CS 4375, Operating Systems, Test 2

Fall 2024, October 31 name _____

78 minutes 10:32 – 11:50

Two pages of handwritten notes are allowed. There are no room re-entry privileges.

/50

1. [13 points] True or False
   - T F The sum of all processes' virtual memory always equals the size of physical memory.
   - T F Cache memory is usually faster to access than main memory.
   - T F Cache memory is usually faster to access than cloud storage.
   - T F Zombie processes (processes which have ended and will never run again) can occupy physical memory.
   - T F Nonblocking `recv()` is like a `recv()` with a timeout of 0.
   - T F 0x7A equals 133_{10}
   - T F If a client process and a server process are running on different machines and communicating via a network, they can synchronize their efforts using semaphores.
   - T F `htonl(0) == 0` on all processors and for all networks (hint: htonl = host to network long-integer conversion)
   - T F `htonl(htonl(x)) == x` for all `x`, on all processors and for all networks
   - T F `htonl(ntohl(x)) == x` for all `x`, on all processors and for all networks
   - T F A process switch is always faster than a thread switch.
   - T F A process switch is sometimes faster than a thread switch.
   - T F Condition variables are always used in pairs.
   - T F A "race condition" in code implies vulnerability to interuption at untimely points in the execution.

2. [1] Because Posix is an international standard
   a. Many OSs run the same kernel code
   b. Many OSs support the same system calls
   c. Many processors have the same number of registers
   d. all of the above
   e. none of the above

3. [1] Fill in the blank:
   A thread which never accesses _____ is guaranteed to have no critical sections.

4. [1] If a local-machine process does a `send()` before doing a `connect()`, what would you expect to happen?
   a. Compile-time error
   b. Run-time error on the local-machine process
   c. Run-time error on the remote process that it's trying to communicate with
   d. Deadlock
   e. No error

5. [1] What happens on access to a virtual memory address for which the page exists in physical memory?
   a. The low-order bits of the Program Counter are XORed with the segment mask and the garbage collector uses the result to flush the corresponding cache lines.
   b. The hardware (specifically the Memory Management Unit) finds the corresponding physical page location and issues an access to the appropriate memory address.
   c. The process issues a system call, and causes the OS to copy the appropriate page from disk to memory so that the process can subsequently directly load or store from the specified location.
   d. none of the above

/18

6. [1] There is recent interest in Memory Encryption Contexts, which allows the memory associated with a process to be uniquely encrypted relative to other processes, etc. An advantage of this would be that it

   a) Reduces the amount of physical memory needed

   b) Reduces the fraction of context switches that require swapping

   c) Provides security even if the isolation created through the page table (or segment table) can't be fully trusted

   d) all of the above

   e) none of the above

7. [1] Which of the following would make an operating system ridiculously slow?
   a. Trapping and switching to kernel mode on every system call
   b. Saving the entire state of all the registers to memory on every context switch
   c. Saving the entire contents of a process's memory to disk on every context switch
   d. all of the above

8. [1] "Confidential computing" has the potential to improve the security of general-purpose computing platforms by taking supervisory systems out of the trusted computing base, thereby reducing the attack surface (the number of attack vectors that defenders must consider).
   Which of the following is compatible with this statement?

   a) "Confidential computing" guarantees truly anonymous messaging.

   b) "Confidential computing" guarantees that the OS cannot access or modify code or data from any process.

   c) "Confidential computing" guarantees that system calls are handled entirely by each individual process, without involvement of the OS.

   d) all of the above

   e) none of the above

9. [2] Which of the following leads to a trap, followed by a context switch and the execution of some OS code?
   a. Calling a function located at a location that is off-limits to user processes
   b. Access to a virtual memory location for which the data has been paged out to disk
   c. A timer interrupt
   d. Attempting to execute a privileged instruction
   e. all of the above
   f. none of the above

10. [2] If a thread acquires a lock, which 2 of the following will *not* result in the lock being released?
    a. The thread terminates
    b. The thread releases it
    c. Another thread releases it
    d. The process terminates
    e. The computer is rebooted
    f. Reaching 2 am in the local time zone

11. [2] Which two of the following are passed as arguments to socket.connect() ?
    a. the socket type
    b. the port number
    c. the address family
    d. the remote host name
    e. isBlocking (a boolean)
    f. the maximum number of threads

12. [1] A virtual address is
    a. A number
    b. An identifier (or name or symbol) in the relevant programming language (e.g. assembler)
    c. Always a power of two
    d. none of the above

13. [1] Which of the following is *not* a necessary condition for deadlock?
    a. Circular wait
    b. No pre-emption
    c. Hold-and-wait
    d. Locks

14. [1] Consider the following old Kansas law: "When two trains approach each other at a crossing, both shall come to a full stop, and neither shall start up again until the other is gone." The result could be
    a. Deadlock
    b. Thrashing
    c. both
    d. neither

15. [2] ReadyBoost is a disk caching software component developed by Microsoft for Windows. ReadyBoost enables USB flash drive devices to be used as a cache between the hard drive and random-access memory. True or False:
    a) **T F** Implementing Ready Boost probably required modifying the process scheduling algorithms.
    b) **T F** Implementing Ready Boost probably required modifying the paging algorithms

16. [2] What usually happens if a server calls `recv()` but the client that it is talking to has not yet done a `send()` ?

17. [5] Imagine that *n* threads are simultaneously solving *n* sudoku puzzles. At the end of each thread are the following two lines of code:

```
puzzlesSolved++

puzzlesRemaining --
```

Where both of these variables are static and thus shared across threads. Add one or more locks to prevent race conditions. How many locks does your solution have? Is that the only possible number of locks? Why or why not?

18. [2] Answer one of the following:
    a. What does the OS do if a thread calls semaphore.release() ?
    b. What does the OS do if a thread calls conditionVariable.notify() ?

19. [5] Katalin proposes a new process scheduling algorithm, to replace round-robin:

    > "from the ready queue, always choose the process which has the greatest fraction of its pages in memory (and thus the smallest fraction swapped out to disk)"

    Her logic is that this will minimize the time spent, first by reducing the time copying pages from disk to memory for the incoming process, and second by reducing the time spent freeing up space for this by copying other processes' pages from memory out to disk. Explain two problems with her proposal.

20. [4] A radiation treatment machine for cancer patients needs to do many things, including: read input from the keyboard (to get the location, time and intensity parameters), turn on and off the magnetron (both to avoid overheating and to give the patient the right dose), and update the status display window. For the control program for this machine, would you use multiple threads, multiple processes, a single non-threaded process, or something else? Why?