Problem 1 (0 Points)

The task graph below shows the desired execution order of 4 tasks (T0, T1, T2, T3). An arrow from Ti to Tj indicates that task i must be executed before task j. For example, T0 must execute before T1, and T2. T1 and T2 can execute in parallel. And T3 cannot execute until T1 and T2 are done.

-  

a) Complete the program shown below. The meaning of the parbegin construct below is that P0(), P1(), P2() and P3() can be done in parallel subject to synchronization dependencies. Pi should call the function Ti() and contain the synchronization code necessary to achieve the precedence expressed by the task graph above. Your solution should use semaphores and achieve the maximum parallelism expressed by the task graph. NOTE: This is NOT a machine problem since there is no such real construct as parbegin on our systems.

... supply declarations with initialization ...
P0() { ... supply code ...}
P1() { ... supply code ...}
P2() { ... supply code ...}
P3() { ... supply code ...}
parbegin ( P0(), P1(), P2(), P3() );

b) Write a program using parbegin but no semaphores that is equivalent to the one in Part a. Assume now that Ti is a process. Note that parbegin can be nested and each process instance terminates only when all statements enclosed in parentheses have terminated.

c) What forms of task graphs CAN NOT be implemented using just the parbegin and sequence (... ; ... ;) control constructs? Give the simplest example of such a graph.
Problem 2 (0 Points)
Tanenbaum, Problem 40 (Chapter 2).

Problem 3 (2 Points)

- Consider the case $N = 2$ where $t(1) = 5$ and $t(2) = 3$. What is the average response time for the two possible service orderings?

- Prove that for arbitrary $N$ and service demands of $t(i)$, $i=1:N$ that SJF will result in the smallest average response time. HINT: Consider proof by contradiction.

Problem 4 (2 Points)
Suppose that a user's last four requests have used 10, 20, 30 and 40 milliseconds of CPU time respectively. What is the predicted CPU demand if we use an aging algorithm with $a = 1/2$?

Problem 5 (0 Points)
Consider a dynamically partitioned memory that has two holes with sizes: 1300 KB and 1200 KB. Assume that memory requests arrive in the following order: A) 1000 KB, B) 1100 KB, and C) 250 KB. Compare the results of the four allocation algorithms first-fit, best-fit, next-fit, and worst-fit.

a) Use a table to show the evolution of free space as each memory request is handled. The table should have five columns (memory request, and free space sizes for the four algorithms) and four rows (initial free space, and each memory request).

b) For the above workload, rank the four algorithms by memory utilization. By speed.

c) Do the results for the above workload correspond to expectations for general workloads? Explain.

Problem 6 (0 Points)
Consider a buddy system and the address 011011110000.

a) If the block size associated with this address is 8 bytes, what is the binary address of the buddy?

b) What is the largest block size $N$ such that the above address still has a buddy? Explain.

Problem 7 (4 Points)
Consider a buddy system and the address 100100001000. Assume the largest possible $U$ and smallest possible $L$.

a) If the block size is 8 bytes, what is the binary address of the buddy?

b) What is the largest block size $N$ such that the above address still has a buddy? Explain.

c) Let $b_k(x)$ be the buddy of address $x$ with block size $2^k$. Write an expression (NOT an algorithm) for $b_k(x)$. Explain why the form of the expression is correct.

d) Demonstrate that your expression in Part c is correct.
Problem 8 (0 Points)

If page table entries are 4 bytes each and the page size is 8 KB, how many levels of page tables would be required to map a 32-bit address space if the top level page table fits into a single page?

Problem 9 (2 Points)

Suppose that a logical address space has eight pages where each page is 1,024 bytes and this logical address space is mapped onto a physical memory of 32 page frames.

a) How many bits are there in the logical address?

b) How many bits are there in the physical address?

c) Describe the structure of the page table for this system.