I have read, understood, and agree to abide by the
Honor Code of the School of Engineering.

Name: ____________________________  

ID: _______________________________  

Signature: __________________________

Date: ______________________________

1. [10 points] Please give the whole names for the following acronyms: 
   CDE, JIT, PCB, SJF, TSS.

2. [40 points] True or false (yes or no, 1 or 0) problems with wrong-answer penalties: 
   a) It takes less time to terminate a process than a thread.
   b) The interrupt can occur at any time and therefore at any point in the execution of a user program.
   c) In a time sharing system, a user's program is preempted at regular intervals, but due to relatively slow human reaction time this occurrence is usually transparent to the user.
   d) It is possible in a single-processor system to not only interleave the execution of multiple processes but also to overlap them.
   e) The OS may create a process on behalf of an application.
   f) A hardware mechanism is needed for translating relative addresses to physical main memory addresses at the time of execution of the instruction that contains the reference.
   g) The size of virtual storage is limited by the actual number of main storage locations.
   h) The addresses a program may use to reference memory are distinguished from the addresses the memory system uses to identify physical storage sites.

3. [50 points] Simple questions: 
   a) Consider a computer system that has cache memory, main memory (RAM) and disk, and the operating system uses virtual memory. It takes 1 nsec to access a word from the cache, 10 nsec to access a word from the RAM, and 10 ms to access a word from the disk. If the cache hit rate is 95% and main memory hit rate (after a cache miss) is 99%, what is the average time to access a word?
   b) Consider a pure paging system that uses three levels of page tables and 64-bit addresses. Each virtual address is the ordered set $v = (p, m, t, d)$, where the ordered triple $(p, m, t)$ is the page number and $d$ is the displacement into the page. Each page table entry is 64 bits (8 bytes). The number of bits that store $p$ is $np$, the number of bits that
4. [20 points] Given that main memory is composed of three page frames for public use and that a program requests pages in the following order: a, d, b, a, f, b, e, c, g, f, b, g,
   a) Using the FIFO page removal algorithm, perform a page trace analysis indicating page faults with asterisks (*). Then compute the failure and success ratios.
   b) Using the LRU page removal algorithm, perform a page trace analysis and compute the failure and success ratios. (The least recently used page is shown in bold.)

5. [20 points] Consider a preemptive priority scheduling algorithm based on dynamically changing priorities. Larger priority numbers imply higher priority. When a process is waiting for the CPU (in the ready queue, but not running), its priority changes at a rate $\alpha$; when it is running, its priority changes at a rate $\beta$. All processes are given a priority of 0 when they enter the ready queue. The parameters $\alpha$ and $\beta$ can be set to give many different scheduling algorithms.
   a) What is the algorithm that results from $\beta > \alpha > 0$?
   b) What is the algorithm that results from $\alpha < \beta < 0$?

6. [20 points] A multiprocessor with eight processors has 20 attached tape drives. There is a large number of jobs submitted to the system that each require a maximum of four tape drives to complete execution. Assume that each job starts running with only three tape drives for a long period before requiring the fourth tape drive for short period toward the end of its operation. Also assume an endless supply of such jobs.
   a) Assume the scheduler in the OS will not start a job unless there are four tape drives available. When a job is started, four drives are assigned immediately and are not released until the job finishes. What is the maximum number of jobs that can be in process at once? What are the maximum and minimum number of tape drives that may be left idle as a result of this policy?
   b) Suggest an alternative policy to improve tape drive utilization and at the same time avoid system deadlock. What is the maximum number of jobs that can be in progress at once? What are the bounds on the number of idling tape drives?

7. [20 points] A swapping system eliminates holes by compaction. Assuming a random distribution of many holes and many data segments and a time to read or write a 32-bit memory word of 4 nsec, about how long does it take to compact 4 GB? For simplicity, assume that word 0 is part of a hole and that the highest word in memory contains valid data.

8. [20 points] In 1978, Dijkstra put forward the conjecture that there was no solution to the mutual exclusion problem avoiding starvation, applicable to an unknown but finite number of processes, using a finite number of weak semaphores. In 1979, J.M. Morris refuted this conjecture by publishing an algorithm using three weak semaphores. The behavior of the algorithm can be described as follows: If one or several process are waiting in a semWait(S) operation and another process is executing semSignal(S), the value of the semaphore S is not modified and one of the waiting processes is unblocked independently of semWait(S). Apart from the three semaphores, the algorithm uses two of the nonnegative integer variables as counters of the number of processes in certain sections of the algorithm. Thus, semaphores A and B are initialized to 1, while semaphore M and counters NA and NM are initialized to 0. The mutual exclusion semaphore B protects access to the shared variable NA. A process attempting to enter its critical section must cross two barriers represented by semaphores A and M. Counter NA and NM, respectively, contain the number of processes ready to cross barrier A and those having already crossed barrier A but not yet barrier M. In the second part of the protocol, the NM processes blocked at M will enter their critical sections one by one, using a cascade technique similar to that used in the first part. Define an algorithm that conforms to this description.