L2910-5335 Final Exam VerB

Phong Vo

TOTAL POINTS

96 / 100

QUESTION 1

1 Microkernel **2 / 2**

✓ **- 0 pts** Correct

QUESTION 2

2 Policy **2 / 2**

✓ **- 0 pts** Correct

QUESTION 3

3 Stack section **2 / 2**

✓ **- 0 pts** Correct

QUESTION 4

4 Child process **2 / 2**

✓ **- 0 pts** Correct

QUESTION 5

5 Modules **2 / 2**

✓ **- 0 pts** Correct

QUESTION 6

6 Message passing **2 / 2**

✓ **- 0 pts** Correct

QUESTION 7

7 Spinlocks **2 / 2**

✓ **- 0 pts** Correct

QUESTION 8

8 Thread signal handling **2 / 2**

✓ **- 0 pts** Correct

QUESTION 9

9 Thread-local storage **2 / 2**

✓ **- 0 pts** Correct

QUESTION 10

10 Priority inversion **2 / 2**

✓ **- 0 pts** Correct

QUESTION 11

11 Named pipes **0 / 2**

✓ **- 2 pts** Incorrect (correct answer is "named pipe")

QUESTION 12

12 Pipe = file **2 / 2**

✓ **- 0 pts** Correct

QUESTION 13

13 Critical section requirements **2 / 2**

✓ **- 0 pts** Correct

QUESTION 14

14 Signal() before wait() **2 / 2**

✓ **- 0 pts** Correct

QUESTION 15

15 Logical address -> page number **2 / 2**

✓ **- 0 pts** Correct

QUESTION 16

16 Inverted or hashed page table **2 / 2**

✓ **- 0 pts** Correct (either "inverted page table" or "Hashed page table" is acceptable)

QUESTION 17

17 Safety algorithm **2 / 2**

✓ **- 0 pts** Correct

QUESTION 18

18 SCAN **2 / 2**

✓ **- 0 pts** Correct

QUESTION 19

19 Running state 2 / 2

✓ - 0 pts Correct

QUESTION 20

20 Mach 2 / 2

✓ - 0 pts Correct

QUESTION 21

21 Amdahl's law 2 / 2

✓ - 0 pts Correct

QUESTION 22

22 Kernel 2 / 2

✓ - 0 pts Correct

QUESTION 23

23 API 2 / 2

✓ - 0 pts Correct

QUESTION 24

24 mutex=binary semaphore 2 / 2

✓ - 0 pts Correct

QUESTION 25

25 non-preemptive kernel 2 / 2

✓ - 0 pts Correct

QUESTION 26

26 hashed page tables 0 / 2

✓ - 2 pts Incorrect (correct answer is False)

QUESTION 27

27 TLB miss 2 / 2

✓ - 0 pts Correct

QUESTION 28

28 deadlock-free / starvation 2 / 2

✓ - 0 pts Correct

QUESTION 29

29 thread components 2 / 2

✓ - 0 pts Correct

QUESTION 30

30 LRU 2 / 2

✓ - 0 pts Correct

QUESTION 31

31 OS modes 10 / 10

✓ - 0 pts Correct

QUESTION 32

32 Working set 10 / 10

✓ - 0 pts Correct (or close enough for full credit!)

QUESTION 33

33 RAID levels 10 / 10

✓ - 0 pts Correct (or close enough for full credit!)

QUESTION 34

34 Intel five levels 10 / 10

✓ - 0 pts Correct

Name: PHONG VO Student ID: 017490283 (b)

Final Exam

COMP.3080-201-202 – Operating Systems; December 19, 2019 – Dr. Wilkes

Note: *This exam is closed book and notes, except for up to three 8.5x11" sheets of paper with handwritten notes (no photocopies).*

Multiple Choice Questions – 2 points each: Mark the correct answer(s).

1. **(MARK A SINGLE ANSWER)** A microkernel is a kernel ____.
   - ○ containing many components that are optimized to reduce resident memory size
   - ○ that is compressed before loading in order to reduce its resident memory size
   - ○ that is compiled to produce the smallest size possible when stored to disk.
   - ● that is stripped of all nonessential components
2. **(MARK A SINGLE ANSWER)** Policy ____.
   - ○ determines how to do something
   - ● determines what will be done
   - ○ is not likely to change across places
   - ○ is not likely to change over time
3. **(MARK A SINGLE ANSWER)** The ____ of a process contains temporary data such as function parameters, return addresses, and local variables.
   - ○ data section
   - ○ program counter
   - ● stack section
   - ○ text section
4. **(MARK ALL THAT APPLY)** When a child process is created, which of the following is a possibility in terms of the execution or address space of the child process?
   - ■ The child process is a duplicate of the parent process.
   - ■ The child process has a new program loaded into it.
   - ■ The child process runs concurrently with the parent.
   - ☐ None of the above
5. **(MARK A SINGLE ANSWER)** _____ allow operating system services to be loaded dynamically.
   - ○ Virtual machines
   - ● Modules
   - ○ File systems
   - ○ Graphical user interfaces
6. **(MARK A SINGLE ANSWER)** Microkernels use _____ for communication.
   - ● message passing
   - ○ shared memory
   - ○ system calls
   - ○ virtualization

7. (**MARK A SINGLE ANSWER**) Which of the following statements is true?
   - ○ A binary semaphore can never be used as a counting semaphore.
   - ○ A counting semaphore can never be used as a binary semaphore.
   - ● Counting semaphores can be used to control access to a resource with a finite number of instances.
   - ○ Spinlocks can be used to prevent busy waiting in the implementation of semaphore.
8. (**MARK A SINGLE ANSWER**) Which of the following would be an acceptable signal handling scheme for a multithreaded program?
   - ○ Deliver the signal to the thread to which the signal applies.
   - ○ Deliver the signal to every thread in the process.
   - ○ Deliver the signal to only certain threads in the process.
   - ● All of the above
9. (**MARK A SINGLE ANSWER**) Thread-local storage is data that ____.
   - ○ is generated by the thread independent of the thread's process
   - ○ is not associated with any process
   - ● is unique to each thread
   - ○ is not associated with any process
10. (**MARK A SINGLE ANSWER**) _____ occurs when a higher-priority process needs to access a data structure that is currently being accessed by a lower-priority process.
    - ○ A critical section
    - ○ A race condition
    - ○ Deadlock
    - ● Priority inversion
11. (**MARK A SINGLE ANSWER**) A(n) _____ allows several unrelated processes to use the pipe for communication.
    - ● anonymous pipe
    - ○ LIFO pipe
    - ○ named pipe
    - ○ ordinary pipe
12. (**MARK A SINGLE ANSWER**) Child processes inherit UNIX ordinary pipes from their parent process because:
    - ○ All IPC facilities are shared between the parent and child processes.
    - ● A pipe is treated as a file descriptor and child processes inherit open file descriptors from their parents.
    - ○ The pipe is part of the code and children inherit code from their parents.
    - ○ The STARTUPINFO structure establishes this sharing.
13. (**MARK A SINGLE ANSWER**) A solution to the critical section problem does not have to satisfy which of the following requirements?
    - ● Atomicity
    - ○ Bounded waiting
    - ○ Mutual exclusion
    - ○ Progress

14. **(MARK A SINGLE ANSWER)** When using semaphores, a process invokes the `wait()` operation before accessing its critical section, followed by the `signal()` operation upon completion of its critical section. Consider reversing the order of these two operations—first calling `signal()`, then calling `wait()`. What would be a possible outcome of this?
    - ○ Deadlock is possible.
    - ○ Mutual exclusion is still assured.
    - ● Several processes could be active in their critical sections at the same time.
    - ○ Starvation is possible.
15. **(MARK A SINGLE ANSWER)** Given the logical address 0xB8D3 (in hexadecimal) with a page size of 256 bytes, what is the logical page number?
    - ○ 0x3
    - ○ 0xB
    - ● 0xB8
    - ○ 0xD3
    - ○ 0x00D3
    - ○ 0x0003
16. **(MARK A SINGLE ANSWER)** Which of the following page table implementations would be most appropriate for an application environment in which the logical address space is sparsely populated (i.e., an application with a relatively small number of logical pages, but these logical pages are scattered throughout a large logical address space?
    - ○ Hashed page table
    - ○ Hierarchical page table
    - ● Inverted page table
    - ○ Multimodal page table
    - ○ None of the above
17. **(MARK A SINGLE ANSWER)** Suppose that there are ten copies of a resource available to three processes. At time 0, the data shown in the table below is collected. The table indicates the process ID, the maximum number of copies of the resource needed by the process, and the number of copies of the resource currently allocated to each process:

Process ID	Maximum need	Currently allocated
P0	10	4
P1	3	1
P2	6	4

Which of the following correctly characterizes the state of this system?

- ○ It is safe.
- ● It is not safe.
- ○ It is an impossible state.
- ○ The state cannot be determined.

18. (**MARK A SINGLE ANSWER**) Consider a disk queue holding requests to the following cylinders in the listed order: 116, 22, 3, 11, 75, 185, 100, 87. Using the SCAN scheduling algorithm, what is the order that the requests are serviced, assuming the disk head initially is at cylinder 88 and moving upward (toward higher cylinder numbers)?

- ○ 87 - 75 - 100 - 116 - 185 - 22 - 11 – 3
- ○ 100 - 116 - 185 - 3 - 11 - 22 - 75 – 87
- ● 100 - 116 - 185 - 87 - 75 - 22 - 11 - 3
- ○ 116 - 22 - 3 - 11 - 75 - 185 - 100 – 87

Name: PHONG UO Student ID: 017902B3 (b)

True/False Questions – 2 points each: Mark the correct answer.

19. For a single-processor/single-core system, there can be multiple processes in the Running state.
    - ○ True
    - ● False

20. The Mach operating system treats system calls with message passing.
    - ● True
    - ○ False

21. Amdahl's Law describes performance gains for applications with both a serial and parallel component.
    - ● True
    - ○ False

22. The operating system kernel consists of all system and application programs in a computer.
    - ○ True
    - ● False

23. Application programmers typically use an API rather than directly invoking system calls.
    - ● True
    - ○ False

24. Mutex locks and binary semaphores are essentially the same thing.
    - ● True
    - ○ False

25. A nonpreemptive kernel is safe from race conditions on kernel data structures.
    - ● True
    - ○ False

26. Hashed page tables are not useful when handling addresses larger than 32 bits.
    - ● True
    - ○ False

27. A page fault must be preceded by a TLB miss.
    - ● True
    - ○ False

28. A deadlock-free solution eliminates the possibility of starvation.
    - ○ True
    - ● False

29. Each thread has its own register set and stack.
    - ● True
    - ○ False

30. When using an LRU approximation page-replacement algorithm, if the page-fault rate is too high, the process may have too many frames.
    - ○ True
    - ● False

Name: PHONG VO Student ID: 017 90 283 (b)

Short Answer Questions – 10 points each: Write your answer in the space provided.

31. Describe the operating system's two modes of operation. Two modes: kernel mode and user mode, where the OS runs, is also known as privileged or supervisor mode. The software in kernel mode can directly access the hardware and can control the switching between CPU modes. Rest of the software runs in user mode. Hardware direct access is restricted in user mode. The program in user mode has enough memory to do its job. It has given an address space which is only visible to them.

32. Many operating systems (including Microsoft Windows 10) use the "working set" model to determine which logical (virtual) pages of a process should be held in physical memory frames at a given time in order to minimize the number of page faults. Briefly explain the working set model, including what happens to the page fault rate when a process transitions between regions of locality in the program code.

- Working set is used to allocate the location of logical (virtual) pages will be held in physical memory frames. Access to a page marked invalid causes a page fault. The working set will notice that the invalid bit is set, causing a trap to the OS. This trap is the result of the OS's failure to bring the desired page into memory.

Name: _____ Student ID: _____ (b)

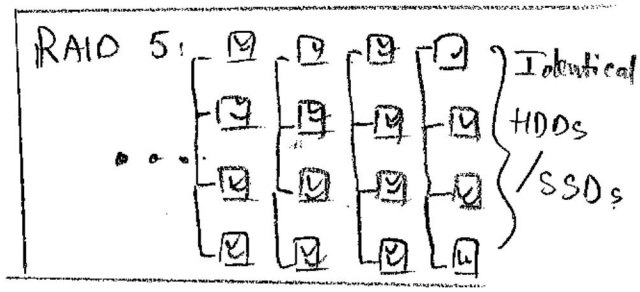
33. Describe RAID levels 0, 1, and 5. Include sample diagrams for each of these three RAID levels showing how data blocks, as well as parity blocks (if applicable), are distributed among the disks. (In your diagrams, use the minimum number of disks necessary for each level.)

RAID 0:

RAID 1:

} Hard drives / SSDs must be identical.

RAID 5:



RAID 0: Non-Redundant Stripping

RAID 1 Mirroring and Shadowing

RAID 5: Block Interleaved Distributed parity

34. Intel's original "IA-32e mode" memory addressing scheme had a hierarchical page table with four levels. Explain why Intel recently added a fifth level of hierarchy to the x86-64 (IA-32e) page tables.

With 5 level paging, the virtual address size increases from a 256TB maximum to 128 PB while the physical address size threshold goes from 64TB to 4PB. This big set of patches increases the virtual or physical address space capacity of the Linux kernel for the future Intel x86-64 hardware which can currently be found via this kernel mailing list thread.