

Homework 1

Due Friday, 02/14/2020 @ 11:59pm

Part I. Multiple-Choice Questions: Choose the answer that best applies. (2 points each)

Q1. A _____ uses an existing thread — rather than creating a new one — to complete a task.

- A) lightweight process
- B) thread pool
- C) scheduler activation
- D) asynchronous procedure call

Ans: **B**

Q2. How many arguments does the fork system call take?

- A) 0
- B) 1
- C) 2
- D) 3

Ans: **A**

Q3. _____ involves distributing tasks across multiple computing cores.

- A) Concurrency
- B) Task parallelism
- C) Data parallelism
- D) Parallelism

Ans: **B**

Q4. In what way is an operating system like a government?

- A) It seldom functions correctly.
- B) It creates an environment within which other programs can do useful work.
- C) It performs most useful functions by itself.
- D) It is always concerned primarily with the individual's needs.

Ans: **B**

Q5. What are some other terms for kernel mode?

- A) supervisor mode
- B) monitor mode
- C) privileged mode
- D) All of the above

Ans: **D**

Q6. ____ is the number of processes that are completed per time unit.

- A) CPU utilization
- B) Response time
- C) Turnaround time
- D) Throughput

Ans: **D**

Q7. What statement concerning privileged instructions is considered false?

- A) They may cause harm to the system.
- B) They can only be executed in kernel mode.
- C) They cannot be attempted from user mode.
- D) They change processor modes from kernel to user

Ans: **C**

Q8. Which symbol denotes a pipe in Linux?

- A) >
- B) <
- C) ^
- D) |

Ans: **D**

Q9. System calls are said to be expensive because?

- A) they require transition between the user and kernel modes
- B) they include library calls
- C) computer centers charge a fee for each system call
- D) None of the above

Ans: **A**

Q10. The _____ model maps many user-level threads to one kernel thread.

- A) many-to-many
- B) two-level
- C) one-to-one
- D) many-to-one

Ans: **D**

Part II. True or False (2 points each)

Q11. A system call is triggered by hardware. (F)

Q12. There is no universally accepted definition of an operating system. (T)

Q13. In RR scheduling, the time quantum should be small with respect to the context-switch time. (F)

Q14. The exec() system call creates a new process. (F)

Q15. A program and a process are one and the same (F)

Q16. System programs run in kernel space. (F)

Part III. Fill in the appropriate word, phrase or value in the space provided (1 point each)

Q17. There are two types of parallelism: data parallelism and task parallelism.

Q18. Many criteria have been suggested to evaluate and compare CPU-scheduling algorithms. The criteria include CPU utilization, Throughput, Turnaround time, Waiting time, and Response time.

Q19. In FCFS Scheduling, FCFS stands for First Come First Served.

Q20. Cooperating processes require an interprocess communication (IPC) mechanism that will allow them to exchange data and information. There are two fundamental models of interprocess communication: shared memory and message passing.

Q21. Modern operating systems are interrupt driven.

Q22. Processes can be described as and divided into two categories, I/O-bound process and CPU-bound process.

Q23. Linux starts from an initial process with PID 1. It forks child processes, child processes fork child processes. This parent and child relation form a tree, called process tree.

Q24. Which Register in CPU holds the address of the next instruction to be executed?
Program counter

Q25. Process switch (process context switch) allocates the CPU from one process to another. It includes two portions: CPU context and Storage context.

Part IV. Answer all of the following questions.

Q26. Time sharing and multiprogramming are two concepts that are sometimes confused by people who haven't taken CS3110. Describe the difference. (4 points)

Ans: Multiprogramming refers to multiple jobs or processes (2 points), whereas time sharing refers to multiple users (2 points) interacting with a system.

Q27. List 5 types of system resource. (5 points)

Ans: the processor (CPU), memory, I/O devices, File storage, network connections

Q28. What is confidentiality? (2 points)

Ans: It assures that confidential or private information is not made available or disclosed to unauthorized individuals (2 points). In other words, only authorized individuals are able to gain access to or view sensitive data.

Q29. Explain the difference between singly, doubly, and circularly linked lists. (6 points)

Ans: A singly linked list is where each item points to its successor. (2 points)

A doubly linked list allows an item to point to its predecessor or successor. (2 points)

A circularly linked list is the where the last element points back to the first. (2 points)

Q30. In Unix systems, what system calls have to be executed by a command interpreter or shell in order to start a new process? What do these system calls actually do? (7 points)

Ans: In Unix systems, a fork (1 point) system call followed (1 point) by an exec (1 point) system call need to be performed to start a new process.

The fork call clones the currently executing process (2 points), while the exec call replaces the calling process or the process that calls it (2 points).

Q31. Explain the difference between preemptive and nonpreemptive scheduling. (4 points)

Ans: Preemptive scheduling allows a process to be interrupted in the midst of its execution, taking the CPU away and allocating it to another process. (2 points)

Nonpreemptive scheduling ensures that a process relinquishes control of the CPU only when it finishes with its current CPU burst or due to I/O. (2 points)

Q32. Consider the following C code that calls fork(). If you assume that the child process is always scheduled before the parent process, what will be the output? (6 points)

```
int main()
{
    int i;
    for (i = 0; i < 3; i++) {
        if (fork() == 0) {
            printf("Child sees i = %d\n", i);
            exit(1);
        } else {
            printf("Parent sees i = %d\n", i);
        }
    }
}
```

```
    }  
  }  
}
```

Ans:

Child sees $i = 0$

Parent sees $i = 0$

Child sees $i = 1$

Parent sees $i = 1$

Child sees $i = 2$

Parent sees $i = 2$

Q33. Consider the following program. How many times hello is printed? _____ (3 points)

```
#include <stdio.h>  
#include <sys/types.h>  
int main()  
{  
    fork();  
    fork();  
    fork();  
    printf("hello\n");  
    return 0;  
}
```

Ans: 8

Q34. What is orphan process? (2 points)

Ans: An orphan process is a process that is still executing (1 point), but whose parent has died (1 point).

Q35. Describe the similarities and differences of doing a context switch between two processes as compared to doing a context switch between two threads in the same process. (4 points)

Ans: The process context switch needs to save and restore all of the process state, including both CPU context and Storage context. (2 points)

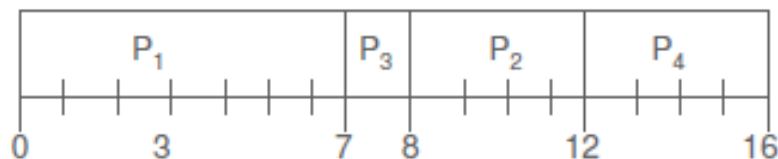
The thread context switch needs to save the program counter and registers or CPU context. The memory and other resource information stays the same. So they don't have to be saved during the thread context switch. (2 points)

Q36. Suppose that the following processes arrive for execution at the times indicated. Each process will run for the amount of time listed.

<u>Process</u>	<u>Arrival Time</u>	<u>Burst Time</u>
P_1	0.0	7
P_2	2.0	4
P_3	4.0	1
P_4	5.0	4

- a) What is the average waiting time for these processes with the SJF scheduling algorithm (or Non-Preemptive SJF)? (3 points)

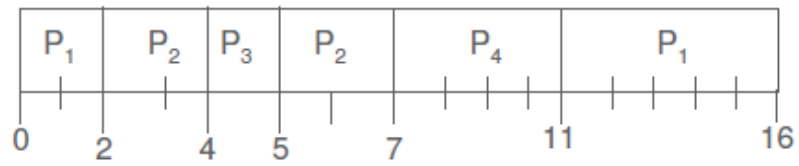
SJF (non-preemptive)



$$\text{Average waiting time} = (0 + 6 + 3 + 7)/4 = 4$$

- b) What is the average waiting time for these processes with the Shortest-Remaining-Time-First scheduling algorithm (or Preemptive SJF)? (3 points)

SJF (preemptive)



$$\text{Average waiting time} = (9 + 1 + 0 + 2)/4 = 3$$