**CS 451 TEST #3 NAME:**

**TERMINOLOGY**: Fill in the blank with the number matching the best term from the "terms section". Terms may be used more than once. (1 point per answer)

1. \_\_1\_\_ is the method that allows a process's priority to increase as it gets older.
2. \_\_10\_\_ is the CPU scheduling algorithm where the process that requests the CPU first is given it.
3. \_\_10\_\_ is the disk scheduling algorithm that moves the drive head to the next requested block. This method does not do any presorting of any kind.
4. \_\_22\_\_ is the name given to a disk when it is used to store memory pages.
5. \_\_3\_\_ is what it is called when the CPU is switched to another process.
6. \_\_41\_\_ is the name given to a program that was deliberately written to disguise itself as a legitimate program.
7. \_\_5\_\_, \_\_\_22\_, and \_\_15\_\_ are the three types of drive space allocation methods.
8. \_\_22\_\_ is the type of memory allocation that may involve the first-fit, best-fit, and worst-fit memory allocation algorithms.
9. \_\_39\_\_ is sometimes called a lightweight process, these can be user-level or kernel-level.
10. \_\_26\_\_ is the name of the block of code that when a process is executing in this block of code it is very important that no other process also be executing in this block of code.
11. \_\_21\_\_ is caused when a process is spending more time generating page faults than processing.
12. \_\_7\_\_ is when two or more processes are waiting for each other.

1. \_\_37\_\_ is when an entire process, including its state and active memory, is moved from memory to storage.
2. \_\_9\_\_ is what it is called when there are little sections of memory unused and scattered about.
3. \_\_36\_\_ is when a process is left waiting for a very long period of time.
4. \_\_17\_\_ is the page replacement algorithm that replaces the oldest page first.
5. \_\_35\_\_ is the disk scheduling algorithm that presorts the block requests to minimize the drive head move distance.
6. \_22\_\_\_ is the class of page replacement algorithms that contributes to thrashing.
7. \_\_34\_\_ is the CPU scheduling algorithm is where the process which requires the least amount of CPU time (before the next I/O) is given the CPU.
8. \_\_5\_\_ is when the allocated memory is slightly larger (not enough to bother with) than what is required.
9. \_\_16\_\_ is the class of page replacement algorithms that prevents thrashing.
10. \_\_33\_\_ is a region of memory that can be accessed by multiple programs.
11. \_\_11\_\_ is the page replacement algorithm that replaces the page that has not been used in the longest time.
12. \_\_32\_\_ is one way to synchronize heavy weight programs (to prevent two programs from entering their critical sections).
13. \_\_18\_\_ is the CPU scheduling algorithm where there may be many levels of priorities. Once a process is assigned to a priority level it does not have to remain at that level.
14. \_\_\_\_ is the disk scheduling algorithm that moves the drive head from the outer track towards the inner track and back out again servicing block requests along the way.
15. \_\_19\_\_ is the CPU scheduling algorithm where there may be many levels of priorities. Once a process is assigned to a priority level it remains at that level.
16. \_\_30\_\_ is the CPU scheduling algorithm designed especially for time-shared systems. Each process gets a time-slice with this method. Also, what is the time slice called? \_\_quanta\_\_
17. \_\_20\_\_ is when memory pages are not loaded until needed.
18. \_\_5\_\_ is the RAID level stripes that data, but also uses blocks to store parity information that can be used to rebuild data lost to any one disk failure.
19. \_\_21\_\_ is what is generated when a program requires a page to be loaded from the drive.
20. \_\_19\_\_ is the method for memory allocation used by most modern OS’s.
21. \_\_0\_\_ What RAID level stripes data blocks across disks without any parity computations, but is not considered a "True" RAID because it is NOT fault-tolerant?
22. \_\_23\_\_ is an interprocess communication device that is commonly assumed to be unidirectional, but which is really bidirectional.
23. \_\_26\_\_ represents each process in the operating system.
24. \_\_24\_\_ is the CPU scheduling algorithm is where the processes are ordered strictly by some significance criteria.
25. \_\_25\_\_ is a program in execution.

**TERMS ALLOWED (for TERMINOLOGY section):**

2. Aging
3. Backing store
4. Context switch
5. Contiguous
6. Contiguous memory allocation
7. Critical section
8. Deadlock
9. Dup
10. External fragmentation
11. FCFS - First Come First Serve
12. FIFO - First In First Out
13. Global
14. Indexed
15. Internal fragmentation
16. Linked
17. Local
18. LRU - Least Recently Used
19. Multi-level feedback queue
20. Multi-level queue
21. On demand paging
22. Page fault
23. Paged memory allocation
24. Pipe
25. Priority
26. Process
27. Process control block
28. RAID 0
29. RAID 1
30. RAID 5
31. Round robin
32. SCAN
33. Semaphore
34. Shared memory
35. SJF - Shortest Job First
36. SSTF - Shortest Seek Time First
37. Starvation
38. Swapping
39. Thrashing
40. Threads
41. Time quantum
42. Trojan horse
43. Worm

**Multiple Choice: 1 point per correct answer.**

1. What types of real-time systems are there?
2. hard
3. soft
4. random task
5. aperiodic task
6. periodic task
7. Gang task
8. What are the costs associated with accessing a hard drive?
9. Seek time
10. Rotational delay
11. Transfer time
12. CPU burst time
13. Page replacement time
14. IDE drives commonly use what head scheduling method?
15. SCAN
16. Eschenback scheme
17. C-SCAN
18. N step SCAN
19. SCSI drives commonly use what head scheduling method?
20. SCAN
21. Eschenback scheme
22. C-SCAN
23. N step SCAN
24. While DOS’s directory structure (a tree) does NOT allow cycles, does UNIX’s directory structure?
25. Yes – via soft/symbolic links
26. Yes – via hard links
27. No
28. What is the name of the deadlock prevention/avoidance algorithm used by UNIX?
29. Dekker’s
30. Peterson’s
31. Ostrich
32. Baker’s
33. Banker’s
34. Assume that 100 users are being served by two threaded processes (50 people per process) and that the average workload per person is about the same. No other processes are running. Which approach would provide the most uniform CPU time to each user?
35. User level threaded processes
36. Kernel level threaded processes
37. Wouldn’t matter
38. Assume that 100 users are being served by two threaded processes (5 people per process #1 and 95 people per process #2) and that the average workload per person is about the same. No other processes are running. Which approach would provide more uniform CPU time to each user?

a) User level threaded processes

b) Kernel level threaded processes

c) Wouldn’t matter

1. What computer security criteria guards against electronic eavesdropping by requiring EM shielding of the system:
2. Air Force Document DO-178B
3. Tempest
4. Faraday cage
5. DHS Document DHS-CBS
6. None of the above
7. What C function should never be used (allows buffer overflows and system security problems)?

## Gets

1. Puts
2. Fopen
3. Fclose
4. Seek
5. None of the above
6. What is the major cost associated with accessing a hard drive?
7. Seek time
8. Rotational delay
9. Transfer time
10. CPU burst time
11. In an environment where a CPU bound process is the only process allowed to run would you want a preemptive CPU scheduler or a non-preemptive CPU scheduler?
12. Preemptive
13. Non-preemptive
14. Compare the process control block (PCB) of a thread and a program:
15. The PCB of the thread is “smaller”
16. The PCB of the program is “smaller”
17. The PCBs are the same
18. A thread is a lightweight process
19. A program is a heavyweight process
20. A thread does not have a PCB

**Short answer: Points as indicated.**

1. (9 points) What disk head scheduling algorithm (exact name required) would you use if your only hard disk was a “RAM disk” and why? (note that “RAM disk” is a disk drive ***simulated*** by memory – i.e. no moving parts) What performance factors would you need to consider for the RAM disk? Note: This is a three-part question.

I would use the shortest seek time first because it reorders drive requests so the head moves the least amount. Since there is no head actually moving, the benefits of it having high throughput and high variance would outweigh the other benefits.

1. (5 points): Given the fact that hackers are always trying to break into systems, that “stuff happens” (rats, fires, floods, terrorists, fires during floods, etc, etc) and the fact that loss of data could destroy a company. What is the single most important thing a company can do to prevent data loss? This is a short answer, but you must be specific and thorough.

The single most important thing a company can do is to have off site storage of data.

1. (5 points) In order to use semget() what header(s) must be included?

Sys/types.h

sys/ipc.h

sys/sem.h

1. (5 points) In order (poor to good), list the 7 levels of security from the U. S. DOD Trusted Computer System Evaluation:

Minimal, Discretionary (Disc. and Controlled), Mandatory (Labeled, Structured, Security Domains), Verified

1. (18 points) Consider a demand-paging system that uses a First-Come-First-Served head scheduler with the following time-measured parameters:

CPU 20% utilization

Paging disk I/O 99.999% utilization

Memory 95% consumption

We want to **improve CPU utilization**. Which of the following will (probably) improve CPU utilization? All require a “yes” or “no” indication – “maybe” is not acceptable. All answers require an explanation.

1. Install a dual-core CPU (same clock rate as the single core).

No. A faster CPU doesn’t solve the memory access problem.

1. Start additional CPU bound jobs.

Yes. More jobs mean the CPU will process more jobs instead of the slow input it is currently receiving.

1. Install a level 5 RAID for the swap partition.

No. This is good for a lot of existing jobs, not increasing jobs.

1. Install a level 1 RAID for the swap partition.

Yes. Distribution throughout the drives allows more memory and less paging to occur.

1. Change the page replacement from a global policy to a local policy.

No. This is for smaller loads. We are trying to increase the load.

1. Change the drive head scheduler to SCAN.

No. C-SCAN is better for heavy loads

1. Change the drive head scheduler to C-SCAN.

Yes. Good for heavy loads.

1. Shout at your disks to make them work faster.

No. They don’t have ears.

**Answers to all of these questions can be obtained by watching the movie: Revolution OS. (2 points each)**

1. The GNU and free software movement started in what decade?

90’s.

1. According to Torvolds, the only mission in life for an operating system is to do what?

Help programs run.

1. According to Stallman, administrators use what to control computer users?

Proprietary Software.

1. What was the return address of Bill Gates’ January 31, 1976 letter to the HomeBrew Computer Club?

New Mexico

1. What is the difference between “Bison” and “Bison++” (I am not referring to NDSU!)?

You can create C++ with Bison.

1. What was the name of the first company that specialized in free software”

Hewlett Packard

1. The name of the company referenced in the previous question is “special,” why?

First tech garage startup company

1. The first version of Linux had how many lines of code?

about 10,000

1. The first killer application for Linux was what?

There are no killer apps for Linux

1. In what type of building did RedHat Linux get its start?

catalog business