CSPB 3753 - Fall 2024 - Knox - Operating Systems

Started on	Friday, 27 September 2024, 4:39 PM
State	Finished
Completed on	Friday, 27 September 2024, 6:19 PM
Time taken	1 hour 40 mins
Grade	**84** out of 100

Question **1**

Partially correct

Mark 7 out of 10

Term Definitions:

1. an environment that enables multiple virtual machines to run securely while sharing hardware

2. method used by the system to change from one process to another

3. OS software that runs inside a virtual machine

4. a set of processes available and waiting to execute

5. a unique number assigned to each process

6. variable available to a process and allocated on the heap

7. one way communication flow between two programs

8. technologies designed to protect each process from other processes in the operating system

9. two way communication flow between two programs

10. a data structure used to manage information about a process

11. system keeps several jobs in memory simultaneously switching between them to keep CPU utilized

12. variable declared inside a specific function and allocated on the stack

13. system call that loads a new program, replacing the previous executable

14. system call that creates a copy of the current process

15. switching among processes so frequently that it behaves as if processes are simultaneous

16. set of kernel routines for handling function requests to the OS

17. a set of processes that needs some operation to complete in order to continue its execution

18. mechanism for moving a process temporarily out and back into main memory

19. OS software that runs the virtual machine

20. method of transferring program control (branching) to another part of a program such as the OS handler routines

Place the number of the best definition next to each term:

exec()	13 ✔	process isolation	8 ✔
multi-programming	15 ✖	guest OS	3 ✔
context switch	20 ✖	waiting queue	4 ✖
socket	9 ✔	process identifier	5 ✔
system call API	16 ✔	dynamically allocated variables	6 ✔

Your answer is partially correct.

7 of your answers are correct.

Question **2**

Correct

Mark 3 out of 3

How does the hardware differentiate between execution of the user and kernel code?

Select one:

- a. uses blocking I/O to switch to kernel code
- b. uses asynchronous calls to switch to user code
- c. uses a mode bit to indicate the type of code ✔
- d. uses controllers and DMA to support kernel

Your answer is correct.

Question **3**

Correct

Mark 3 out of 3

What is a context switch?

Select one:

- a. moving the process from one device to another
- b. interrupting the CPU to process a signal
- c. moving the mode bit from 0 to 1
- d. switching the active PCB from one process to another ✔
- e. switching the PC register to another address

Your answer is correct.

Question **4**

Correct

Mark 4 out of 4

Global variables, Dynamically allocated variables, and Local variables are stored in the _____ blocks of memory respectively. (match them up)

Select one:

- a. Heap, Data, and Stack
- b. Data, Heap, and Stack ✔
- c. Stack, Heap, and Data
- d. Data, Stack, and Heap
- e. Heap, Stack, and Data
- f. Stack, Data, and Heap

Your answer is correct.

Question **5**

Correct

Mark 2 out of 2

Mode bit == 0 indicates:

Select one:

- a. User mode
- b. kernel mode ✔
- c. None of the other
- d. processor mode

Question **6**

Correct

Mark 4 out of 4

Which of the following are process states?

Select one or more:

- [ ] a. preempted
- [x] b. running ✔
- [ ] c. queued
- [x] d. ready ✔
- [x] e. blocked ✔
- [ ] f. cooperating

Your answer is correct.

Question **7**

Complete

Mark 3 out of 6

Describe **two** of the four OS components that comprise a monolithic kernel.
Please be succinct and provide no more than two paragraphs per component.

System Call Interface

The system call interface for a monolithic kernel is the way in which a program or application communicates (interacts) with the kernel. This is like a psuedo 'pipe' (physical pipe) that allows user level programs to be able to request services at the kernel level. There are a number of kernel level system calls that can be called from the user level such as read() or fork() that serve as an example of making said request.

File System Management

This is the mechanism in which entities like directories and files are managed within an OS. This mechanism also manages file permisssions (like being able to execute a bash script) as well as other information that pertains to the machine and the OS. The file system also controls what types of files are supported in the OS. A Unix machine for example cannot execute a Windows executable file (.exe) etc.

The OS components of a monolithic kernel are Memory Management, Scheduler, File System, and Device Drivers.

Comment: System Call Interface - is not a component of the OS. See lectures and reading from Week 1 to review.

Question **8**

Partially correct

Mark 3 out of 5

Which of the following are benefits of using a Multi-Tasking?

Select one or more:

- [x] a. supports interactive applications ✔
- [x] b. fault isolation ✔
- [x] c. forces the CPU to not be idle ✖ CPU can still be idle if all processes are in the wait queue
- [x] d. efficient use of CPU ✔
- [ ] e. makes applications cooperative

Your answer is partially correct.

You have selected too many options.

Question **9**

Correct

Mark 7 out of 7

The process execution pattern for a process that runs for 10 ticks, pauses for IO for 20 ticks, and finishes executing its final 30 ticks would have a pattern

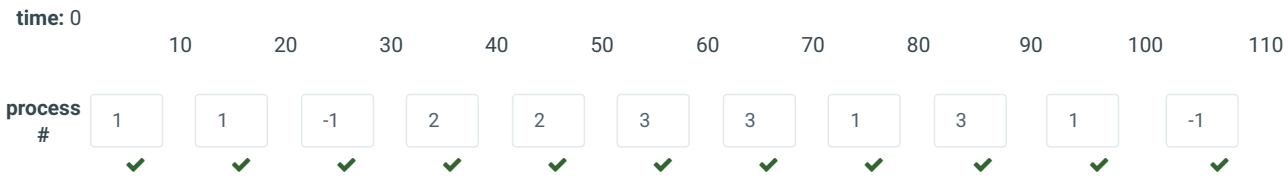
10 <20> 30

listed in the last column.

process	arrival time	execution time	IO blocking pattern
P1	0	40	20 <40> 10 <10> 10
P2	30	20	20
P3	35	30	20 <10> 10

Please place a process **number (1,2,3)** or -1 for "idle" into each time slot based upon the given data.

Assume *multitasking scheduling* (like the Unit 1 Exam review question).



Your answer is correct.

Question **10**

Correct

Mark 2 out of 2

What is a distributed system?

Select one:

- a. collection of loosely coupled nodes interconnected by a communications network ✔
- b. set of CPUs and memory all controlled by one operating system
- c. distribution of the data between a number of file systems
- d. migration of processes between CPUs

Your answer is correct.

Question **11**

Correct

Mark 2 out of 2

Match the Distributed System Components with their descriptions:

service	set of operations provided by the system ✔
server	machine supplying a set of operations ✔
client	machine using a set of operations ✔

Your answer is correct.

Question **12**

Correct

Mark 2 out of 2

What is the name of moving a running VM from one machine to another?

Select one:

- a. live migration ✔
- b. pseudo virtualization
- c. hypervisor guest services
- d. suspend, clone, and resume (SCR)

Your answer is correct.

Question **13**

Incorrect

Mark 0 out of 2

Which of the following is NOT a benefit of using a VM?

Select one:

- a. to suspend, clone, and resume a complete virtual environment ✖
- b. running multiple OS on single machine
- c. virtual machines make it easier to share physical resources
- d. fault or corruption isolation

Your answer is incorrect.

Question **14**

Correct

Mark 2 out of 2

The _____ is running within a virtual machine on the _____ .

Select one:

- a. remote OS, hypervisor
- b. host OS, guest OS
- c. guest OS, host OS ✔
- d. hypervisor, remote OS

Your answer is correct.

Question **15**

Correct

Mark 2 out of 2

What is used when there are multiple paths for sending messages between machines?

Select one:

- a. recurring
- b. resolving
- c. routing ✔
- d. recombining

Your answer is correct.

Question **16**

Correct

Mark 2 out of 2

What is the name for the 7 layer communications model?

Select one:

- a. OSI protocol stack ✔
- b. user datagram protocol
- c. networking protocol
- d. original model standard

Your answer is correct.

Question **17**

Correct

Mark 2 out of 2

What type of network is WiFi?

Select one:

- a. NFS
- b. LAN ✔
- c. SAN
- d. WAN

Your answer is correct.

Question **18**

Incorrect

Mark 0 out of 2

Which of the following are examples of computation migration?

Select one or more:

- [ ] a. using shared memory to communicate between processes
- [ ] b. message passing between client and server
- [x] c. making the data local for process using it ✖
- [ ] d. remote procedure call (RPC)

Your answer is incorrect.

Question **19**

Complete

Mark 2 out of 4

Describe **two** major benefits of Virtualization.
Please be succinct and provide no more than two paragraphs per benefit.

Benefit 1

The first and biggest benefit (in my eyes) is that virtualization allows you to run multiple OSs on a single host OS. This means if I have a computer that normally runs Windows, I can emulate a Linux environment if I use virtualization. This is a huge benefit because it allows developers the ability to compile and build programs and multiple OSs without the necessity to buy more hardware. Huge for developing cross platform applications!

Benefit 2

The second benefit for virtualization is that it allows one to utilize the hardware of one machine for multiple 'virtual' machines. This means if you have one computer with a lot of resources, you have the capability of essentially running multiple OSs (that each represent their own machines) where different programs can run on different guest OSs doing different things. One program on one OS can be creating data, one program can be analyzing it on a different OS, etc. This allows for lots of positives when it comes to productivity and feature implementation to applications and programs in development.

Comment: Your second benefit seems to be the same as the first, Multiple Guest OS on single host OS.

Question **20**

Correct

Mark 2 out of 2

What is the name of the OS data structure holding information about individual tasks?

Select one:

- a. controller registers
- b. process control block ✔
- c. context switch
- d. job queue

Your answer is correct.

Question **21**

Correct

Mark 2 out of 2

What is a PID?

Select one:

- a. process information descriptor
- b. unique process identifier ✔
- c. parent identifier
- d. constant value defined in the kernel

Your answer is correct.

Question **22**

Correct

Mark 2 out of 2

Which of the following are part of a task's in memory image?

Select one or more:

- [ ] a. disk controller
- [ ] b. source code
- [x] c. stack ✔
- [ ] d. ready queue
- [x] e. data ✔
- [x] f. heap ✔
- [x] g. code ✔

Your answer is correct.

Question **23**

Correct

Mark 6 out of 6

Which of the following are TRUE for Linux process calls for fork() and exec()?

Select one or more:

- [x] a. exec() replaces the current processes code ✔
- [ ] b. exec() creates a new process to run
- [ ] c. fork() cannot be called after fork()
- [x] d. fork() returns child process id to parent ✔
- [x] e. fork() starts the process from the statement following the command ✔
- [ ] f. fork() returns parent process id to the child
- [x] g. fork() creates a new copy of the current process ✔

Your answer is correct.

Question **24**

Correct

Mark 2 out of 2

In UNIX, what is the *name* of the system call used for synchronization between parent and child processes that have exited? (Just enter the function name -- no parameters or other syntax).

Answer: wait ✔

Question **25**

Partially correct

Mark 2 out of 4

In UNIX, if a parent process exits before waiting on a child process, then the child enters what process state? Your answer should be one word.

Answer: orphan ☑

"Orphan" generally refers to files or processes that have lost their connections, while a "zombie" specifically describes a type of process that has completed its execution but has not yet been fully terminated and cleaned up in the system.

Question **26**

Correct

Mark 2 out of 2

Shared memory is faster than using pipes or sockets for doing IPC.

Select one:

- (•) True ✔
- ( ) False

Question **27**

Correct

Mark 8 out of 8

Using only fork() statements (no if, while, or for statements) write code that prints "Hi!" to `stdout` 4 times.

Answer: (penalty regime: 10, 20, ... %)

Reset answer

Ace editor not ready. Perhaps reload page?
Falling back to raw text area.

```
#include <stdio.h>
#include <sys/types.h>
#include <unistd.h>
int main(){

    /*
        using only fork() statements (no if, while, or for statements)
        write code that prints "Hi!" to stdout 4 times.
    */

    fork(); // Two processes
    fork(); // Four processes

    sleep(1); // don't delete me!
    printf("Hi!\n"); // don't delete me!

    return 0;
}
```

	Test	Input	Expected	Got	
✔	null	null	Hi! Hi! Hi! Hi!	Hi! Hi! Hi! Hi!	✔

Passed all tests! ✔

Correct

Marks for this submission: 8/8.

Question **28**

Correct

Mark 5 out of 5

Complete the given partial program so that it performs bidirectional IPC between the parent and child process using pipes. In particular, complete the code below so that `msg1` is sent from the parent process to the child process, and `msg2` is sent from the child process to the parent process. (This question is auto-graded, but partial credit will be assigned by later by hand.)

Answer: (penalty regime: 10, 20, ... %)

Reset answer

Ace editor not ready. Perhaps reload page?
Falling back to raw text area.

```
#define  _GNU_SOURCE
#include <stdio.h>
#include <stdlib.h>
#include <sys/types.h>
#include <sys/wait.h>
#include <unistd.h>
#include <string.h>

int main(int argc, char* argv[]){

    char buf;
    size_t len = 0;
    char *msg1 = NULL; // the string to be sent from the parent to the child
    char *msg2 = NULL; // the string to be sent from the child to the parent

    getline(&msg1, &len, stdin); // read from standard in
    len = 0;
    getline(&msg2, &len, stdin); // read from standard in
```

	Test	Input	Expected	Got	
✔	`going_to_child going_to_parent`	`going_to_child` `going_to_parent`	`going_to_child` `going_to_parent`	`going_to_child` `going_to_parent`	✔

Passed all tests! ✔

Correct

Marks for this submission: 5/5.

Question **29**

Complete

Mark 1 out of 1

Write some comments about the exam or write the justification on your answers to some difficult questions (may allow for partial credit).
You should NOT write comments for every question, just a few that you felt were the **most** difficult for you.
You should **write "no comments"** in the answer if you have none.

I believe my answer about the monolithic kernel is going to be pretty short because I don't really have many other ways to describe the components than the way that I did. So I left those shorter than what you may want for length.

The same kind of goes for the virtualization question. I didn't really want to leave an essay answer for either of the short responses because I think short and succinct answers are generally more palatable.

I remember the first programming question (Problem 27) from Channheum's office hours in systems. When I first looked at it I freaked out but I remembered the power of two thing with forks and it was pretty simple. The other programming question (Problem 28) seemed just like a short abstraction from the practice exam and after how many times I did the practice exam I felt pretty prepared. I'm glad I spent Wednesday and Thursday night studying for the exam (especially pipes after my interview).

Thank you for making the window for this exam 24 hours, it really helps with me and my work schedule. See you in office hours soon!

Comment: