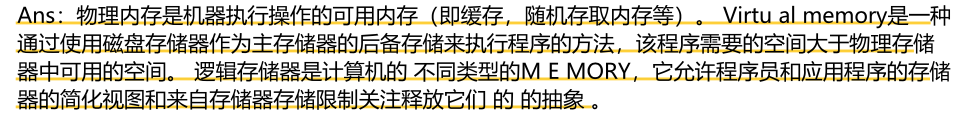
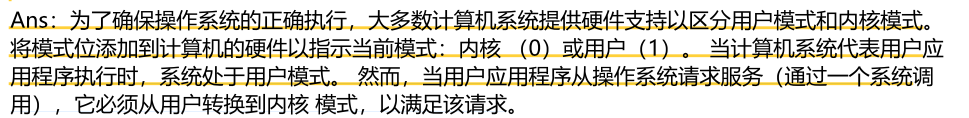
Describe the differences between physical, virtual, and logical memory.

Ans: Physical memory is the memory available for machines to execute operations (i.e., cache, random access memory, etc.). Virtual memory is a method through which programs can be executed that requires space larger than that available in physical memory by using disk memory as a backing store for main memory. Logical memory is an abstraction of the computer’s different types of memory that allows programmers and applications a simplified view of memory and frees them from concern over memory-storage limitations.



Describe the operating system's two modes of operation.

Ans: In order to ensure the proper execution of the operating system, most computer systems provide hardware support to distinguish between user mode and kernel mode. A mode bit is added to the hardware of the computer to indicate the current mode: kernel (0) or user (1). When the computer system is executing on behalf of a user application, the system is in user mode. However, when a user application requests a service from the operating system (via a system call), it must transition from user to kernel mode to fulfill the request.



\_\_\_\_\_ provide(s) an interface to the services provided by an operating system.

A) Shared memory

B) System calls

C) Simulators

D) Communication

\_\_\_\_\_ is not one of the major categories of system calls.

A) Process control

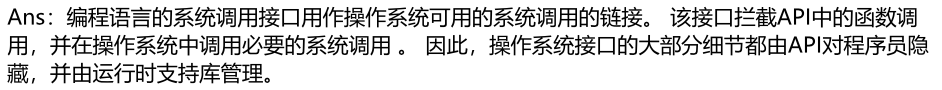
B) Communications

C) Protection

D) Security

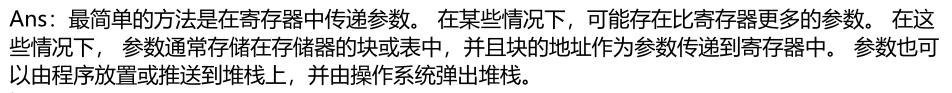
Describe the relationship between an API, the system-call interface, and the operating system.

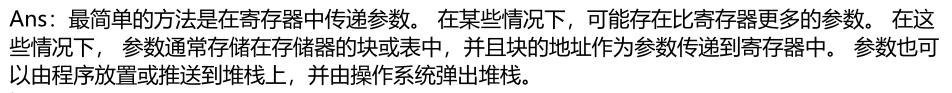
Ans: The system-call interface of a programming language serves as a link to system calls made available by the operating system. This interface intercepts function calls in the API and invokes the necessary system call within the operating system. Thus, most of the details of the operating-system interface are hidden from the programmer by the API and are managed by the run-time support library.



Describe three general methods used to pass parameters to the operating system during system calls.

Ans: The simplest approach is to pass the parameters in registers. In some cases, there may be more parameters than registers. In these cases, the parameters are generally stored in a block, or table, of memory, and the address of the block is passed as a parameter in a register. Parameters can also be placed, or pushed, onto the stack by the program and popped off the stack by the operating system.





Chapter 3:

The \_\_\_\_ of a process contains temporary data such as function parameters, return addresses, and local variables.

A) text section

B) data section

C) program counter

D) stack

A process control block \_\_\_\_.

A) includes information on the process's state

B) stores the address of the next instruction to be processed by a different process

C) determines which process is to be executed next

D) is an example of a process queue

The \_\_\_\_\_\_\_\_\_\_\_\_\_ refers to the number of processes in memory.

A) process count

B) long-term scheduler

C) degree of multiprogramming多道程序化程度

D) CPU scheduler

When a child process is created, which of the following is a possibility in terms of the execution or address space of the child process?

A) The child process runs concurrently同时 with the parent.

B) The child process has a new program loaded into it.

C) The child is a duplicate副本 of the parent.

D) All of the above

A \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ saves the state of the currently running process and restores the state of the next process to run.

A) save-and-restore

B) state switch

C) context switch上下文切换

D) none of the above

A process may transition to the Ready state by which of the following actions?

A) Completion of an I/O event

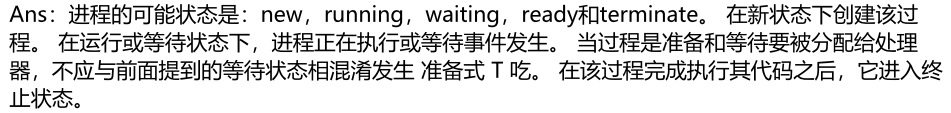
B) Awaiting its turn on the CPU

C) Newly-admitted process

D) All of the above

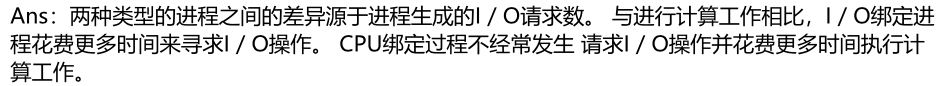
Name and describe the different states that a process can exist in at any given time.

Ans: The possible states of a process are: new, running, waiting, ready, and terminated. The process is created while in the new state. In the running or waiting state, the process is executing or waiting for an event to occur, respectively. The ready state occurs when the process is ready and waiting to be assigned to a processor and should not be confused with the waiting state mentioned earlier. After the process is finished executing its code, it enters the termination state.



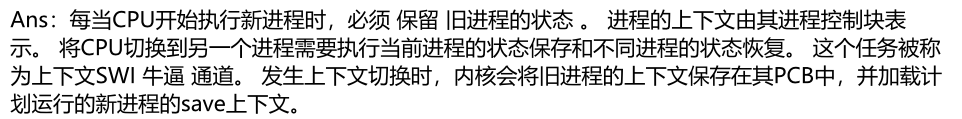
Explain the difference between an I/O-bound process and a CPU-bound process.

Ans: The differences between the two types of processes stem from the number of I/O requests that the process generates. An I/O-bound process spends more of its time seeking I/O operations than doing computational work. The CPU-bound process infrequently requests I/O operations and spends more of its time performing computational work.



Explain the concept of a context switch.上下文切换：

Ans: Whenever the CPU starts executing a new process, the old process's state must be preserved. The context of a process is represented by its process control block. Switching the CPU to another process requires performing a state save of the current process and a state restore of a different process. This task is known as a context switch. When a context switch occurs, the kernel saves the context of the old process in its PCB and loads the saves context of the new process scheduled to run.



For a single-processor system, there will never be more than one process in the Running state.

Ans: True

Chapter 4:

Pthreads refers to \_\_\_\_.

A) the POSIX standard.

B) an implementation for thread behavior.

C) a specification for thread behavior.线程行为的规范

D) an API for process creation and synchronization.

The \_\_\_\_ multithreading model multiplexes many user-level threads to a smaller or equal number of kernel threads.

多对多多线程模型将许多用户级线程复用到较小或相等数量的内核线程

A) many-to-one model

B) one-to-one model

C) many-to-many model

D) many-to-some model

Thread-local storage线程的本地储存 is data that \_\_\_\_.

A) is not associated with any process

B) has been modified by the thread, but not yet updated to the parent process

C) is generated by the thread independent of the thread's process

D) is unique to each thread每个线程是唯一的

A \_\_\_\_ provides an API for creating and managing threads.

A) set of system calls

B) multicore system

C) thread library线程库

D) multithreading model

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

The \_\_\_\_\_ model maps each user-level thread to one kernel thread.

A) many-to-many

B) two-level

C) one-to-one

D) many-to-one

Distinguish between parallelism并行 and concurrency.并发

Ans: A parallel system can perform more than one task simultaneously. A concurrent system supports more than one task by allowing multiple tasks to make progress.

并行系统可以同时执行多个任务，并发系统允许多个任务取得进展，从而支持多个任务。

Each thread has its own register set and stack.

Ans: True

Linux distinguishes between processes and threads.

Ans: False

Chapter 5:

\_\_\_\_ is the number of processes that are completed per time unit.

A) CPU utilization

B) Response time

C) Turnaround time

D) Throughput吞吐量

Which of the following is true of cooperative scheduling?

A) It requires a timer.

B) A process keeps the CPU until it releases the CPU either by terminating or by switching to the waiting state.

C) It incurs a cost associated with access to shared data.

D) A process switches from the running state to the ready state when an interrupt occurs.

\_\_\_\_ scheduling is approximated by predicting the next CPU burst with an exponential average of the measured lengths of previous CPU bursts.

A) Multilevel queue

B) RR

C) FCFS

D) SJF

The \_\_\_\_ scheduling algorithm is designed especially for time-sharing systems.时间共享系统

A) SJF

B) FCFS

C) RR

D) Multilevel queue

Which of the following scheduling algorithms must be nonpreemptive?非抢占式

A) SJF

B) RR

C) FCFS

D) priority algorithms

Which of the following is true of multilevel queue scheduling?多级队列

A) Processes can move between queues.

B) Each queue has its own scheduling algorithm.每个队列有自己的调度算法

C) A queue cannot have absolute priority over lower-priority queues.

D) It is the most general CPU-scheduling algorithm.

\_\_\_\_\_\_ allows a thread to run on only one processor.

A) Processor affinity处理器亲和力

B) Processor set

C) NUMA

D) Load balancing

A significant problem with priority scheduling algorithms is \_\_\_\_\_.

A) complexity

B) starvation

C) determining the length of the next CPU burst

D) determining the length of the time quantum

The rate of a periodic task in a hard real-time system硬实时系统 is \_\_\_\_, where *p* is a period and *t* is the processing time.

A) 1/*p*

B) *p/t*

C) 1*/t*

D) *pt*

Explain the concept of a CPU–I/O burst cycle.

Ans: The lifecycle of a process can be considered to consist of a number of bursts belonging to two different states. All processes consist of CPU cycles and I/O operations. Therefore, a process can be modeled as switching between bursts of CPU execution and I/O wait.

一个进程的生命周期可以被认为是由多个属于两个不同状态的突发组成的。所有进程都由CPU周期和I/O操作组成。因此，可以将进程建模为在CPU执行和I/O等待之间切换。

What role does the dispatcher play in CPU scheduling?

Ans: The dispatcher gives control of the CPU to the process selected by the short-term scheduler. To perform this task, a context switch, a switch to user mode, and a jump to the proper location in the user program are all required. The dispatch should be made as fast as possible. The time lost to the dispatcher is termed dispatch latency.

调度程序将CPU的控制权交给短期调度程序选择的进程。要执行此任务，需要上下文切换、切换到用户模式以及跳转到用户程序中的适当位置。调度应该尽快。调度器损失的时间称为调度延迟。

What effect does the size of the time quantum have on the performance of an RR algorithm?

Ans: At one extreme, if the time quantum is extremely large, the RR policy is the same as the FCFS policy. If the time quantum is extremely small, the RR approach is called processor sharing and creates the appearance that each of *n* processes has its own processor running at 1/*n* the speed of the real processor.

在一个极端情况下，如果时间量非常大，RR策略与FCFS策略相同。如果时间量非常小，RR方法被称为处理器共享，并创建一个外观，即n个进程中的每个进程都有自己的处理器以实际处理器速度的1/n运行。

Explain the process of starvation and how aging can be used to prevent it.

Ans: Starvation occurs when a process is ready to run but is stuck waiting indefinitely for the CPU. This can be caused, for example, when higher-priority processes prevent low-priority processes from ever getting the CPU. Aging involves gradually increasing the priority of a process so that a process will eventually achieve a high enough priority to execute if it waited for a long enough period of time.

Explain the fundamental difference between asymmetric and symmetric multiprocessing.

非对称和对称多处理

Ans: In asymmetric multiprocessing, all scheduling decisions, I/O, and other system activities are handled by a single processor, whereas in SMP, each processor is self-scheduling.

Round-robin (RR) scheduling degenerates to first-come-first-served (FCFS) scheduling if the time quantum is too long.

\_\_\_\_\_\_\_\_\_\_ involves the decision of which kernel thread to schedule onto which CPU.

A) Process-contention scope

B) System-contention scope

C) Dispatcher

D) Round-robin scheduling

Which of the following is true of earliest-deadline-first (EDF) scheduling algorithm?

A) When a process becomes runnable, it must announce its deadline requirements to the system.

B) Deadlines are assigned as following: the earlier the deadline, the lower the priority; the later the deadline, the higher the priority.

C) Priorities are fixed; that is, they cannot be adjusted when a new process starts running.

D) It assigns priorities statically according to deadline.

Chapter 6:

A race condition \_\_\_\_.

A) results when several threads try to access the same data concurrently

B) results when several threads try to access and modify the same data concurrently

多个线程试图同时访问和修改相同的数据时的结果

C) will result only if the outcome of execution does not depend on the order in which instructions are executed

D) None of the above

An instruction that executes atomically \_\_\_\_.原子指令

A) must consist of only one machine instruction

B) executes as a single, uninterruptible unit单个不间断单元执行

C) cannot be used to solve the critical section problem

D) All of the above

A mutex lock \_\_\_\_.互斥锁

A) is exactly like a counting semaphore

B) is essentially a boolean variable本质上是一个布尔变量

C) is not guaranteed to be atomic

D) can be used to eliminate busy waiting

In Peterson's solution, the \_\_\_\_ variable indicates if a process is ready to enter its critical section.

A) turn谦让

B) lock

C) flag[i]

D) turn[i]

The first readers-writers problem \_\_\_\_.

A) requires that, once a writer is ready, that writer performs its write as soon as possible.

B) is not used to test synchronization primitives.

C) requires that no reader will be kept waiting unless a writer has already obtained permission to use the shared database.

D) requires that no reader will be kept waiting unless a reader has already obtained permission to use the shared database.

What is the correct order of operations for protecting a critical section using mutex locks?

使用互斥锁保护关键部分的正确操作是什么

A) release() followed by acquire()

B) acquire() followed by release()

C) wait() followed by signal()

D) signal() followed by wait()

What is the correct order of operations for protecting a critical section using a binary semaphore?

A) release() followed by acquire()

B) acquire() followed by release()

C) wait() followed by signal()二值信号量

D) signal() followed by wait()

Wait signal

A solution to the critical section problem临界区问题 does not have to satisfy which of the following requirements?

A) mutual exclusion互斥

B) progress空闲让进

C) atomicity

D) bounded waiting有限等待

A(n) \_\_\_\_\_\_\_ refers to where a process is accessing/updating shared data.

A) critical section临界区

B) entry section

C) mutex

D) test-and-set

\_\_\_\_\_ can be used to prevent busy waiting 忙等 when implementing a semaphore.

A) Spinlocks

B) Waiting queues等待队列

C) Mutex lock

D) Allowing the wait() operation to succeed

What is the purpose of the mutex semaphore in the implementation of the bounded-buffer problem using semaphores?

A) It indicates the number of empty slots in the buffer.

B) It indicates the number of occupied slots in the buffer.

C) It controls access to the shared buffer.

D) It ensures mutual exclusion.确保相互排斥

How many philosophers may eat simultaneously in the Dining Philosophers problem with 5 philosophers?

A) 1

B) 2

C) 3

D) 5

Which of the following statements is true?

A) A counting semaphore can never be used as a binary semaphore.

B) A binary semaphore can never be used as a counting semaphore.

C) Spinlocks can be used to prevent busy waiting in the implementation of semaphore.

自旋锁可用于防止在信号量的实现中忙等待

D) Counting semaphores can be used to control access to a resource with a finite number of instances.

When using semaphores, a process invokes the wait() operation before accessing its critical section, followed by the signal() operation upon completion of its critical section. Consider reversing the order of these two operations—first calling signal(), then calling wait(). What would be a possible outcome of this?

A) Starvation is possible.

B) Several processes could be active in their critical sections at the same time.几个进程同时在临界区被激活

C) Mutual exclusion is still assured.

D) Deadlock is possible.

What three conditions must be satisfied in order to solve the critical section problem?

Ans: In a solution to the critical section problem, no thread may be executing in its critical section if a thread is currently executing in its critical section. Furthermore, only those threads that are not executing in their critical sections can participate in the decision on which process will enter its critical section next. Finally, a bound must exist on the number of times that other threads are allowed to enter their critical state after a thread has made a request to enter its critical state.

在临界区问题的解决方案中，如果一个线程当前正在其临界区执行，则没有线程可以在其临界区执行。此外，只有那些不在其关键部分执行的线程才能参与决定哪个进程将进入其关键部分。最后，在一个线程请求进入其临界状态后，一个界限必须存在于允许其他线程进入其临界状态的次数上。

Explain two general approaches to handle critical sections in operating systems.

Ans: Critical sections may use preemptive or nonpreemptive kernels. A preemptive kernel allows a process to be preempted while it is running in kernel mode. A nonpreemptive kernel does not allow a process running in kernel mode to be preempted; a kernel-mode process will run until it exits kernel mode, blocks, or voluntarily yields control of the CPU. A nonpreemptive kernel is essentially free from race conditions on kernel data structures, as the contents of this register will be saved and restored by the interrupt handler.

临界区：可以使用抢占式或非抢占式内核。抢占式内核允许进程在内核模式下运行时被抢占。非抢占式内核不允许在内核模式下运行的进程被抢占；内核模式进程将一直运行，直到它退出内核模式、阻塞或自动获得对CPU的控制。一个非抢答的内核基本上不受内核数据结构上的竞争条件的影响，因为这个寄存器的内容将由中断处理程序保存和恢复。

Write two short methods that implement the simple semaphore wait() and signal() operations on global variable S.

Ans: wait (S) {

while (S <= 0);

S--;

}

signal (S) {

S++;

}

Assume you had a function named update() that updates shared data. Illustrate how a mutex lock named mutex might be used to prevent a race condition in update().

Ans:

void update()

{

mutex.acquire();

// update shared data

mutex.release();

}

Chapter 7:

A deadlocked state occurs whenever \_\_\_\_.

A) a process is waiting for I/O to a device that does not exist

B) the system has no available free resources

C) every process in a set is waiting for an event that can only be caused by another process in the set

D) a process is unable to release its request for a resource after use

One necessary condition for deadlock is \_\_\_\_, which states that at least one resource must be held in a nonsharable mode.

A) hold and wait

B) mutual exclusion互斥

C) circular wait

D) no preemption

One necessary condition for deadlock is \_\_\_\_\_\_, which states that a process must be holding one resource and waiting to acquire additional resources.

A) hold and wait

B) mutual exclusion

C) circular wait

D) no preemption

One necessary condition for deadlock is \_\_\_\_\_\_, which states that a resource can be released only voluntarily自愿 by the process holding the resource.

A) hold and wait

B) mutual exclusion

C) circular wait

D) no preemption非抢占式

One necessary condition for deadlock is \_\_\_\_\_\_, which states that there is a chain of waiting processes whereby P0 is waiting for a resource held by P1, P1 is waiting for a resource held by P2, and P*n* is waiting for a resource held by P0.

A) hold and wait

B) mutual exclusion

C) circular wait循环等待

D) no preemption

In a system resource-allocation graph, \_\_\_\_.系统资源分配图

A) a directed edge from a process to a resource is called an assignment edge

B) a directed edge from a resource to a process is called a request edge

C) a directed edge from a process to a resource is called a request edge请求边

D) None of the above

A cycle in a resource-allocation graph is \_\_\_\_.

A) a necessary and sufficient condition for deadlock in the case that each resource has more than one instance

B) a necessary and sufficient condition for a deadlock in the case that each resource has exactly one instance每个资源只有一个实例的情况下，死锁的必要和充分条件

C) a sufficient condition for a deadlock in the case that each resource has more than once instance

D) is neither necessary nor sufficient for indicating deadlock in the case that each resource has exactly one instance

Describe the four conditions that must hold simultaneously in a system if a deadlock is to occur.

Ans: For a set of processes to be deadlocked: at least one resource must remain in a nonsharable mode, a process must hold at least one resource and be waiting to acquire additional resources held by other processes, resources in the system cannot be preempted, and a circular wait has to exist between processes.

Describe two protocols to ensure that the hold-and-wait condition never occurs in a system.

Ans: One protocol requires each process to request and be allocated all its resources before it begins execution. We can implement this provision by requiring that system calls requesting resources for a process precede all other system calls. An alternative protocol allows a process to request resources only when it has none. A process may request some resources and use them. Before it can request any additional resources, however, it must release all the resources that it is currently allocated.

What is one way to ensure that a circular-wait condition does not occur?

Ans: One way to ensure that this condition never holds is to impose a total ordering of all resource types, and to require that each process requests resources in an increasing order of enumeration. This can be accomplished by assigning each resource type a unique integer number to determine whether one precedes another in the ordering.

Describe two methods for eliminating processes by aborting a process.

Ans: The first method is to abort all deadlocked processes. Aborting all deadlocked processes will clearly break the deadlock cycle; however, the deadlocked processes may have to be computed for a long time, and results of these partial computations must be discarded and will probably have to be recomputed later. The second method is to abort one process at a time until the deadlock cycle is eliminated. Aborting one process at a time incurs considerable overhead, since, after each process is aborted, a deadlock-detection algorithm must be invoked to determine whether any processes are still deadlocked.

Chapter8-9:

An address generated by a CPU is referred to as a \_\_\_\_.

A) physical address

B) logical address

C) post relocation register address

D) Memory-Management Unit (MMU) generated address

Suppose a program is operating with execution-time binding and the physical address generated is 300. The relocation register is set to 100. What is the corresponding logical address?

A) 199

B) 201

C) 200

D) 300

The mapping of a logical address to a physical address is done in hardware by the \_\_\_\_\_\_\_\_.

A) memory-management-unit (MMU)

B) memory address register

C) relocation register

D) dynamic loading register

\_\_\_\_\_ is the dynamic storage-allocation algorithm which results in the smallest leftover hole in memory.

A) First fit

B) Best fit

C) Worst fit

D) None of the above

Consider a logical address with a page size of 8 KB. How many bits must be used to represent the page offset in the logical address?

A) 10

B) 8

C) 13

D) 12

Assume a system has a TLB hit ratio of 90%. It requires 15 nanoseconds to access the TLB, and 85 nanoseconds to access main memory. What is the effective memory access time in nanoseconds for this system?

A) 108.5 0.9\*（15+85）+0.1\*（15+85\*2）

B) 100

C) 22

D) 176.5

Given the logical address 0xAEF9 (in hexadecimal)十六进制 with a page size of 256 bytes, what is the page number?

A) 0xAE

B) 0xF9

C) 0xA

D) 0x00F9

Given the logical address 0xAEF9 (in hexadecimal) with a page size of 256 bytes, what is the page offset?

A) 0xAE

B) 0xF9

C) 0xA

D) 0xF900

Consider a 32-bit address for a two-level paging system with an 8 KB page size. The outer page table has 1024 entries. How many bits are used to represent the second-level page table?

A) 10

B) 8

C) 12

D) 9

8k=213

32-13=19

1024=210

19-10=9

With segmentation, a logical address consists of \_\_\_\_\_.

A) segment number and offset

B) segment name and offset

C) segment number and page number

D) segment table and segment number

Which of the following data structures is appropriate for placing into its own segment?

A) heap

B) kernel code and data

C) user code and data

D) all of the above

Assume the value of the base and limit registers are 1200 and 350 respectively. Which of the following addresses is legal?

A) 355

B) 1200

C) 1551

D) all of the above

What is the advantage of using dynamic loading?

With dynamic loading a program does not have to be stored, in its entirety, in main memory. This allows the system to obtain better memory-space utilization. This also allows unused routines to stay out of main memory so that memory can be used more effectively. For example, code used to handle an obscure error would not always use up main memory.

What is the context switch time, associated with swapping, if a disk drive with a transfer rate of 2 MB/s is used to swap out part of a program that is 200 KB in size? Assume that no seeks are necessary and that the average latency is 15 ms. The time should reflect only the amount of time necessary to swap out the process.

Ans: 200KB / 2048 KB per second + 15 ms = 113 ms

When does external fragmentation occur?

Ans: As processes are loaded and removed from memory, the free memory space is broken into little pieces. External fragmentation exists when there is enough total memory space to satisfy a request, but the available spaces are not contiguous; storage is fragmented into a large number of small holes. Both the first-fit and best-fit strategies for memory allocation suffer from external fragmentation.

Distinguish between internal and external fragmentation.

Ans: Fragmentation occurs when memory is allocated and returned to the system. As this occurs, free memory is broken up into small chunks, often too small to be useful. External fragmentation occurs when there is sufficient total free memory to satisfy a memory request, yet the memory is not contiguous, so it cannot be assigned. Some contiguous allocation schemes may assign a process more memory than it actually requested (i.e. they may assign memory in fixed-block sizes). Internal fragmentation occurs when a process is assigned more memory than it has requested and the wasted memory fragment is internal to a process.

Explain the basic method for implementing paging.

Ans: Physical memory is broken up into fixed-sized blocks called frames while logical memory is broken up into equal-sized blocks called pages. Whenever the CPU generates a logical address, the page number and offset into that page is used, in conjunction with a page table, to map the request to a location in physical memory.

Describe how a transaction look-aside buffer (TLB) assists in the translation of a logical address to a physical address.

Ans: Typically, large page tables are stored in main memory, and a page-table base register points are saved to the page table. Therefore, two memory accesses are needed to access a byte (one for the page-table entry, one for the byte), causing memory access to be slowed by a factor of 2. The standard solution to this problem is to use a TLB, a special, small fast-lookup hardware cache. The TLB is associative, high speed memory. Each entry consists of a key and value. An item is compared with all keys simultaneously, and if the item is found, the corresponding value is returned.

describe how a logical address is translated to a physical address.

Ans: A logical address is generated by the CPU. This logical address consists of a page number and offset. The TLB is first checked to see if the page number is present. If so, a TLB hit, the corresponding page frame is extracted from the TLB, thus producing the physical address. In the case of a TLB miss, the page table must be searched according to page number for the corresponding page frame.

In the enhanced second chance algorithm, which of the following ordered pairs represents a page that would be the best choice for replacement?二次机会置换算法

A) (0,0)

B) (0,1)

C) (1,0)

D) (1,1)

\_\_\_\_\_ is the algorithm implemented on most systems.

A) FIFO

B) Least frequently used

C) Most frequently used

D) LRU

\_\_\_\_\_ occurs when a process spends more time paging than executing.

A) Thrashing抖动

B) Memory-mapping

C) Demand paging

D) Swapping

Explain the sequence of events that happens when a page-fault occurs.

Ans: When the operating system cannot load the desired page into memory, a page-fault occurs. First, the memory reference is checked for validity. In the case of an invalid request, the program will be terminated. If the request was valid, a free frame is located. A disk operation is then scheduled to read the page into the frame just found, update the page table, restart the instruction that was interrupted because of the page fault, and use the page accordingly.

Chapter 10-13

The path name /home/people/os-student/chap10.txt is an example of

A) a relative path name

B) an absolute path name绝对路径

C) a relative path name to the current directory of /home

D) an invalid path name

The path name os-student/src/vm.c is an example of

A) a relative path name

B) an absolute path name

C) a relative path name to the current directory of /os-student

D) an invalid path name

Which of the following is the simplest method for implementing a directory?

A) tree data structure

B) linear list

C) hash table

D) nonlinear list

Which of the following allocation methods ensures that only one access is needed to get a disk block using direct access?

A) linked allocation

B) indexed allocation

C) hashed allocation

D) contiguous allocation

The file-allocation table (FAT) used in MS-DOS is an example of \_\_\_\_\_.

A) contiguous allocation

B) indexed allocation

C) linked allocation

D) multilevel index

The surface of a magnetic disk platter is divided into \_\_\_\_.

A) sectors

B) arms

C) tracks

D) cylinders

The SSTF scheduling algorithm \_\_\_\_.

A) services the request with the maximum seek time

B) services the request with the minimum seek time以最短寻道时间为请求提供服务

C) chooses to service the request furthest from the current head position

D) None of the above

Low-level formatting \_\_\_\_.低级格式化

A) does not usually provide an error-correcting code

B) is usually performed by the purchaser of the disk device

C) is different from physical formatting

D) divides a disk into sections that the disk controller can read and write

A RAID structure \_\_\_\_.独立磁盘冗余阵列

A) is primarily used for security reasons

B) is primarily used to ensure higher data reliability

C) stands for redundant arrays of inexpensive disks

D) is primarily used to decrease the dependence on disk drives减少对磁盘驱动器的依赖

Which of the following disk head scheduling algorithms does not take into account the current position of the disk head?

A) FCFS

B) SSTF

C) SCAN

D) LOOK

RAID level \_\_\_\_ is the most common parity RAID system.

A) 0

B) 0+1

C) 4

D) 5

Which RAID level is best for storing large volumes of data?

A) RAID levels 0 + 1 and 1 + 0

B) RAID level 3

C) RAID level 4

D) RAID level 5

Which of the following statements regarding solid state disks (SSDs) is false?

A) They generally consume more power than traditional hard disks.

B) They have the same characteristics as magnetic hard disks, but can be more reliable.

C) They are generally more expensive per megabyte than traditional hard disks.

D) They have no seek time or latency.

Ans: A