Preview Test: Homework 1

Test Information

Description	This homework is intended to test your prerequisites.
Instructions	Please work on this homework independently. Please answer the multiple choice questions below. If more than one answer appears to be applicable, please indicate the best answer. You have only one attempt at this homework. Please do not submit the homework until you are done. You may save the homework and come back later, but once you submit, you will no longer be able to change your answers. The homework is due Saturday, Feb 6th, by 11pm.
Multiple Attempts	Not allowed. This test can only be taken once.
Force Completion	This test can be saved and resumed later.

Question Completion Status:

Save All Answers | Close Window | **Save and Submit**

QUESTION 1 — 1 points — Save Answer

The **fork** system call

a. Creates a new child thread and gives it an exact copy of the current process's address space.

b. Creates a new child process and gives it an exact copy of the current process's address space.

c. Runs a new program in a child process.

d. Provides an exact copy of the current process's address space to an existing child process.

QUESTION 2 — 1 points — Save Answer

Which of the following systems may never exhibit a "hold-and-wait" situation? (Note: "hold and wait" is one of the preconditions of deadlock.) You may assume that the only blocking that occurs in these systems occurs on mutexes.

- a. Systems that ensure that all resources needed for an application are locked in a single atomic operation that either succeeds and locks all requested resources or fails and locks none.
- b. Systems with only one mutex.
- c. Systems where all mutexes are numbered. A user cannot lock a mutex with a lower number, X, after they have locked a mutex with a larger number, Y > X.
- d. Systems of type (a) and (b) only
- e. Systems of type (a), (b), or (c)

QUESTION 3 1 points Save Answer

Consider the following scheduling policies: (i) First In, First Out (FIFO), (ii) Round Robin with a small quantum, (iii): Round Robin with a large quantum, (iv): Preemptive Shortest Job First (SJF), (v): Preemptive Priority Scheduling (PPRI). Which of these policies can lead to starvation of processes?

- a. (i), (iii), and (iv) only
- b. (iv) and (v) only
- c. (i) and (iii) only
- d. (iii), (iv), and (v) only
- e. (i), (ii), (iii), (iv), and (v)

QUESTION 4 1 points Save Answer

Which scheduling algorithm gives the smallest average wait time?

- a. Priority
- b. First come, first served
- c. Round robin
- d. Shortest job first

QUESTION 5 1 points Save Answer

A difference between **system calls and function calls** is that:

- a. Only function calls preserve the stack frame; system calls do not.
- b. Function calls are within the same address space, but system calls are not.
- c. Function calls require a change of priviledge level, but system calls do not.
- d. System calls can pass arguments on the stack; function calls cannot.

QUESTION 6 1 points Save Answer

Which of the following lock request sequences results in a deadlock? Assume that no locks are released in each of the sequences below. Assume no resources were locked at the beginning of each sequence. Assume that a request for a lock succeeds in acquiring the lock if it has not already been acquired by a previous request in the sequence. A request for locked resource blocks. Each sequence is a different case, independent from the other sequences. Processes are denoted by letters A, B, C, … and locks are denoted by letters R1, R2, R3, …

- a. A requests R1. B requests R1. C requests R1
- b. A requests R1. B requests R2. C requests R3. D requests R1
- c. A requests R1. B requests R2. A requests R2. C requests R3. B requests R1.
- d. A requests R1. A requests R2. B requests R2. C requests R1

QUESTION 7 1 points Save Answer

Threads belonging to the same process **share**

- a. Stack
- b. Thread ID
- c. Register set
- d. Heap

QUESTION 8 1 points Save Answer

POSIX is …

a. An operating system that implements a standard set of system calls and system behaviors.

b. A specification for *how* an operating system must implement a standard set of system calls and system behaviors.

c. A specification for a programming language and library that provide a standard set of system calls and system behaviors.

d. A specification for a standard set of system calls and system behaviors that can be implemented in different ways by operating systems.

QUESTION 9 1 points Save Answer

Which of the following commands sends a signal to a process?

a. kill()

b. sigsetmask()

c. signal()

d. send()

QUESTION 10 1 points Save Answer

What is the purpose of signals in Linux?

a. Signals are synchronization primitives that enforce mutual exclusion among processes

b. Signals are asynchronous notifications sent to a process in order to inform it of an event that occurred

c. Signals are data structures used for communication over a network.

d. Signals are only used for a program to notify the operating system that it is terminating and that the operating system may now clean up the program's memory.

QUESTION 11 1 points Save Answer

Which of the following code snippets are wrong?

```
Case 1              Case 2              Case 3
int *p;             char a[2];          int b[10];
*p=10;              strcpy (a, "Hi");   *b=11;
```

- a. Case 1
- b. Case 2
- c. Case 3
- d. Case 1 and 2
- e. Case 1 and 3

QUESTION 12 1 points Save Answer

A process enters the ready state when

- a. its quantum expires
- b. it reaches the top of the run queue
- c. it is unblocked
- d. it tries to access a locked semaphore
- e. either (a) or (c)

QUESTION 13 1 points Save Answer

In the **POSIX** standard, the call waitpid() does which of the following?

- a. Blocks on a semaphore until a signal call is executed
- b. Waits for a process with a specified process ID to terminate
- c. Waits for a child with a specified process ID to terminate
- d. Waits until any of the children of the calling parent terminates
- e. Blocks a process with the specified process ID

QUESTION 14 1 points Save Answer

Which of the following best describes a UNIX shell?

- a. Part of the UNIX kernel that executes user commands
- b. A process forked off at UNIX initialization to accept inputs from a user
- c. A system call executed by the UNIX startup routine to accept commands from users
- d. A library that implements various UNIX commands
- e. The UNIX keyboard device driver that interprets keyboard input

QUESTION 15 1 points Save Answer

Which of the following are **not** shared among threads within the same address space?

- a. The stack
- b. A single address space
- c. Open file descriptors
- d. A process ID
- e. Open sockets

QUESTION 16 1 points Save Answer

If process A, with variable X=1, executes fork() creating process B, then both A and B execute X=X+1, which of the following is true?

- a. The value of X becomes 2
- b. The value of X becomes 3
- c. The value of X may become either 2 or 3 depending on the timing of execution
- d. The value of X in process A becomes 2 and in process B becomes 3 if A executes X=X+1 first
- e. None of the above

QUESTION 17 1 points Save Answer

Which of the following scheduling policies is more likely to achieve better fairness?

- a. Priority scheduling
- b. Earliest deadline first
- c. Shortest job first
- d. Round robin
- e. Preemptive earliest deadline first

QUESTION 18 1 points Save Answer

Which of the following state transitions is not possible?

- a. Blocked to running
- b. Ready to running
- c. Blocked to ready
- d. Running to blocked
- e. Running to ready

QUESTION 19 1 points Save Answer

Thrashing is said to occur when:

- a. The OS runs out of memory resources and cannot start up more processes.
- b. The scheduler implements a round robin scheduling policy with a small quantum size
- c. A program uses physical memory pages not allocated to it, eventually leading to a segmentation fault.
- d. A program's working set requires more memory than the physical pages available for it

QUESTION 20 1 points Save Answer

Which of the following is true when a child process terminates and its parent never waits for it?

- a. The parent becomes a zombie
- b. The child becomes a zombie
- c. The child becomes an orphan
- d. The parent crashes
- e. None of the above problems occur

Click Save and Submit to save and submit. Click Save All Answers to save all answers.

Save All Answers | Close Window | **Save and Submit**