TODO - get past quizzes from Dr. Leverington and import the questions here.

***Quiz #1, 2, & Appendix A***

**Explain how the OS manages the system CPU, including how the use of the CPU looks different between the user experience and the system management.**

OS would manage the resources for programmers, which is like managing the time of CPU for programs and the order of which to be first computed.

User: When a user manages an input, there would be an output from CPU and be posted to user.

System management: the programs would be ordered and managed about when to be computed and how long it would take in CPU

**Identify and briefly describe three services that are provided by the OS to user applications.**

1. File management. Including create, delete, modify files.

2. State information. Some programs may want to know about the state of computer, including time, date, and etc.

3. File modificatio. Let text editor be available to create and modify the content of files stored in disk.

4. Communication. These programs provide the mechanism for creating virtual connections among processes, users and computer systems.

**Identify and briefly describe three ways to pass data to a system call**

1. Pass data through an API system call as a parameter.
2. Pass data into a system register.
3. Pass data through system messages.

**For purposes of OS design, explain the difference between policy and mechanism**

A policy is what something will do; a mechanism is how it will be done. The main difference between policy and mechanism is that policy describes the “what”, while mechanism describes the implementation specific details of “how”.

Policy is about what to do while mechanism is about how to do.

***Quiz #3***

**Explain what happens during a context switch.**

During a context switch, the kernel saves the state of the currently running process and performs a state-restore on the process to run next. While this is the general workflow that is employed during a context switch, it’s important to note the OS is operating in the background throughout these stages. Before a state-restore occurs, and throughout the context-switch, the OS may determine what process is to be run next or perform other management functions.

**Explain what a PCB is and how it is used by the OS.**

The PCB or process control block is a kernel data structure created for each process. A PCB is also known as a task control block. The PCB contains information about the process that is useful for the OS in scheduling and resource management. The PCB serves as a repository for all the data needed to start, or restart, a process, along with some accounting data.

The OS interacts with the PCB in order to store the current state of a process. The set of all PCBs at any given moment is a view of the OS’s current state.

**What condition would most likely cause a process to be held up in the new state?**

System is busy. Ready is full.

**What condition would most likely cause a process to be held up in the ready state?**

Another process is currently running in the CPU.

**What action is most likely to cause a transition from waiting to ready?**

An I/O events occur or interrupts. The process is going to be allocated to somewhere else.

**What condition is most likely to cause a transition from running to waiting?**

The current process is in need of an I/O operation or memory request.

**What condition is most likely to cause a transition directly back to ready from running?**

The current process experiences an interrupt.

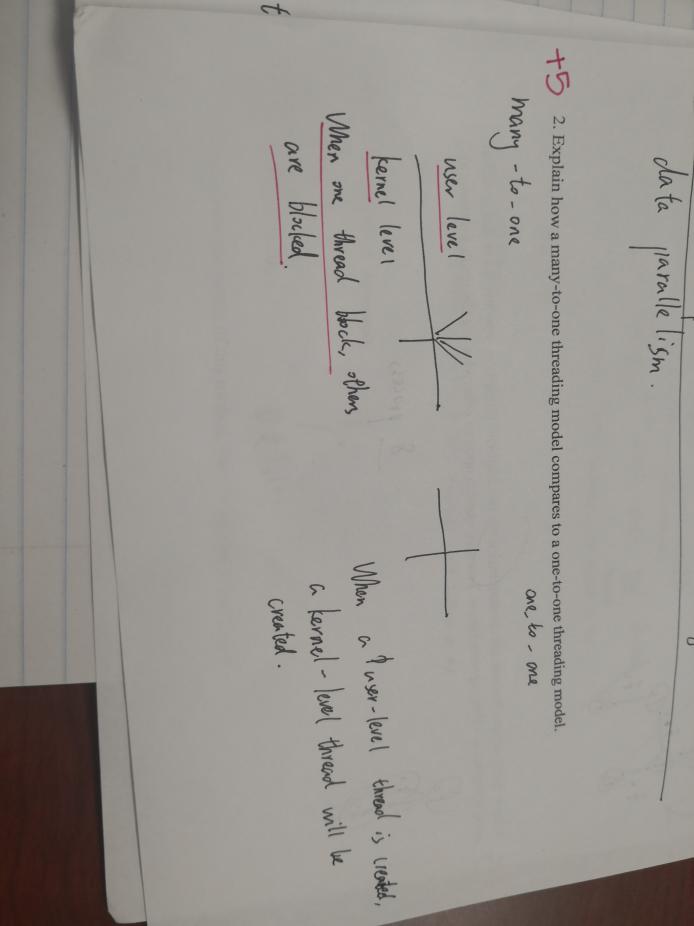
***Quiz #4***

**Explain what task parallelism is, and provide an example that used task parallelism.**

It focuses on distributing tasks to different cores, sharing the same data.

Example : When doing the recursive, do part 1 first then do part 2 and so on. Finally combine them all.

**Explain how a many-to-one threading model compares to a one-to-one threading model**



**Explain the difference between a kernel-level thread and a user-level thread; provide an example.**

- User-level threads are unknown by the kernel, whereas the kernel is aware of kernel threads.

- On systems using either many-to-one or many-to-many model mapping, user threads are scheduled by the thread library, and the kernel schedules kernel threads

1. User-level threads are unknown by the kernel, whereas the kernel is aware of kernel threads.
2. On systems using either M:1 or M:N mapping, user threads are scheduled by the thread library and the kernel schedules kernel threads.

(3) Kernel threads need not be associated with a process where as every user thread belongs to a process

(b) when to use kernel thread better?

- Kernel threads need not be associated with a process, whereas every user thread belongs to a process.

- Kernel threads are generally more expensive to maintain than user threads, as they must be represented with a kernel data structure.

**#include <pthread.h>**

**#include <stdio.h>**

**int value = 0;**

**void \*runner(void \*param);**

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

**{**

**pid\_t pid;**

**pthread\_t tid;**

**pthread\_attr\_t attr;**

**pid = fork();**

**if (pid == 0)**

**{**

**pthread\_attr\_init(&attr);**

**pthread create(&tid,&attr,runner,NULL);**

**pthread\_join(tid,NULL);**

**printf("CHILD: value = %d",value); /\* LINE C \*/**

**}**

**else if (pid > 0)**

**{**

**wait(NULL);**

**printf("PARENT: value = %d",value); /\* LINE P \*/**

**}**

**}**

**void \*runner(void \*param)**

**{**

**value = 5;**

**pthread\_exit(0);**

**}**

On the next page, show the output of this method, identifying Line C and Line P

**Show the results of question #4 on this page**

Line C: CHILD: value = 5

Line P: PARENT: value = 0;

***Quiz #5***

**Explain how RR scheduling works, identify the benefit of using RR, and identify a problem with using RR.**

RR is similar with FCFS, it has a time slice which is a small time interval. First, we treat ready queue as FIFO queue, allocate each process time slice, if the process can be executed in a time slice, the process would release CPU, or the process’s CPU burst is longer than time slice, it would be interrupted to next after being executed for a time slice.

Benefit : It dramatically improve the response time. Just a time slice.

Problems : If the time is big enough, the first process would wait a really really long time to be executed. Starvation.

**With consideration for RR( Preemptive ) and FCFS ( Non-Preemptive ), answer the following questions:**

1. **Identify one commonality between RR and FCFS.**

Round Robin and FCFS both execute processes as they enter ( on a First-In-First-Serve) basis.

1. **Identify one difference between RR and FCFS**

An RR algorithm will only execute a process for a given quantum slice, whereas a FCFS algorithm will typically allow a process to finish before executing the next.

**Suppose that a CPU scheduling algorithm favors those processes that have used the least processor time in the recent past. Why will this algorithm favor I/O-bound programs and yet not permanently starve CPU-bound programs?**

Such a scheduling algorithm will favor I/O-bound programs because they will take a relatively short amount of cycles before blocking, therefore they will not hog the CPU and will not starve CPU-bound processes because of their short processing time.

**Quiz 9**

1.briefly explain pages, frames and page table, and the relationship between each of them

Pages : blocks of memory of logical memory

Frames : kind of page in physical memory

Page table : a table holds base address of each frame in physical memory.

Relationship : Frames create pages, page table stores frame, frames are pages in physical memory.

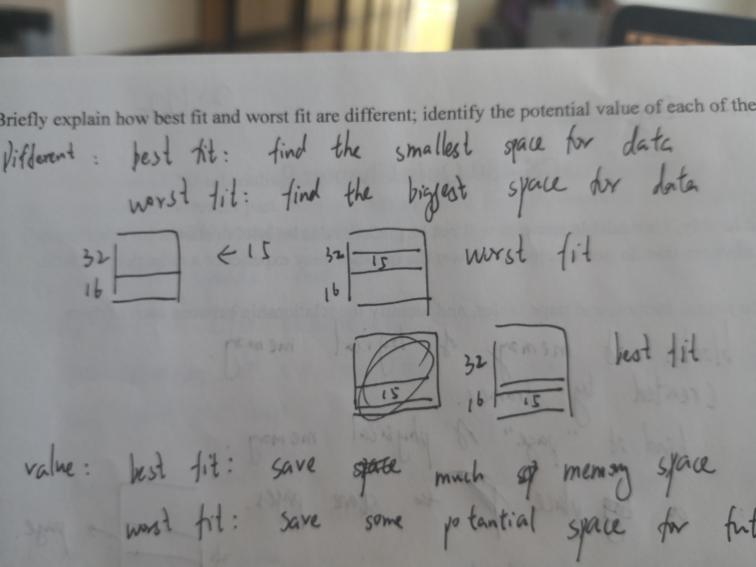
2.Explain how a page table is used to access memory

The page table would be given a logical address which is treated as an index in the page table. The index given to page table represents a base address in memory which is then used with a appropriate offset to access a frame in physical memory which holds some data.

3 Explain how best fit and worst fit are different, identify the potential value of each of them.

Difference : Best fit : find the smallest space for data

Worst fit : find the biggest space for data.



Value : best fit : Save much memory space

Worst fit : Save some potential space for future use

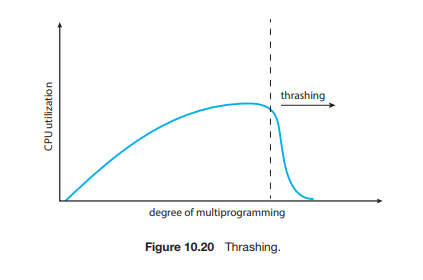
4 Explain the difference between a standard paging system and an inverted page system

Standard paging system : each process would have its own page table

Inverted page system : Several processes would share one page table

**Quiz 10**

**Analyze the plot at Figure 10.20 and explain system dynamics at specified points over the curve.**

****

a.  Briefly explain what is happening at Point A.

The system is working normally. As the degree of multiprogramming increases, the CPU utilization increases

At the beginning of the curve, multiprogramming is low and CPU utilization is low. This indicates that to best utilize the system CPU, a greater degree of multiprogramming is required.

b.  Briefly explain what is happening at Point B; note if this is a better condition than at Point A.

The system is still working normally, and is handling more multiprogramming that at Point A, thus it is a better situation

As multiprogramming is introduced into the system, we gradually witness an increase in CPU utilization due to multiple processes interacting with the CPU through context-switching and scheduling.

c.  Briefly explain what the OS might **incorrectly** conclude from the data represented by Point C.

Note, but not required in the answer: The system is thrashing as it is attempting to use resources that are not easily available.

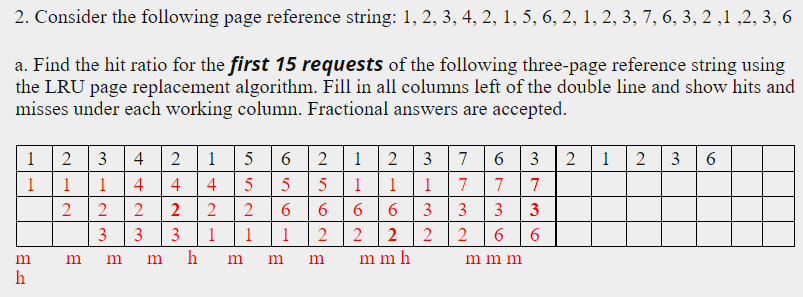
Answer: The system might incorrectly conclude that there are not enough processes going through the system since the CPU usage is down

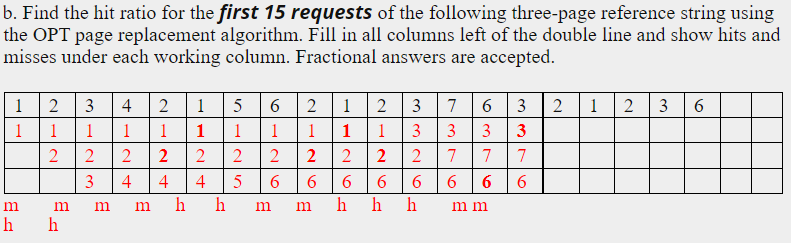
Once the degree of multiprogramming reaches close to the maximum of what the system can support, there is a plateauing of CPU utilization due heavy competition for CPU time.

d.  Briefly explain what could be happening with memory management at Point C.

It is taking too long to access available memory, so the system is thrashing.

After the degree of multiprogramming exceeds this maximum, thrashing starts to occur and CPU utilization drops significantly. To increase CPU utilization at this pointer, we must reduce the degree of multiprogramming to solve for thrashing.





**Chapter 6, 7**

1. Briefly explain the bounded buffer problem; describe some real-world condition that would represent this problem

You have two procedures for increasing and decreasing buffer sizes.When executed separately, the buffer count is generated correctly.However, they may not work properly when executed concurrently

For example, one process is creating information that will be used by another process. For example, suppose one process reads information from disk and the other process compiles this information from the source into machine code.

2.What are the three parts of a solution to the critical section problem, and briefly explain each

**Mutual exclusion**. If process *Pi* is executing in its critical section, then no other processes can be executing in their critical sections.

**Progress**. If no process is executing in its critical section and some processes wish to enter their critical sections, then only those processes that are not executing in their remainder s ections can participate in deciding which will enter its critical section next, and this selection cannot be postponed indefinitely.

**Bounded waiting**. There exists a bound, or limit, on the number of times that other processes are allowed to enter their critical sections after a process has made a request to enter its critical section and before that request is granted.

3.Briefly explain the Readers-Writers problem; describe some real-world condition that would represent this problem

▪ Suppose a database is Shared between several concurrent processes.

Some of these processes may only want to read the database, while others may want to update (read and write) the database.

▪ We distinguish between the former as a reader and the latter as a writer.

If two readers access the Shared data, there is no harm.

However, if the writer and other processes access the database at the same time, it can cause confusion.

Example: If several people can access a file.

Some people just read and access information, but if others write more information, the current information prepared by the reader may become outdated.

4.Briefly explain the Dining Philosophers problem; describe some other real-world condition that would represent this problem

▪ Five philosophers sit at a table and eat and think.

There is only one chopstick between each philosopher, and philosophers need two chopsticks left and right in order to eat.

If a philosopher has a chopstick, his neighbor cannot use it.

It demonstrates a potential deadlock scenario.

If all the philosophers decided to eat at the same time, holding on to the left chopstick, they would wait forever for the right chopstick to appear.

**Chapter 8**

1. Suppose that a system is in an unsafe state. Explain how it might or might not be possible for

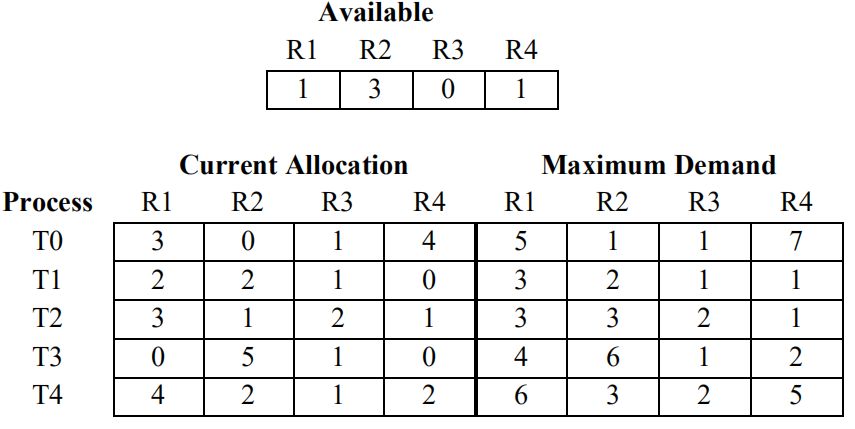
the threads to complete their execution without entering a deadlocked state.

a. Even though some threads might be used by a program, they may not all be used at the same time.

b. For this reason, a process may attempt to use threads that another one is not using at the moment and finish it’s tasks

2. Consider the following snapshot of a system. There are no outstanding unsatisfied requests for

resources.

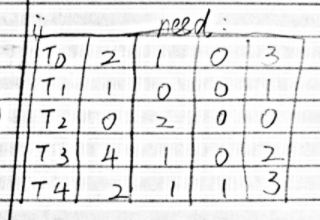


1. What reference data must be calculated to conduct this resource analysis?

the “needs resources” information must be calculated, the current allocation and maximum demand must to be calculated to conduct this resource analysis.

ii. Draw a table like the following to find the needed reference data. Note that more cells

are provided than required; use the table cells as needed to organize the data.



ii. Is this system currently in a safe or unsafe state? Show and/or explain how the

resources might be distributed in order (i.e., the order in which threads would be

processed) to arrive at either a safe or an unsafe state.

Either T0 or T1 can go, then the other can go, but no further threads can be processed

**Chapter 11,13,14,15**

1. explain how the SCAN hard drive scheduling algorithm is better than the SSTF scheduling system

Scanning may be fairer than SSTF because it will scan all requests, while SSTF may stay in a narrow area and lead to starvation.

1. briefly describe the striping process for a RAID system

Striping distributes a given block of data over multiple drives, so that if one drive fails, the other drives continue to store large amounts of data

1. identify and describe three types of OS operations that might occur with a file
2. Create a file.

Creating a file takes two steps. First, you must find space in the file system for the files. Second, you must create an entry in the directory for the new file.

2.Open a file.

Instead of having all file operations specify filenames (causing the operating system to calculate filenames, check access, and so on), all operations except for create and delete require file Open () first. On success, the open call returns a file handle that is used as a parameter in other calls.

3.Write a file.

To write to the file, we need to make a system call specifying the file handle to open and the information to write to the file.

If sequential, the system must hold a write pointer to the location in the file where the next write will take place.

The write pointer must be updated whenever a write operation occurs.

1. indexed allocation

Indexed allows for growth and change in a file where contiguous allocations are difficult to change

1. Briefly describe what VFS and NFS are, and explain how they might be used in a given computing system

VFS is the protocol and interface that defines how different systems communicate network information, where NFS is more relevant to a particular system on a local device.

NFS on a given system can support VFS operations between other systems

VFS can Provides a standard file operation interface for the application laye

**Chapter 16**

1. Briefly explain the difference between breach of confidentiality and breach of integrity.

A breach of confidentiality is a loss of protected or private information whereas a breach of integrity is an inappropriate modification of data

2. Briefly explain the difference between a man-in-the-middle attack and a session hijacking attack.

A man-in-the-middle attack involves someone intervening in communication between two nodes (people, devices, etc.) whereas a session hijacking attack is the interception of some communication. Session hijacking could lead to a man-in-the-middle attack.

3. Beside the letters below, identify four of the five security measure levels of protecting a system, and briefly describe them.

* Physical The site containing the computer system must be physically secured against armed or surreptitious entry by intruders
* Human Authorization, Careful human authorization must be taken to ensure that only the right users can access the system. Users can also be tricked into providing access
* **Operating system System must protect itself from accidental or purposeful security breaches**
* Network, Networks that intercept data networks can reveal private information; The interception of data may constitute a remote denial of service attack

4. Most grocery stores do not want to give away their merchandise. However, some people would rather not pay for the food they remove from the store.

a. Identify and briefly describe a secure by default strategy grocery stores would commonly use.

locked doors, customers have to pass through checkstands to get out, etc.

b. Identify and briefly describe a defense in depth strategy that grocery stores are likely to use.

first layer is above

second layer might be electronic tags that set off alarms

next layer might be cameras

third layer might be roving security folks

etc.

5. Explain whether you would use a risk assessment strategy or a penetration testing strategy to protect from a zero-day attack, and briefly explain your rationale.

* A zero-day attack is a software-related attack that exploits weaknesses that the vendor or developer is not aware of.

Risk Assessment Assess and assume your greatest weaknesses. So I choose risk assessment.

**Chapter 17:**

1. Define the concept of protection as pertains to a computer.

Protection as pertains to a computer involve the action(s) necessary to keep the computer/system from breaking.

* Computer security, also known as network security or IT security, is protecting information systems from theft or damage to hardware, software, and information, as well as from interrupting or misleading the services they provide.

1. Explain why a system might need to relinquish or revoke access rights, and provide an example.

To maintain principle of least privilege (or need to know access) an employee’s access may need to be modified as conditions change in her/his employment

Examples: An employee my have resigned or have been released The employee may not have a need for certain access any more. Other legitimate examples

* When the system is attacked, in order to protect the internal data information of the system, the access permission should be revoked or destroyed to protect the data
* Example: When a communications company is attacked, they revoke or destroy access rights to protect customer information

3. Explain the difference between a policy and a mechanism and provide an example.

Policy is the desired goal or outcome of a given process while mechanism is how the policy is executed

* For example, a first-come-first-served policy can be used to grant resources to a process

4. Explain the meaning of using a sandbox in computers, and provide an example.

The concept of using a sandbox is to keep the contents (e.g., sand or viruses) inside a contained system.

Example: Classroom computer system that has not network access to the Internet or other parts of a given network

* In computer security, a sandbox is a security mechanism that separates running programs, usually to reduce the spread of system failures and/or software vulnerabilities.
* Example The sandbox can redirect the software operation files, registry and other paths in the sandbox to other locations (the location specified by the sandbox), so that the resources intended to be operated by the software will not be accessed or operated, ensuring the security of the resources.

5. Provide an example of how you use the principle of least privilege in your own life and explain how this provides protection

My house is locked, I don't allow other students to use my student ID, I only allow my closest friends on Facebook to access my personal information and so on. My house is locked then others can not get into my house which protect me and my assets