

Midterm
April 8, 2005 11.40–13.30
Good Luck!

Part A. Short Answer, answer the questions with one or two sentences (each 5 pt)

1 What are the two main purposes of an operating system?

2 i. Why is the process table needed in a time sharing system?

ii. Is process table also needed in personal computer systems in which only one process exists, that process taking over the entire machine until it is finished?

3 In the solution to the dining philosophers problem, why is the state variable set to **HUNGRY** in the procedure *take_forks*?

(each 8 pt)

4 Why caches are useful? What problems do they solve? What problems do they cause? If a cache can be made as large as the device for which it is caching (for instance, a cache as large as a disk), why not make it that large and eliminate the device?

5 What is a Critical Region? List and explain the four conditions that need to be satisfied to solve the critical-region problem?

6 What are the four conditions for deadlock to occur? Describe them briefly.

It takes 15 msec to get a request for work, dispatch it, and do the rest of the necessary processing, assuming that the data needed are in the block cache. If a disk operation is needed, as is the case one-third of the time, an additional 75 msec is required, during which time the thread sleeps. How many requests/sec can the server handle if it is single threaded? If it is multithreaded?

8 Does Peterson's solution to the mutual exclusion problem work when process scheduling is preemptive? How about when it is nonpreemptive?

9 (20 Pts.) The Sleeping-Barber Problem. A barbershop consists of a waiting room with n chairs and the barber room containing the barber chair. If there are no customers to be served, the barber goes to sleep. If a customer enters the barbershop and all chairs are occupied, then the customer leaves the shop. If the barber is busy but chairs are available, then the customer sits in one of the free chairs. If the barber is asleep, the customer wakes up the barber. Write pseudo code to coordinate the barber and the customers.

Hints: Need a global counter for the number of customers that are currently waiting in a chair. Use 3 semaphores - one for the customers (barber waits til a customer signals they are ready), one for the barber (customer waits til barber signals he is ready) and one for the critical section (where the global counter is modified).

second one and the second process is the parent of the third one. Your program should print;

- *pid* and *parent pid* of each process.
- *parent pid* of the first process
- *group id* of the processes.

given in milliseconds. Arrival time is the time at which the process is added to the ready queue.

Process	Burst Time	Arrival Time
P1	30	0
P2	15	0
P3	9	0
P4	12	0
P5	6	12
P6	3	18

- a Draw appropriate charts illustrating the execution of these processes using FCFS, SJF non-preemptive, and SJF preemptive.
- b Calculate the wait times of each process for each strategy. Calculate the average wait times under each strategy.

Process	FCFS	SJF-nonpre	SJF
P1			
P2			
P3			
P4			
P5			
P6			
Average			