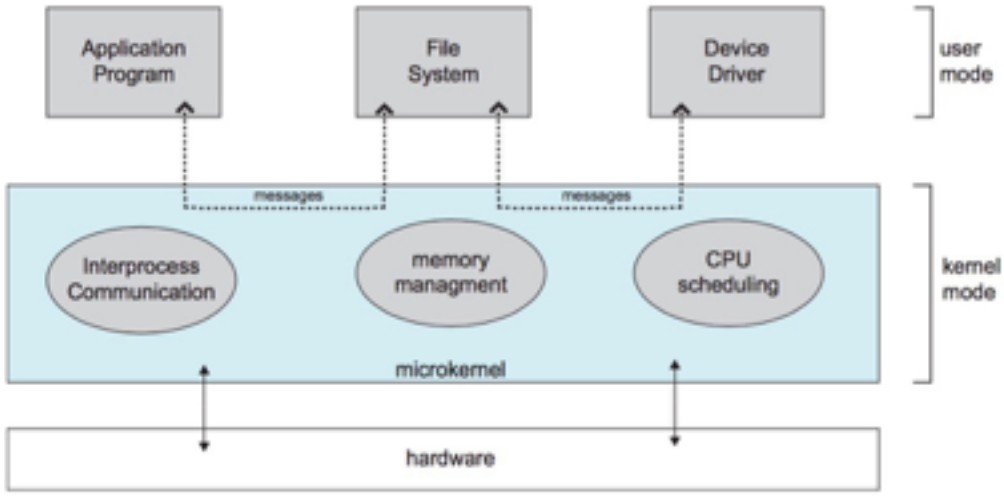
OS Assignment I

1.

A. What is the main advantage of the microkernel approach to system design? How do user programs and system services interact in a microkernel architecture? What are the disadvantages of using the microkernel approach?

Microkernel approach structures the operating system by removing all nonessential components from the kernel and implementing them as system and user-level programs.

The result is a smaller kernel. Microkernels provide minimal process and memory management, in addition to a communication facility.



Architecture of a typical microkernel.

User programs and services interact via message passing in a microkernel architecture. The client program and service never interact directly. Rather, they communicate indirectly by exchanging messages with the microkernel.

Some disadvantages of microkernel approach include:

- System calls can require a lot of protection mode changes.
- OS personalities are easier to port to new hardware after porting to microkernel, but porting to microkernel may be harder than porting to new hardware.

B. Which of the following instructions should be privileged? State Yes or No and justify your answer.

a. Disable all interrupts

YES, so that a process cannot dominate over the CPU.

b. Read Program Status Word (register which tracks current state of OS)

NO, every process should be able to read the PSW.

c. Clearing virtual memory

NO, because this only involves the process calling it.

d. Switch from user to kernel mode

NO, its how applications invoke system calls.

2.

A. Give two reasons why caches are useful. What problems do they solve? What problems do they cause? If a cache can be made as large as the device for which it is caching (for instance, a cache as large as a disk), why not make it that large and eliminate the device?

Caches are a extremely fast storage systems which holds the copy of the information to be accessed on a temporary basis, so that if that information is needed again, it can be fetched from the cache, allowing faster access and shit.

Some problems caused by the caches is the need to manage it, since they have limited size.

Manufacturing caches is a lot more expensive than manufacturing slow secondary storage devices.

B. What is the relationship between a guest operating system and a host operating system in a system like VMware? What factors need to be considered in choosing the host operating system?

The **host** operating system runs an application like VMware/Parallels/VirtualBox that allows it to run several different **guest** processes (usually operating systems) as independent virtual machines, each being provided with a copy of the host OS.

The application may provide an interface between the guest and the host to communicate with each other.

Factors that need to be considered:

- Each VM needs to have it's own CPU, Memory, Storage, Graphics, Interfaces, et.al, the host must have enough resources to share with the guest whilst running itself without lag.
- The host must have a virtualization layer that allows it to abstract physical hardware into isolated virtual machines.

3.

A. What is the output for the variable `val` in the parent and child process for the code given below. Justify your answer.

a.

```
int main () {
	int val = 5;
	if (fork())
		wait(&val);
	val++;
	printf("%d\n", val);
	return val;
}
```

```
6 // Child process, val = 6
7
/* The wait(&wstatus) will cause the parent process to wait till the child executes, and the val=6 is returned from the child to the parent, where it's incremented to 7.*/
```

b.

```
int main () {
	int val = 5;
	if (fork())
		wait(&val);
	else
		exit(val);
	val++;
	printf("%d\n", val);
	return val;
}
```

```
6
/* Child process exits as fork() in child process fails to fork, and exits with value return value=5, this is passed to main process, where it's incremented once. */
```

B. Discuss how the problem of maintaining coherence of cached data manifests itself in the following processing environments:

a. Single processor systems

In single-processor systems, the memory needs to be updated when a processor issues updates to cached values. These updates can be performed immediately or in a lazy manner.

b. Multi processor systems

In a multiprocessor system, different processors might be caching the same memory location in its local caches. When updates are made, the other cached locations need to be invalidated or updated.

c. Distributed systems

In distributed systems, consistency of cached memory values is not an issue. However, consistency problems might arise when a client caches file data.

4.

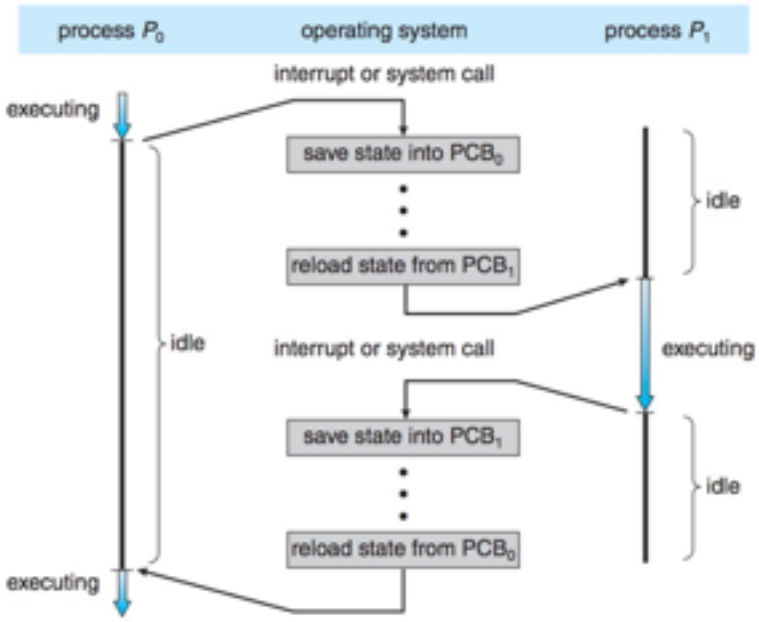
A. Give an example for something that will make a process to do: Also relate your answer for each to the process state diagram.

a. Voluntary context switch

When a process has to wait for an I/O operation, it makes a voluntary context switch.

b. Involuntary context switch

When the time allotted to a process expires, it makes a involuntary context switch.



B. An alert reviewer notices a consistent spelling error in the manuscript on Operating Systems textbook, about to go to press. The book has about 700 pages, each with 50 lines of 80 characters each. How long would it take to process (i.e., make one pass through) the manuscript, assuming that the manuscript is in the following levels of the memory hierarchy:

S.No.	Type	Capacity	Access time
1	Registers	< 1 KB	1 ns
2	Cache	1MB	2 ns
3	Main Memory	1GB	10 ns
4	Magnetic Disk	50GB	10 ms
5	Magnetic Tape	100GB	100 s

For internal storage methods, assume the access time is per character, for disk devices, assume the access time is per block of 1024 characters, and for tape assume the access time is to the time to get to the start of the data, with subsequent access time similar to disk access.

The file size is = 700 * 50 * 80 characters = 2.8 M characters = 2734.375 * 1024 characters.

- It will not fit in the registers.
- It can maybe fit in caches of a large machine.
  - Access time = 2E-9 * 2.8E6 = 5.6E-3 = **5.6 ms**
- For main memory,
  - Access time = 10E-9 * 2.8E6 = 28E-3 = **28 ms**
- For magnetic disk,
  - Each block takes 10ms, total access time = 2735 * 10ms = **27.35 s**
- For magnetic tape
  - Access time = 100 sec + 27.35 sec = **127.35 s**

5.

A. **Given a system with n processes, how many possible ways can those processes be scheduled. Give a formula in terms of n. Also for each of the following transitions between process state, indicate whether transition is possible or not. If its possible, give an example of one thing that would cause it and if its impossible, justify your answer.**

A system with **N** processes can schedule them in **N!** ways.

a. **Run → Ready**

This transition is possible if the process makes an involuntary context switch which can happen if the time allotted to that process runs out.

b. **Waiting → Run**

This transition is not favorable because a process in the waiting states would first have to go to the ready state, and then it will be sent to the processor by the process-scheduler.

c. **Run → Terminated**

This transition is possible if the process completes eggecution and is no longer required.

B. **What will happen to the child processes if the parent process dies in UNIX operating system. Also write how parent process detects the termination of a child process mentioning the required system calls.**

In a UNIX OS, if the parent process diededes the child process will also dieded immediately.

A parent process detects the termination of a child process by the UNIX system call `wait()` or `waitpid()`. The wait function will block the parent process till the child process is eggecuted.
One other non blocking way for a parent process to detect child's termination is for the child to notify the parent process after eggecution.