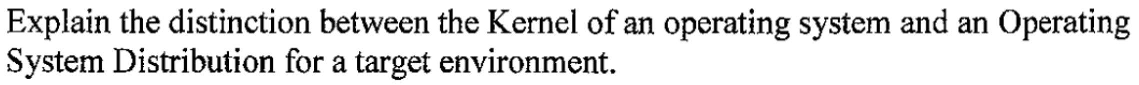
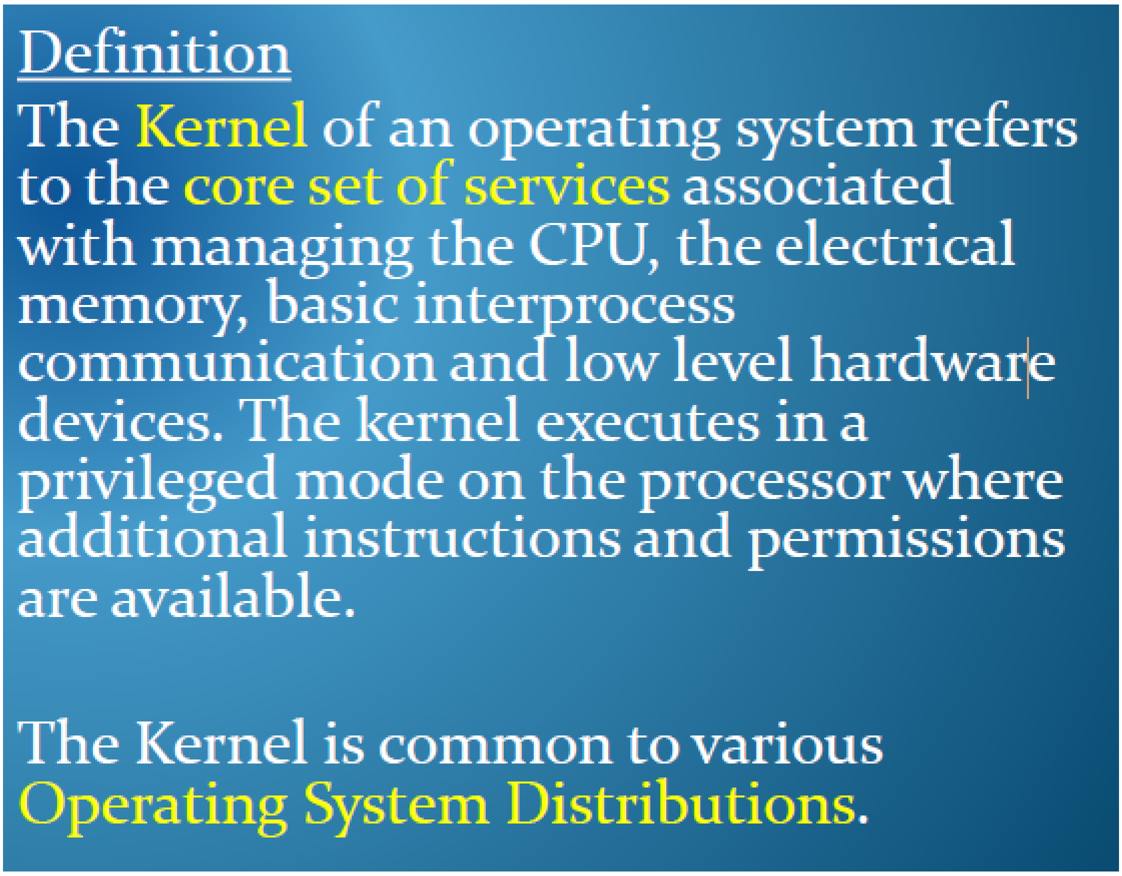
Q1,L2内容



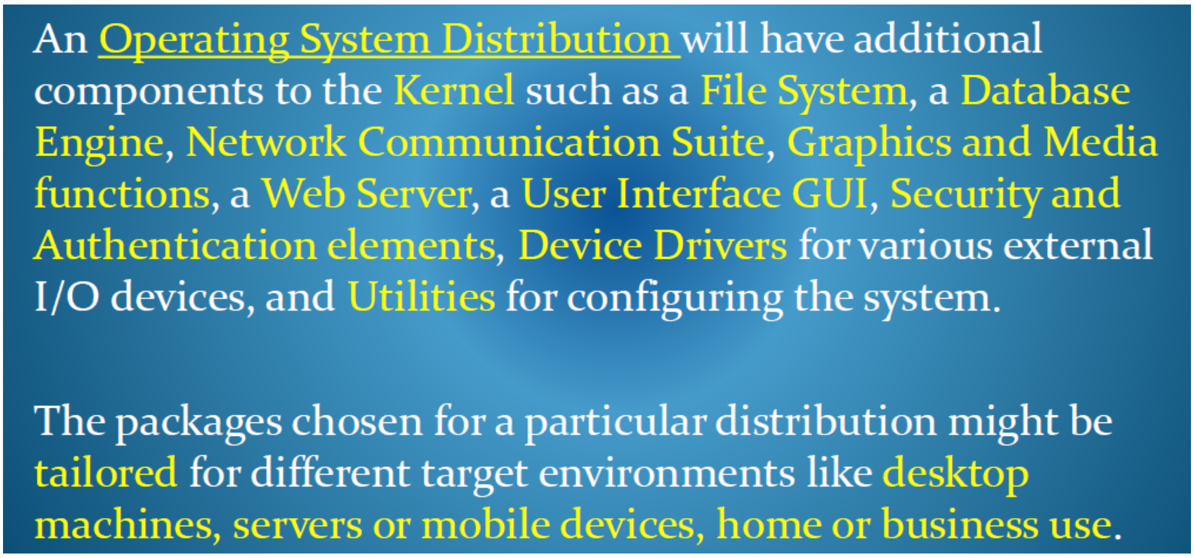
Explain the distinction between the Kernel of an operating system and an Operating System Distribution for a target environment.



Definition

The Kernel of an operating system refers to the core set of services associated with managing the CPU, the electrical memory, basic interprocess communication and low level hardware devices. The kernel executes in a privileged mode on the processor where additional instructions and permissions are available.

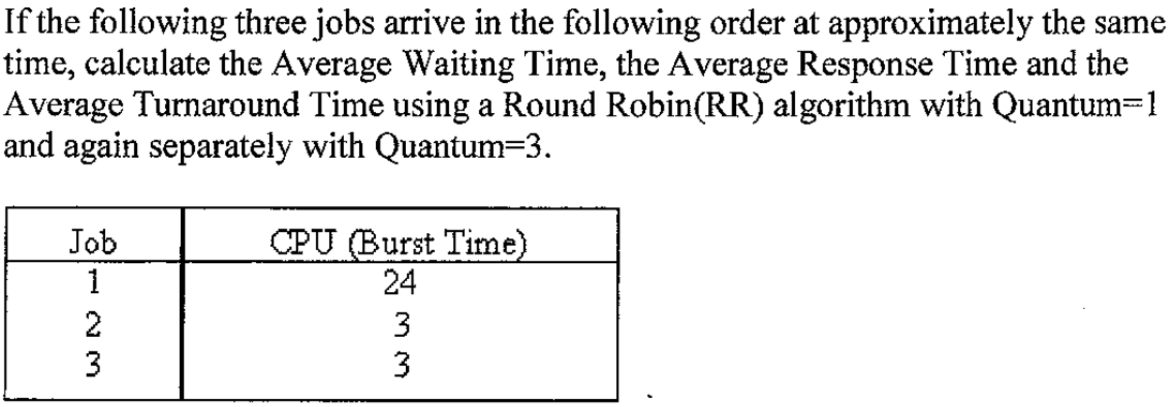
The Kernel is common to various Operating System Distributions.



An Operating System Distribution will have additional components to the Kernel such as a File System, a Database Engine, Network Communication Suite, Graphics and Media functions, a Web Server, a User Interface GUI, Security and Authentication elements, Device Drivers for various external I/O devices, and Utilities for configuring the system.

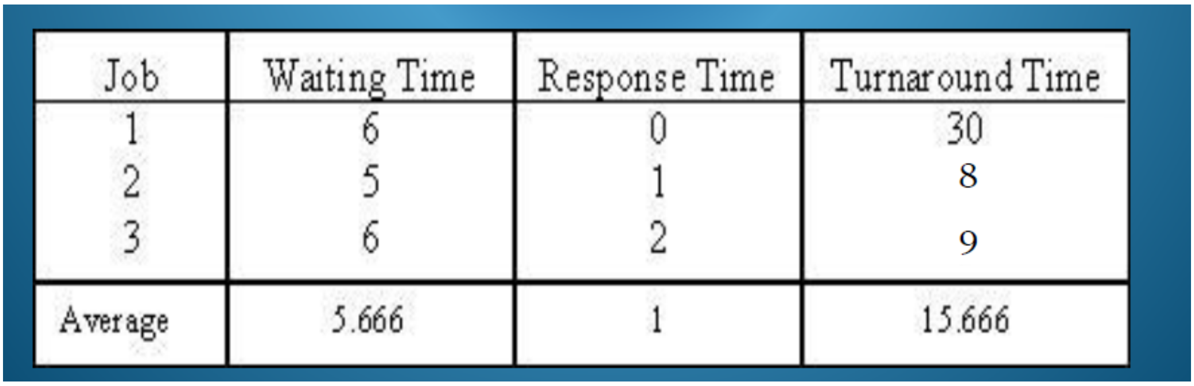
The packages chosen for a particular distribution might be tailored for different target environments like desktop machines, servers or mobile devices, home or business use.

Q2，L4内容

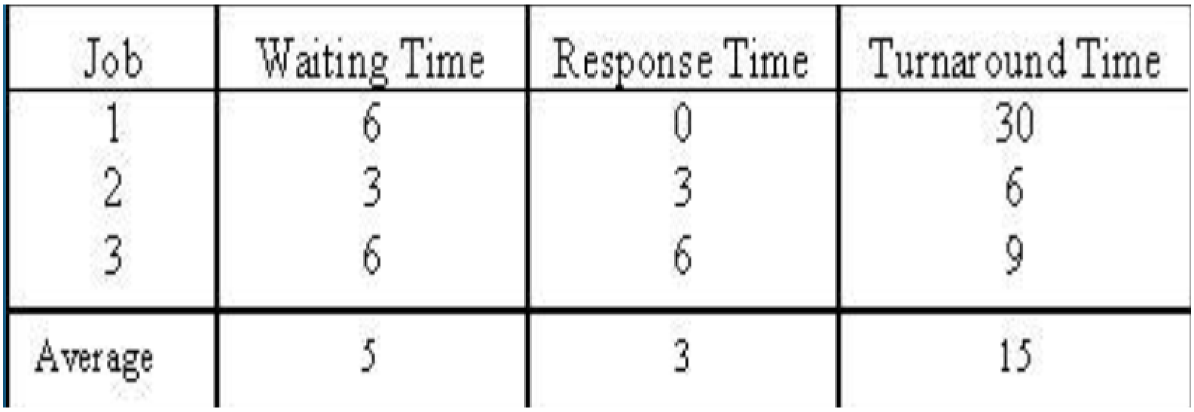


If the following three jobs arrive in the following order at approximately the same time, calculate the Average Waiting Time, the Average Response Time and the Average Turnaround Time using a Round Robin(RR) algorithm with Quantum=1 and again separately with Quantum=3.

Job	CPU (Burst Time)
1	24
2	3
3	3

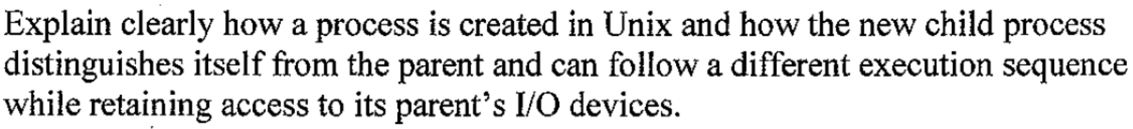


Job	Waiting Time	Response Time	Turnaround Time
1	6	0	30
2	5	1	8
3	6	2	9
Average	5.666	1	15.666



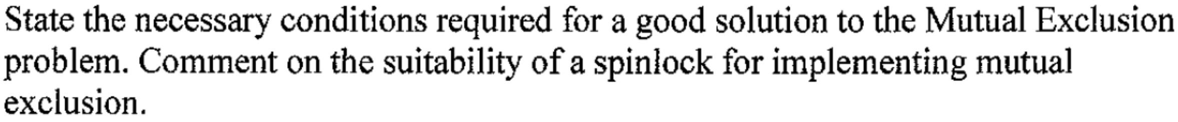
Job	Waiting Time	Response Time	Turnaround Time
1	6	0	30
2	3	3	6
3	6	6	9
Average	5	3	15

Q3，同19年

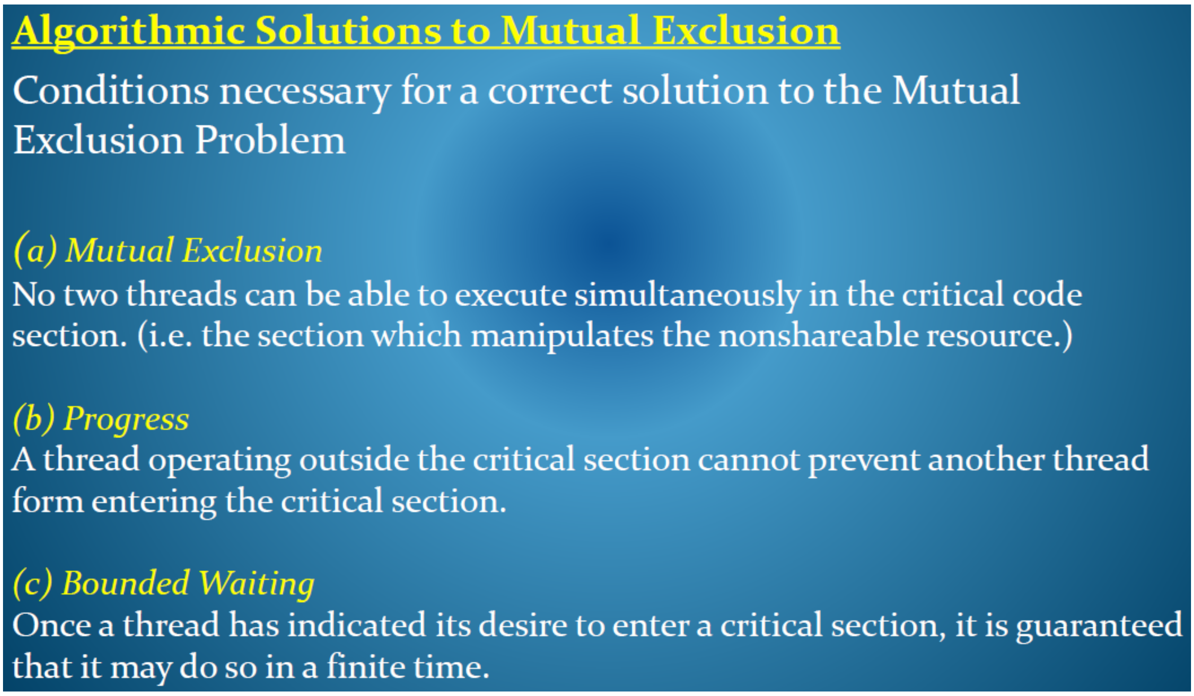


Explain clearly how a process is created in Unix and how the new child process distinguishes itself from the parent and can follow a different execution sequence while retaining access to its parent’s I/O devices.

Q4,L12，13



State the necessary conditions required for a good solution to the Mutual Exclusion problem. Comment on the suitability of a spinlock for implementing mutual exclusion.



Algorithmic Solutions to Mutual Exclusion

Conditions necessary for a correct solution to the Mutual Exclusion Problem

(a) Mutual Exclusion
No two threads can be able to execute simultaneously in the critical code section. (i.e. the section which manipulates the nonshareable resource.)

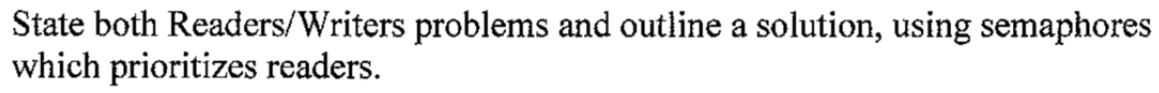
(b) Progress
A thread operating outside the critical section cannot prevent another thread form entering the critical section.

(c) Bounded Waiting
Once a thread has indicated its desire to enter a critical section, it is guaranteed that it may do so in a finite time.

是一种较为安全的解决indivisibility problem的方法。

缺点是The spinlock solutions don’t guarantee Bounded Waiting either. It is pure luck which thread manages to acquire the lock.

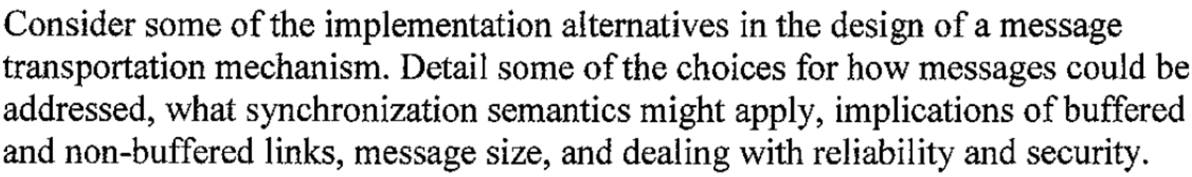
Q5，同19



State both Readers/Writers problems and outline a solution, using semaphores which prioritizes readers.

大概考读写者问题只会考读者优先了，写者优先比较复杂。

Q6,L7,9略开放

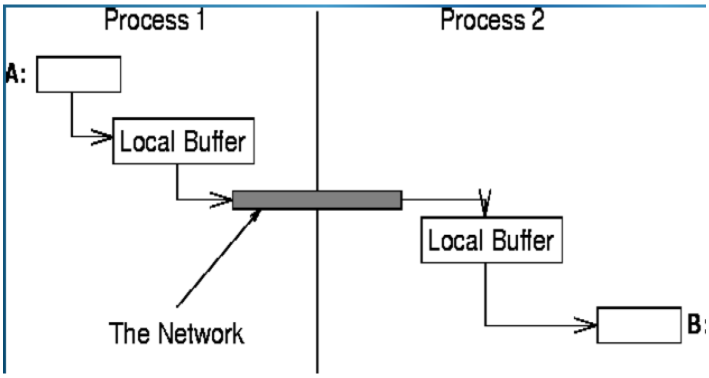


Consider some of the implementation alternatives in the design of a message transportation mechanism. Detail some of the choices for how messages could be addressed, what synchronization semantics might apply, implications of buffered and non-buffered links, message size, and dealing with reliability and security.

Naming发送端与接收端相互间身份确认

Synchronisation就是阻塞，接收端，发送端，阻塞非阻塞一共四种组合。阻塞就是等着回应，非阻塞就是发消息时还能干别的。

Buffering缓冲区是非阻塞的必要条件

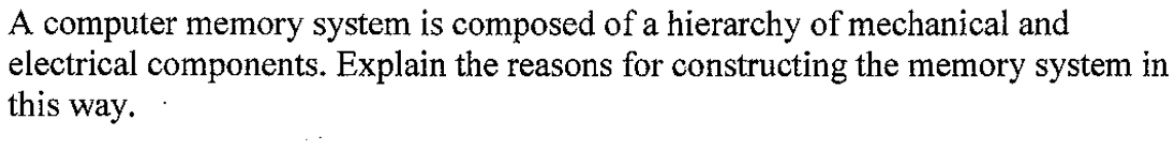


Message Parameters消息包的大小，固定还是动态

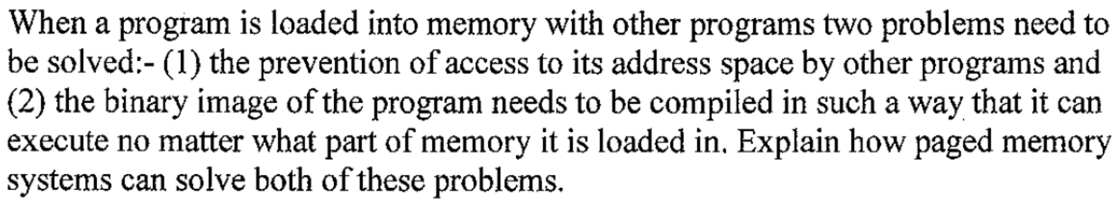
Reliability & Security，乱码，安全通信等等

Q7,课本没有确切答案

个人认为可以从L18冯诺依曼各层次结构入手分析作用


A computer memory system is composed of a hierarchy of mechanical and electrical components. Explain the reasons for constructing the memory system in this way.

Q8，L19&20

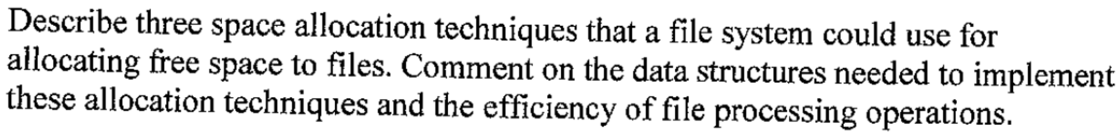

When a program is loaded into memory with other programs two problems need to be solved:- (1) the prevention of access to its address space by other programs and (2) the binary image of the program needs to be compiled in such a way that it can execute no matter what part of memory it is loaded in. Explain how paged memory systems can solve both of these problems.

Protection - As processes only have a logical view of their address spaces, it is not possible for them to access the address space of other processes because the page table won't contain pointers to pages for any other process.

The memory space assigned to a process can be composed of pages which are scattered randomly throughout the physical memory and do not all have to be adjacent to one another. Easy to keep track of free space and to allocate space of requested size. It is easy to relocate or expand the address space of a process by altering the mapping of pages into its page table.

A memory address is divided into two parts, a page number and an offset within that page. The MMU exchanges the page number from the logical address generated by the processor, with the corresponding physical page found in the page table.

Q9,L22，基本是黄字内容


Describe three space allocation techniques that a file system could use for allocating free space to files. Comment on the data structures needed to implement these allocation techniques and the efficiency of file processing operations.

contiguous allocation包括：

Automatic File Space Allocation

Dynamic Space Allocation

其他的：

Linked Allocation结构是a linked list

Linked allocation gives quite good performance for sequential access processing of a file,

Random access to data is not as efficient.

resource overhead associated with the method

Indexed Allocation结构是table

index block is a table which maps logical file blocks to their physical block locations.

顺序和随机都很好，额外资源开销大。