

3) 信息保护。对信息的逻辑单位进行保护,分段能更有效方便地实现信息保护功能。

4) 动态增长。在实际应用中,有些段特别是数据段,在使用过程中会不断增长,事先又无法确切知道增长多少。分段存储管理方式能较好解决这个问题。

5) 动态链接。运行时先将主程序对应的目标程序装入内存并启动运行,运行过程中又需要调用某段时,才将该段调入内存链接。所以动态链接也要求以段作为管理单位。

24. 在具有快表的段页式存储管理方式中,如何实现地址变换?

答:在CPU给出有效地址后,由地址变换机构自动将页号P送入高速缓冲寄存器,并将此页号与高速缓存中的所有页号比较,若找到匹配页号,表示要访问的页表项在快表中。可直接从快表读出该页对应物理块号,送到物理地址寄存器中。如快表中没有对应页表项,则再访问内存页表,找到后,把从页表项中读出物理块号送地址寄存器;同时修改快表,将此页表项存入快表。但若寄存器已满,则OS必须找到合适的页表项换出。

25. 为什么说分段系统较之分页系统更易于实现信息共享和保护?

答:a. 对于分页系统,每个页面是分散存储的,为了实现信息共享和保护,则页面之间需要一一对应起来,为此需要建立大量的页表项;

b. 而对于分段系统,每个段都从0开始编址,并采用一段连续的地址空间,这样在实现共享和保护时,只需为所要共享和保护的程序设置一个段表项,将其中的基址与内存地址一一对应起来即可。

26. 分页和分段有何区别?

答:a. 分页和分段都采用离散分配的方式,且都要通过地址映射机构来实现地址变换,这是它们的共同点;

b. 对于它们的不同点有三,第一,从功能上看,页是信息的物理单位,分页是为实现离散分配方式,以消减内存的外零头,提高内存的利用率,即满足系统管理的需要,而不是用户的需要;而段是信息的逻辑单位,它含有一组其意义相对完整的信息,目的是为了能更好地满足用户的需要;第二页的大小固定且由系统确定,而段的长度却不固定,决定于用户所编写的程序;第三分页的作业地址空间是一维的,而分段的作业地址空间是二维的。

27. 试全面比较连续分配和离散分配方式。

答:a. 连续分配是指为一个用户程序分配一个连续的地址空间,包括单一连续分配方式和分区式分配方式,前者将内存分为系统区和用户区,系统区供操作系统使用,用户区供用户使用,是最简单的一种存储方式,但只能用于单用户单任务的操作系统中;分区式分配方式分为固定分区和动态分区,固定分区是最简单的多道程序的存储管理方式,由于每个分区的大小固定,必然会造成存储空间的浪费;动态分区是根据进程的实际需要,动态地为之分配连续的内存空间,常用三种分配算法:首次适应算法,该法容易留下许多难以利用的小空闲分区,加大查找开销;循环首次适应算法,该算法能使内存中的空闲分区分布均匀,但会致使缺少大的空闲分区;最佳适应算法,该算法也易留下许多难以利用的小空闲区;

b. 离散分配方式基于将一个进程直接分散地分配到许多不相邻的分区中的思想,

分为分页式存储管理,分段存储管理和段页式存储管理。分页式存储管理旨在提高内存利用率,满足系统管理的需要,分段式存储管理则旨在满足用户(程序员)的需要,在实现共享和保护方面优于分页式存储管理,而段页式存储管理则是将两者结合起来,取长补短,即具有分段系统便于实现,可共享,易于保护,可动态链接等优点,又能像分页系统那样很好的解决外部碎片的问题,以及为各个分段可离散分配内存等问题,显然是一种比较有效的存储管理方式;

c. 综上所述,连续分配方式和离散分配方式各有各的特点,应根据实际情况加以改进和利用。

## 第五章

1. 常规存储器管理方式具有哪两大特征?它对系统性能有何影响?

答:一次性:进程必须全部装入内存,对空间浪费非常大;

驻留性:在程序运行过程中,进程全部驻留在内存,暂时不用的数据无法释放。

2. 什么是程序运行时的时间局限性和空间局限性？

答：（1）时间局限性：如果程序中的某条指令一旦执行，则不久的将来该指令可能再次被执行；如果某个存储单元被访问，则不久的将来该存储单元可能再次被访问。产生时间局限性的典型原因是在程序中存在大量的循环操作。

（2）空间局限性：一旦程序访问了某个存储单元，则在不久的将来，其附近的存储单元也最有可能被访问，即程序在一段时间内所访问的地址，可能集中在一定的范围内。产生空间局限性的典型原因是程序是顺序执行的。

3. 虚拟存储器有哪些特征？其中最本质的特征是什么？

答：虚拟存储器有多次性、对换性、虚拟性三大特征。最本质的特征是虚拟性。

4. 实现虚拟存储器需要哪些硬件支持？

a. 请求分页（段）的页（段）表机制 b. 缺页（段）中断机构 c. 地址变换机构

5. 实现虚拟存储器需要哪几个关键技术？

答：（1）在分页请求系统中是在分页的基础上，增加了请求调页功能和页面置换功能所形成的页式虚拟存储系统。允许只装入少数页面的程序（及数据），便启动运行。（2）在请求分段系统中是在分段系统的基础上，增加了请求调段及分段置换功能后形成的段式虚拟存储系统。允许只装入少数段（而非所有段）的用户程序和数据，即可启动运行。

6. 在请求分页系统中，页表应包括哪些数据项？每项的作用是什么？

答：页表应包括：页号、物理块号、状态位 P、访问字段 A、修改位 M 和外存地址。其中状态位 P 指示该页是否调入内存，供程序访问时参考；访问字段 A 用于记录本页在一段时间内被访问的次数，或最近已有多长时间未被访问，提供给置换算法选择换出页面时参考；修改位 M 表示该页在调入内存后是否被修改过；外存地址用于指出该页在外存上的地址，通常是物理块号，供调入该页时使用。

7. 试比较缺页中断机构与一般的中断，它们之间有何明显的区别？

答：a. 一般中断只需要保护现场然后就直接跳到需及时处理的地方。

b. 缺页中断除了保护现场之外，还要判断内存中是否有足够的空间存储所需的页或段，然后再把所需页调进来再使用。

8. 试说明请求分页系统中的地址变换过程。

- 1) 取逻辑地址分解为页号 **P** 和页内偏移 **w**;
- 2) 根据页号查找页表，获得该页的描述信息;
- 3) 若该页中断位为 **1**，产生缺页中断;
- 4) 更新该页的描述信息;
- 5) 根据页块号和页内偏移 **w**，计算物理地址。

9. 何谓固定分配局部置换和可变分配和全局置换的内存分配策略？

**(1) 固定分配局部置换**

固定分配是指，为每个进程分配一组固定数目的物理块，在进程运行期间不再改变。

局部置换是指，如果进程在运行中发现缺页，则只能从分配给该进程的  $n$  个页面中，选出一页换出，然后再调入一页。

**(2) 可变分配全局置换**

可变分配是指，先为每个进程分配一定数目的物理块，在进程运行期间，可根据情况做适当地改变。

全局置换是指，如果进程在运行中发现缺页，则将 OS 所保留的空闲物理块或者以所有进程的全部物理块为标的，选择一块换出，然后将所缺之页调入。

10. 在请求分页系统中，应从何处将所需页面调入内存？

答：请求分页系统中的缺页从何处调入内存分三种情况：

(1) 系统拥有足够对换区空间时，可以全部从对换区调入所需页面，提高调页速度。在进程运行前将与该进程有关的文件从文件区拷贝到对换区。

(2) 系统缺少足够对换区空间时，不被修改的文件直接从文件区调入；当换出这些页面时，未被修改的不必换出，再调入时，仍从文件区直接调入。对于可能修改的，在换出时便调到对换区，以后需要时再从对换区调入。

(3) UNIX 方式。未运行页面从文件区调入。曾经运行过但被换出页面，下次从对换区调入。UNIX 系统允许页面共享，某进程请求的页面有可能已调入内存，直接使用不再调入。

11. 试说明在请求分页系统中页面的调入过程。

每当程序所要访问的页面未在内存时（存在位为“0”），便向CPU发出一缺页中断，中断处理程序首先保留CPU环境，分析中断原因后，转入缺页中断处理程序。该程序通过查找页表，得到该页在外存的物理块后，如果此时内存能容纳新页，则启动磁盘I/O，将所缺之页调入内存，然后修改页表。如果内存已满，则须先按照某种置换算法，从内存中选出一页准备换出；如果该页未被修改过（修改位为“0”），可不必将该页写回磁盘；但如果此页已被修改（修改位为“1”），则必须将它写回磁盘，然后再把所缺的页调入内存，并修改页表中的相应表项，置其存在位为“1”，并将此页表项写入快表中。在缺页调入内存后，利用修改后的页表，去形成所要

访问数据的物理地址，再去访问内存数据。整个页面的调入过程对用户是透明的。

12. 在请求分页系统中，常采用哪几种页面置换算法？

答：采用的页面置换算法有：最佳置换算法和先进先出置换算法，最近最久未使用（LRU）置换算法，Clock 置换算法，最少使用置换算法，页面缓冲算法等。

13. 在一个请求分页系统中，采用 FIFO 页面置换算法时，假如一个作业的页面走向为 4、3、2、1、4、3、5、4、3、2、1、5，当分配给该作业的物理块数 M 分别为 3 和 4 时，试计算在访问过程中所发生的缺页次数和缺页率，并比较所得结果。

	4	3	2	1	4	3	5	4	3	2	1	5
M=3	4	4	4	1	1	1	5			5	5	
		3	3	3	4	4	4			2	2	
			2	2	2	3	3			3	1	
M=4	4	4	4	4			5	5	5	5	1	1
		3	3	3			3	4	4	4	4	5
			2	2			2	2	3	3	3	3
				1			1	1	1	2	2	2

M=3 时，采用 FIFO 页面置换算法的缺页次数为 9 次，缺页率为 75%；

M=4 时，采用 FIFO 页面置换算法的缺页次数为 10 次，缺页率为 83%。

由此可见，增加分配给作业的内存块数，反而增加了缺页次数，提高了缺页率，这种现象被称为是 Belady 现象。

14. 实现 LRU 算法所需的硬件支持是什么？

答：需要寄存器和栈等硬件支持。寄存器用于记录某进程在内存中各页的使用情况，栈用于保存当前使用的各个页面的页面号。

15. 试说明改进型 Clock 置换算法的基本原理。

答：因为修改过的页面在换出时付出的开销比未被修改过的页面大，在改进型 Clock 算法中，既考虑页面的使用情况，还要增加置换代价的因素；在选择页面作为淘汰页面时，把同时满足未使用过和未被修改作为首选淘汰页面。

16. 影响页面换进换出效率的若干因素是什么？

(1) **页面置换算法**：影响页面换进换出效率最重要的因素，直接影响进程在运行过程中的缺页率，影响页面换进换出的开销。

(2) **写回磁盘的频率**：如果是采取每个页面换出时，就将它写回磁盘的策略，这意味着每换出一个页面，便需要启动一次磁盘。但如果在系统中建立了一个已修改换出页面链表，对每一个要被换出的页面（已修改），系统可暂不把它们写回磁盘，而是将它们挂在已修改换出页面链表上，仅当被换出页面数目达到一定值时，再将它们一起写回到磁盘上，这样就显著地减少了磁盘I/O的操作次数。

或者说，减少已修改页面换出的开销。

(3) **读入内存的频率**：在设置了已修改换出页面链表后，在该链表上就暂时有一批装有数据的页面，如果需要再次访问这些页面时，就不需从外存上调入，而直接从已修改换出页面链表中获取，这样也可以减少将页面从磁盘读入内存的频率，减少页面换进的开销。或者说，只需花费很小的开销，便可使这些页面，又回到该进程的驻留集中。

17. 页面缓冲算法的主要特点是什么？它是如何降低页面换进换出的频率的？

①显著地降低了页面换进、换出的频率，使磁盘I/O的操作次数大为减少，因而减少了页面换进、换出的开销；

②由于换入换出的开销大幅度减小，才能使其采用一种较简单的置换策略，如先进先出（FIFO）算法，它不需要特殊硬件的支持，实现起来非常简单。

在该系统中，内存分配策略上采用了可变分配和局部置换方式。为了能显著地降低了页面换进、换出的频率，在内存中设置了如下两个链表：

(1) **空闲页面链表**：是一个空闲物理块链表，用于分配给频繁发生缺页的进程，以降低该进程的缺页率。当有一个未被修改的页要换出时，实际上并不将它换出到外存，而是把它们所在的物理块，挂在空闲链表的末尾。

(2) **修改页面链表**：由已修改的页面所形成的链表。设置该链表的目的，是为了减少已修改页面换出的次数。降低将已修改页面写回磁盘的频率，以及降低将磁盘内容读入内存的频率。

18. 什么是抖动？产生抖动的原因是什么？

a. 抖动(Thrashing)就是指当内存中已无空闲空间而又发生缺页中断时，需要从内存中调出一页程序或数据送磁盘的对换区中，如果算法不适当，刚被换出的页很快被访问，需重新调入，因此需再选一页调出，而此时被换出的页很快又要被访问，因而又需将它调入，如此频繁更换页面，使得系统把大部分时间用在了页面的调进换出上，而几乎不能完成任何有效的工作，我们称这种现象为“抖动”。  
b. 产生抖动的原因是由于 CPU 的利用率和多道程序度的对立统一矛盾关系引起的，为了提高 CPU 利用率，可提高多道程序度，但单纯提高多道程序度又会造成缺页率的急剧上升，导致 CPU 的利用率下降，而系统的调度程序又会为了提高 CPU 利用率而继续提高多道程序度，形成恶性循环，我们称这时的进程是处于“抖动”状态。

19. 何谓工作集？它是基于什么原理确定的？

答：工作集（或驻留集）是指在某段时间间隔内，进程要访问的页面集合。经常被使用的页面需要在工作集中，而长期不被使用的页面要从工作集中被丢弃。为了防止系统出现抖动现象，需要选择合适的工作集大小。工作集模型的原理是：让操作系统跟踪每个进程的工作集，并为进程分配大于其工作集的物理块。如果还有空闲物理块，则可以再调一个进程到内存以增加多道程序数。如果所有工作集之和增加以至于超过了可用物理块的总数，那么操作系统会暂停一个进程，将其页面调出并且将其物理块分配给其他进程，防止出现抖动现象。正确选择工作集的大小，对存储器的利用率和系统吞吐量的提高，都将产生重要影响。

20. 当前可以利用哪几种方法来防止“抖动”？

(1) **采取局部置换策略**      (2) **把工作集算法融入到处理机调度中**

(3) **利用“L=S”准则调节缺页率**      (4) **选择暂停的进程**

21. 试试说明如何利用“ $L=S$ ”准则来调节缺页率，以避免“抖动”的发生？ P173

于1980年Denning提出了“ $L=S$ ”的准则，来调节多道程序度，其中 $L$ 是缺页之间的平均时间， $S$ 是平均缺页服务时间，即用于置换一个页面所需的时间。利用“ $L=S$ ”准则，对于调节缺页率是十分有效的。

22. 为了实现请求分段式存储管理，应在系统中增加配置哪些硬件结构？

答：请求段表机制、缺段中断机制和地址变换机构。

23. 在请求段表机制中，应设置哪些段表项？

段名	段长	段的基址	存取方式	访问字段A	修改位M	存在位P	增补位	外存始址
----	----	------	------	-------	------	------	-----	------

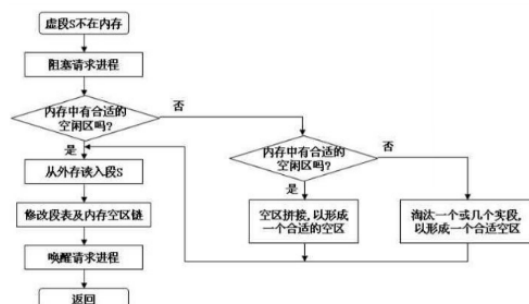
存取方式，访问字段A，修改位M，存在位P，增补位，外存始址。

24. 说明请求分段系统中的缺页中断处理过程。

答：请求分段系统中的缺页中断处理过程描述如下：

- (1) 根据当前执行指令中的逻辑地址查页表，判断该页是否在主存储器中
- (2) 该页标志为“0”形成缺页中断，中断装置通过交换PSW让操作系统的中断处理程序占用处理器。
- (3) 操作系统处理缺页中断处理的办法是查主存分配表找一个空闲的主存块，查页表找出该页在磁盘上位置，启动磁盘读出该页信息。
- (4) 把从磁盘上读出的信息装入找到的主存块中。
- (5) 当页面住处被装入主存后，应修改页表中对应的表目，填上该页所占用的主存块把标志置为“1”，表示该页已在主存储器中
- (6) 由于产生缺页中断时的那条指令并没执行完，所以在把页面装入之后应重新执行被中断指令。

请求分段系统中的缺页中断处理过程如下图所示：



25. 请对共享段表中的各项作简要说明。

在系统中配置一张共享段表，所有各共享段都在共享段表中占有一表项。在表项的上面记录了共享段的段号、段长、内存始址、状态（存在）位、外存始址以及共享计数等信息。接下去就是记录了共享此分段的每个进程的情况。

	段名	段长	内存始址	状态	外存始址
	共享进程计数count				
	状态	进程名	进程号	段号	存取控制
	:	:	:	:	:

共享段表

① 共享进程计数count：记录有多少进程正在共享该分段。

② 存取控制字段：对于一个共享段，应为不同的进程赋予不同的存取权限。

③ 段号：每个进程可用自己进程的段号，去访问该共享段。