Available
	A B C D	A B C D	A B C D
P_0	0 0 1 2	0 0 1 2	1 5 2 0
P_1	1 0 0 0	1 7 5 0	
P_2	1 3 5 4	2 3 5 6	
P_3	0 6 3 2	0 6 5 2	
P_4	0 0 1 4	0 6 5 6	

Answer the following questions using the banker's algorithm:
a. What is the content of the matrix *Need*?
b. Is the system in a safe state?
c. If a request from process P_1 arrives for (0,4,2,0), can the request be granted immediately?

----

8. (5pt) **8.18** Consider a logical address space of 32 pages with 1024 words per page; mapped onto a physical memory of 16 frames.
a. How many bits are required in the logical address?
b. How many bits are required in the physical address?

----

9. (5pt) **8.19** Consider a computer system with a 32-bit logical address and 4-KB page size. The system supports up to 512 MB of physical memory. How many entries are there in each of the following?
a. A conventional single-level page table?
b. An inverted page table?

----

10. (10pt) **9.20** A certain computer provides its users with a virtual-memory space of 2^{32} bytes. The computer has 2^{18} bytes of physical memory. The virtual memory is implemented by paging, and the page size is 4096 bytes. A user process generates the virtual address 11123456. Explain how the system establishes the corresponding physical location. Distinguish between software and hardware operations.

----

11. (10pt) **9.31** A page-replacement algorithm should minimize the number of page faults. We can achieve this minimization by distributing heavily used pages evenly over all of memory, rather than having them compete for a small number of page frames. We can associate with each page frame a counter of the number of pages associated with that frame. Then, to replace a page, we can search for the page frame with the smallest counter.
a. Define a page-replacement algorithm using this basic idea. Specifically address these problems:
i. What the initial value of the counters is
ii. When counters are increased
iii. When counters are decreased
iv. How the page to be replaced is selected
b. How many page faults occur for your algorithm for the following reference string, for four page frames?
1, 2, 3, 4, 5, 3, 4, 1, 6, 7, 8, 7, 8, 9, 7, 8, 9, 5, 4, 5, 4, 2.
c. What is the minimum number of page faults for an optimal page replacement strategy for the reference string in part b with four page frames?

Grading: This is a 100-point homework.