

# 王道考研——操作系统

WWW.CSKAOYAN.COM

## 第三章 存储管理

1

历年真题考频统计

章节	索引	核心考点	2009	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019	2020	2021	2022	小题考频	大题考频	章节考频
Ch1	1	操作系统的概念、特征和功能	#23							#23	#28	#23				#23	5		24+3
	2	内核态与用户态			#24	#23	#28	#25	#24		&45				#23	#27	7	1	
	3	中断、异常				#24			#23		#32 &45	#29		#25			5	1	
	4	系统调用		#23							#24					#32	#31	5	
	5	操作系统引导											#25			&46	#24	1	
Ch2	6	进程与线程			#25	#31		#31		#24			#23	#29				6	50+15
	7	进程状态与进程控制		#24 #26				#26	#25			#27	#24		#24	#28	8		
	8	处理机调度	#24		#23	#29 #30	#31	#23		&46	#23 #27	#24	#27	#26	#25 #27	#25	14	1	
	9	进程同步与互斥	&45	#25 #27	#32 &45		&45	&47	&45	#27 #30 #31	&46	#25 #28 #32		#32	&45		10	7	
	10	经典同步问题（今年注意读者-写者）	&45		&45		&45		&45				&43	&45		&46		7	
Ch3	11	死锁	#25		#27	#27	#32	#24	#26	#25		#26	#30	#27	#31	#26	12		27+13
	12	内存管理的概念					#29										2		
	13	连续分配管理方式	#26	#28							#25		#32				4		
	14	非连续分配管理方式	#27 &46	#29 &46				&46	#32	&46	#28	&45	&45	#28 #31	&46		6	7	
	15	虚拟式存储管理	&46	&46	#28 #29	#25 &45	#30	#30	#27 #30	#26 #29	&45	&45	#29	#28 &46	#28 #29	#29 #30	15	6	
Ch4	16	文件元数据和索引节点																1	28+16
	17	文件的操作	#31	#31		#28	#23	#29			#31	#31				&45	7	1	
	18	文件的逻辑结构和物理结构	#28	#30	&46	&46	#24 #26	&46	#29	&47		&46	#26	#24		&45	7	6	
	19	文件共享和文件保护	#30								#30			#23			3		
	20	目录结构和管理			&46			#26	&46	&47		&46		#31	#30	&45	3	5	
Ch5	21	磁盘的组织与管理	#29	&45		#32		#27	#31		#26 #29	#30	&44		#26 &46		8	3	12+0
	22	I/O控制方式																	
	23	I/O软件的层次结构		#32	#26	#26	#25		#28 #32							#32	6		
	24	高速缓存与缓冲区			#31												4		
	25	设备分配与回收	#32											#30			2		
	26	SPOOLing 技术																	

王道考研/CSKAOYAN.COM

2

### 第三章（内存管理）命题重点

#### 【命题重点】

1. 程序装入的过程与原理，编译与链接，逻辑地址与物理地址。
2. 连续分配方式的原理与特点，动态分区分配算法的原理。
3. 分页存储管理的页表机制，分段存储管理的段表机制，分页与分段的比较。
4. 虚拟存储器的原理、特征，缺页中断的处理过程和特点，虚拟地址和物理地址的变换，引入快表后的页式虚拟存储器的地址变换过程，各种页面置换算法的原理，页面分配策略，工作集的定义，抖动产生的原因和解决办法。

王道考研/CSKAOYAN.COM

3

### 考点12：内存管理的概念

		• 小题 30		• 小题 29		
2009	2010	2011	2012	2013	2014	2015
2016	2017	2018	2019	2020	2021	2022

历年考频： 小题×2、综合题×0

4

## 操作系统考点12

## 内存管理的概念

王道考研/CSKAOYAN.COM

5

## 内存管理的概念

【考点笔记】程序运行的基本原理

将用户源程序变为可在内存中执行的程序，通常需要以下几个步骤：

- 编译，由编译程序将用户源代码编译成若干个目标模块，每个模块具有各自的逻辑地址空间。
- 链接，由链接程序将上述目标模块，以及所需库函数链接，形成具有完整的逻辑地址空间的装入模块。
- 装入，由装入程序将装入模块装入内存。

而程序执行的过程中，在指令寻址与数据寻址时，CPU 不断地进行从逻辑地址到物理地址的转换。图 3-1 展示了从源程序（C 程序为例）变为在内存中执行的程序的过程。

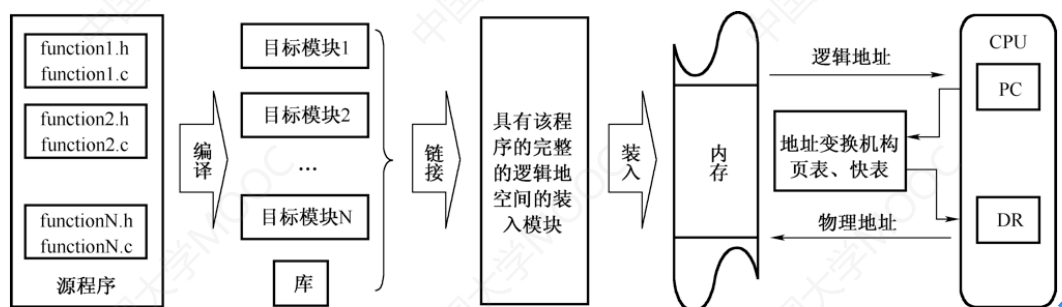


图 3-1 C 语言源程序变为在内存中执行程序的过程

王道考研/CSKAOYAN.COM

6

### 考点13：连续分配管理方式

• 小题 26 2009	• 小题 28 2010	2011	2012	2013	2014	2015
2016	• 小题 25 2017	2018	• 小题 32 2019	2020	2021	2022

历年考频： 小题×4、综合题×0

7

### 操作系统考点13

连续分配管理  
方式

王道考研/CSKAOYAN.COM

8

## 连续分配管理方式

### 【考点笔记】连续分配方式

分配方式	单一连续分配	固定分区分配	动态分区分配
说明	分为系统区和用户区。 系统区仅提供给操作系统使用，通常在低地址部分。 用户区是为用户提供的除系统外的内存空间。	将内存用户空间划分为若干个固定大小的区域，每个分区只装入一道作业。当有空闲分区时，便可以再从外存的后备作业队列中选择适当大小的作业装入该分区，如此循环。	是一种动态划分内存的分区方法。不预先将内存划分，而是在进程装入内存时，根据进程的大小动态地建立分区，并使分区的大小正好适合进程的需要。因此系统中分区的大小和数目是可变的。
碎片	内部碎片	内部碎片	外部碎片
作业道数	1	$\leq N$ (用户空间划为 $N$ 块)	不确定
硬件	界地址寄存器、越界检查机构	上下界寄存器、越界检查机构 基地址寄存器、长度寄存器、动态地址转换机构	
解决空间不足	覆盖	覆盖/交换	交换

王道考研/CSKAOYAN.COM

9

## 连续分配管理方式

### 【考点笔记】动态分区分配算法

算法	说明	特点
首次适应算法	空闲分区以地址递增的次序链接。分配内存时从链首开始顺序查找，直至找到大小能满足要求的第一个空闲分区。	实现方法简单； 查找速度快，平均性能最好； 碎片多出现于低地址空间。
循环首次适应算法	空闲分区以地址递增的次序链接。分配内存时，从上次查找结束的位置开始继续查找，直到找到第一个满足要求的空闲分区。	平均性能比首次适应算法差； 碎片多出现于高地址空间。
最佳适应算法	空闲分区按容量递增的次序链接。分配内存时，从链首开始顺序查找，直到找到第一个能满足要求的空闲分区。	需要对分区排序，开销大； 形成许多难以利用的小碎片。
最差适应算法	空闲分区以容量递减的次序链接。分配内存时，从链首开始顺序查找，直到找到第一个能满足要求的空闲分区，也就是挑选出最大的分区。	需要对分区排序，开销大； 使系统缺少大的连续空闲地址空间。

王道考研/CSKAOYAN.COM

10

### 考点14：非连续分配管理方式

<ul style="list-style-type: none"> <li>• 小题27</li> <li>• 综合题46</li> </ul>	<ul style="list-style-type: none"> <li>• 小题29</li> <li>• 综合题46</li> </ul>			<ul style="list-style-type: none"> <li>• 综合题46</li> </ul>	<ul style="list-style-type: none"> <li>• 小题32</li> </ul>	<ul style="list-style-type: none"> <li>• 综合题46</li> </ul>
2009	2010	2011	2012	2013	2014	2015
<ul style="list-style-type: none"> <li>• 小题28</li> </ul>	<ul style="list-style-type: none"> <li>• 综合题45</li> </ul>	<ul style="list-style-type: none"> <li>• 综合题45</li> </ul>	<ul style="list-style-type: none"> <li>• 小题28</li> <li>• 小题31</li> </ul>	<ul style="list-style-type: none"> <li>• 综合题46</li> </ul>		
2016	2017	2018	2019	2020	2021	2022

历年考频： 小题×6、综合题×7

11

### 操作系统考点14

## 非连续分配管 理方式

王道考研/CSKAOYAN.COM

12



## 非连续分配管理方式

### 【考点笔记】分页存储管理方式

把主存空间划分为大小相等且固定的块，块相对较小，作为主存的基本单位。每个进程也以块为单位进行划分，进程在执行时，以块为单位逐个申请主存中的块空间。进程中的块称为页，内存中的块称为页框（页帧）。外存也以同样单位进行划分，直接称为块。

分页存储管理的逻辑地址结构包含两部分：页号  $P$ ，页内偏移量  $W$ 。如图 3-3 所示。

系统为每个进程都建立一张页表，记录页面在内存中对应的物理块号。如图 3-4 所示。



图 3-3 分页存储管理的地址结构

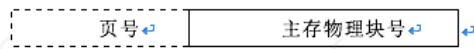


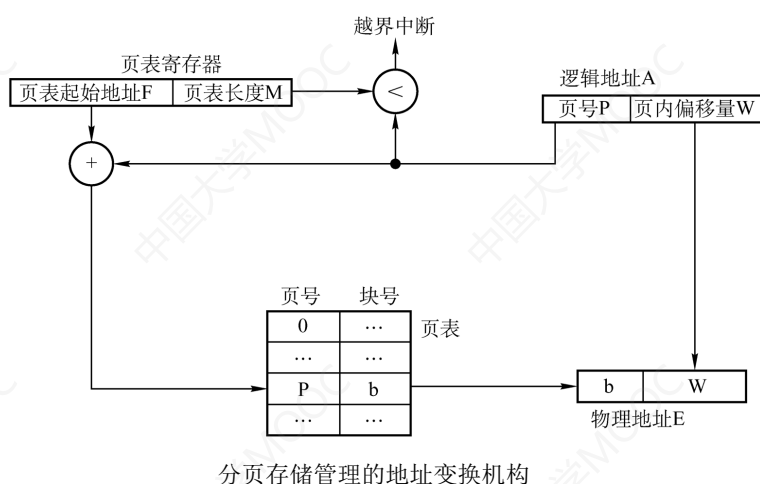
图 3-4 页表项

硬件支持：页表寄存器（PTR），存放页表在内存的始址和页表长度。

王道考研/CSKAOYAN.COM

13

## 非连续分配管理方式



逻辑地址与物理地址的转变过程：

### 1) 查页表

- ① 先将页号  $P$  与页表长度  $M$  比较；若越界，即  $P \geq M$ ，则表示地址越界并中断。
- ② 若未越界，则将页表始址与页号和页表项长度的乘积相加，便得到该表项在页表中的位置，从中得到该页的物理块号，将之装入物理地址寄存器中。

### 2) 根据页表项算出物理地址

将有效地址中的页内偏移量送入物理地址寄存器的块内地址字段中，即可得到要访问的内存物理地址。

王道考研/CSKAOYAN.COM

14

## 非连续分配管理方式

### 【考点笔记】分段存储管理方式

分段存储管理方式的逻辑地址由两部分组成：段号  $S$  与段内偏移量  $W$ ，如图 3-6 所示。段号字段决定了作业最大段数，段内偏移量字段决定了最大段长。

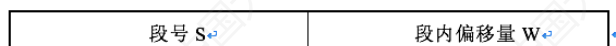


图 3-6 分段系统中的逻辑地址结构

系统为每个进程都建立一张逻辑空间与主存空间映射的段表，每一个段表项对应进程的一个段，段表项记录该段在内存中的起始地址和段的长度。段表的内容如图 3-7 所示。

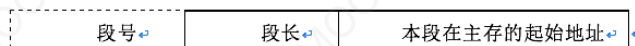


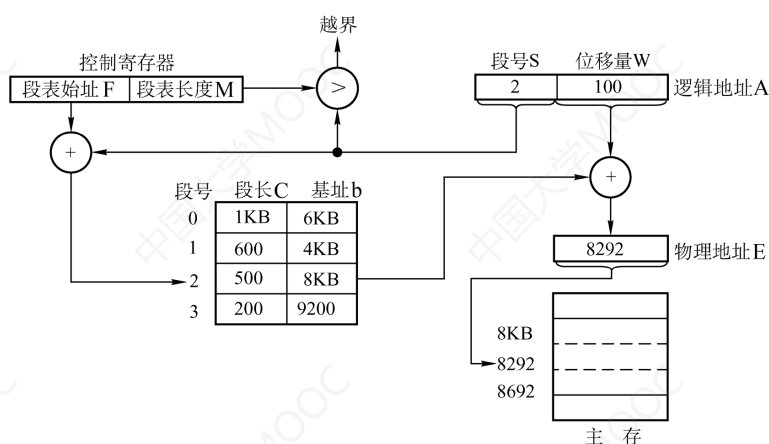
图 3-7 段表项

硬件支持：段表寄存器，用于存放段表始址和段表长度  $TL$ 。

王道考研/CSKAOYAN.COM

15

## 非连续分配管理方式



分段存储管理的地址变换机构

逻辑地址到物理地址的转换过程如下：

### 1) 查段表

系统将逻辑地址中的段号  $S$ ，与段表长度  $TL$  进行比较。

① 若越界，即  $S > TL$ ，表示段号太大，访问越界，于是产生越界中断信号。

② 若未越界，则根据段表的始址和该段的段号，计算出该段对应段表项的位置，从中读出该段在内存中的起始地址。

### 2) 根据段表项算出物理地址

检查段内地址  $W$  是否超过该段的段长  $SL$ 。

① 若超过，即  $W > SL$ ，同样发出越界中断信号。

② 若未越界，则将该段的基址  $d$  与段内地址相加，即可得到要访问的内存物理地址。

王道考研/CSKAOYAN.COM

16



## 非连续分配管理方式

### 【考点笔记】分页式与分段式对比

分页式存储管理方式	分段式存储管理方式
是从计算机的角度考虑设计的，以提高内存的利用率，提升计算机的性能为目标。	考虑了用户关于方便编程、信息保护和共享、动态增长及动态链接等多方面的需要。
在页式系统中，分页通过硬件机制实现，逻辑地址的页号和页内偏移量对用户是透明的。	段号和段内偏移量必须由用户显示提供，在高级程序设计语言中，这个工作由编译程序完成。
逻辑地址是一维结构	逻辑结构是二维结构
产生内部碎片	产生外部碎片
页面大小一致	段长大小不等

王道考研/CSKAOYAN.COM

17

## 考点15：虚拟页式存储管理

• 综合题46 2009	• 综合题46 2010	• 小题28 • 小题29 2011	• 小题25 • 综合题45 2012	• 小题30 2013	• 小题30 2014	• 小题27 • 小题30 2015
• 小题26 • 小题29 2016	• 综合题45 2017	• 综合题45 2018	• 小题29 2019	• 小题28 • 综合题26 2020	• 小题28 • 小题29 2021	• 小题29 • 小题30 2022

历年考频： 小题×15、综合题×6

18

## 操作系统考点15

虚拟页式存储  
管理

王道考研/CSKAOYAN.COM

19

## 虚拟页式存储管理

## 【考点笔记】虚拟存储器的定义和特征

基于局部性原理，在程序装入时，可以将程序的一部分装入内存，而将其余部分留在外存，就可以启动程序执行。在程序执行过程中，当所访问的信息不在内存时，由操作系统将所需要的部分调入内存。另一方面，操作系统将内存中暂时不使用的内容换出到外存上。这样，系统好像为用户提供了一个比实际内存大得多的存储器，称为虚拟存储器。

虚拟存储器有以下三种实现方式：请求分页；请求分段；请求段页式。

虚拟存储器有以下三个主要特征：多次性；对换性；虚拟性。

虚拟存储器的容量取决于地址空间的大小，而不是由实际的内存容量决定。

王道考研/CSKAOYAN.COM

20

## 虚拟页式存储管理

### 【考点笔记】缺页中断机构

在页式虚拟存储管理系统中，程序是部分装入的，还有部分是处于外存上的，因此，当需要访问那部分位于外存上的代码或数据时，系统就会产生缺页中断。

产生缺页中断的目的是要将位于外存上的代码或数据装入内存，此时应将缺页的进程阻塞（调页完成唤醒），如果内存中有空闲块，则分配一个块，将要调入的页装入该块，并修改页表中相应页表项，若此时内存中没有空闲块，则需要淘汰某页。

缺页中断作为中断同样要经历，诸如保护 CPU 环境、分析中断原因、转入缺页中断处理程序、恢复 CPU 环境等几个步骤。但与一般的中断相比，它有以下两个明显的区别：

- 在指令执行期间产生和处理中断信号，而非一条指令执行完后。
- 一条指令在执行期间，可能产生多次缺页中断。

王道考研/CSKAOYAN.COM

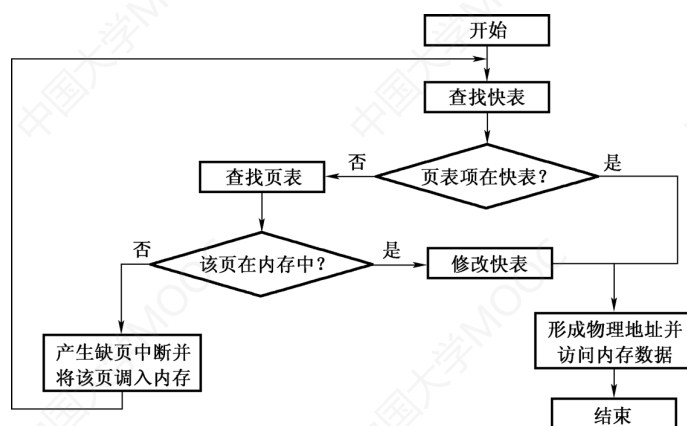
21

## 虚拟页式存储管理

### 【考点笔记】快表

在地址变换机构中增设了一个具有并行查找能力的高速缓冲存储器——快表，又称联想寄存器（TLB），用以存放当前访问的若干页表项。

在具有快表的分页机制中，地址的变换过程如下：



王道考研/CSKAOYAN.COM

22

## 虚拟页式存储管理

### 【考点笔记】请求分页存储管理

请求分页存储管理只要求将当前需要的一部分页面装入内存，便可以启动作业运行。在作业执行过程中，当所要访问的页面不在内存时，再通过请求调页功能将其调入，同时还可以通过页面置换功能将暂时不用的页面换出到外存上，以便腾出内存空间。如图 3-8 所示。

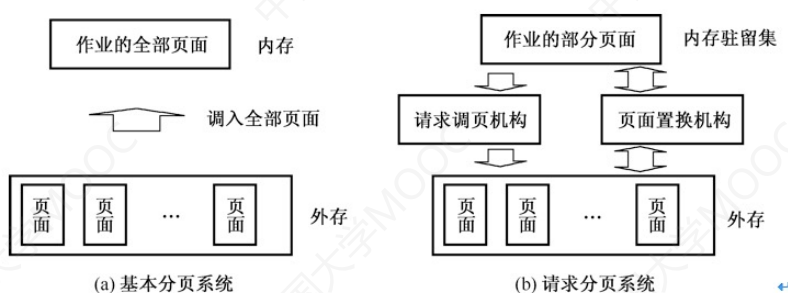


图 3-8 请求分页与基本分页的区别

王道考研/CSKAOYAN.COM

23

## 虚拟页式存储管理

硬件机制	说明						
页表机制	<p>请求分页系统页表项中增加了四个字段，如下图所示。</p> <table><tr><td>页号</td><td>物理块号</td><td>状态位 P</td><td>访问字段 A</td><td>修改位 M</td><td>外存地址</td></tr></table> <p>增加的四个字段说明如下：</p> <ul style="list-style-type: none"><li>● 状态位 P：用于指示该页是否已调入内存。</li><li>● 访问字段 A：用于记录本页在一段时间内被访问的情况。</li><li>● 修改位 M：标识该页在调入内存后是否被修改过。</li><li>● 外存地址：用于指出该页在外存上的地址，通常是物理块号。</li></ul>	页号	物理块号	状态位 P	访问字段 A	修改位 M	外存地址
页号	物理块号	状态位 P	访问字段 A	修改位 M	外存地址		
缺页中断机构	<p>在请求分页系统中，当所要访问的页面状态为 P 无效时，便产生一个缺页中断，此时硬件陷入内核，操作系统将所缺的页调入内存，并更新页表，中断后返回产生缺页中断的那条指令继续执行。</p>						
地址变换机构	<p>① 先根据逻辑地址高位的页号检索快表。</p> <p>② 若找到要访问的页，便修改页表项中的访问位（写指令则还须重置修改位），然后利用页表项中给出的物理块号和页内地址形成物理地址。</p> <p>③ 否则，再到内存中查找页表，找到该页表项后检查状态位 P，看该页是否已调入内存，</p> <p>a) 若已经在内存中，则修改页表项中的访问位和快表，并根据页表项中的物理块号和逻辑地址低位的页内地址形成物理地址。</p> <p>b) 若未调入内存，则产生缺页中断，请求从外存把该页调入内存，并修改页表和快表。然后返回原指令重新执行，再次寻址。</p>						

24

## 虚拟页式存储管理

### 【考点笔记】页面置换算法

假定系统为某进程分配了三个物理块，并考虑有以下页面号引用串：

7, 0, 1, 2, 0, 3, 0, 4, 2, 3, 0, 3, 2, 1, 2, 0, 1, 7, 0, 1

#### 1. 最佳置换算法

置换原则：选择的被淘汰页面将是以后永不使用的，或者是在最长时间内不再被访问的页面。

① 最佳置换算法可以保证获得最低的缺页率。

② 最佳置换算法可以用来评价其他算法。

访问页面	7	0	1	2	0	3	0	4	2	3	0	3	2	1	2	0	1	7	0	1
物理块1	7	7	7	2		2		2			2			2				7		
物理块2		0	0	0		0		4			0			0				0		
物理块3			1	1		3		3			3			1				1		
缺页否	√	√	√	√		√		√			√			√				√		

25

## 虚拟页式存储管理

### 【考点笔记】页面置换算法

假定系统为某进程分配了三个物理块，并考虑有以下页面号引用串：

7, 0, 1, 2, 0, 3, 0, 4, 2, 3, 0, 3, 2, 1, 2, 0, 1, 7, 0, 1

#### 2. 先进先出页面置换 (FIFO) 算法

置换原则：优先淘汰最早进入内存的页面，亦即在内存中驻留时间最久的页面。

① FIFO 算法可能会产生当所分配的物理块数增大而页故障数不减反增的 Belady 异常。

访问页面	7	0	1	2	0	3	0	4	2	3	0	3	2	1	2	0	1	7	0	1
物理块1	7	7	7	2		2	2	4	4	4	0			0	0			7	7	7
物理块2		0	0	0		3	3	3	2	2	2			1	1			1	0	0
物理块3			1	1		1	0	0	0	3	3			3	2			2	2	1
缺页否	√	√	√	√		√	√	√	√	√	√			√	√			√	√	√

26

## 虚拟页式存储管理

### 【考点笔记】页面置换算法

假定系统为某进程分配了三个物理块，并考虑有以下页面号引用串：

7, 0, 1, 2, 0, 3, 0, 4, 2, 3, 0, 3, 2, 1, 2, 0, 1, 7, 0, 1

### 3. 最近最久未使用 (LRU) 置换算法

置换原则：选择最近最长时间未访问过的页面予以淘汰。

访问页面	7	0	1	2	0	3	0	4	2	3	0	3	2	1	2	0	1	7	0	1
物理块1	7	7	7	2		2		4	4	4	0			1		1		1		
物理块2		0	0	0		0		0	0	3	3			3		0		0		
物理块3			1	1		3		3	2	2	2			2		2		7		
缺页否	√	√	√	√		√		√	√	√	√			√		√		√		

27

## 虚拟页式存储管理

### 4. 时钟 (CLOCK) 置换算法

系统为每一帧关联一个附加位，称为使用位。

置换原则：

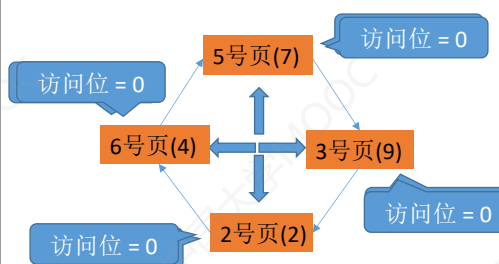
选择一个页面  
最多两轮扫描

① 将候选帧集合看做一个循环缓冲区。

② 当需要替换一页时，从当前指针位置开始查找首个使用位为 0 的帧。查找过程重复以下操作直至找到为止：

- 若帧的使用位为 0，则该帧中的页被替换，其使用位被置为 1，并将指针指向缓冲区中的下一帧。
- 若帧的使用位为 1，操作系统就将该位重新置为 0，并查找缓冲区的下一帧。

例：系统为某个进程依次分配了页框号为“7, 9, 2, 4”总共 4 个页框（进程驻留集大小为 4），刚开始所有页面都没有调入内存。进程按 0→3→2→1→5→3→2→6，的顺序访问页面，采用 CLOCK 算法...



要点1：根据驻留集大小、页面调入内存的先后顺序来确定循环队列的顺序。

要点2：扫描指针初始时指向第一个到达的页面(有的题目会单独说明指针的指向)

要点3：分配给进程的页框不够用时，需要换出一个页面，再将新的页面调入相应的页框中。页面放入后，访问位应设置为 1，且扫描指针指向循环队列中的下一个页面。

28



## 虚拟页式存储管理

改进型的 CLOCK 置换算法

在使用位的基础上再增加一个修改位。

置换原则：

选择一个页面  
最多四轮扫描

- ① 从指针的当前位置开始，扫描帧缓冲区，选择遇到的第一个帧（使用位=0，修改位=0）用于替换。
- ② 如果第 1 步失败，则重新扫描，选择遇到的第一个（使用位=0，修改位=1）帧用于替换。在这个扫描过程中，对每个跳过的帧，把它的使用位设置成 0。
- ③ 如果第 2 步失败，指针将回到它的最初位置，并且集合中所有帧的使用位均为 0。重复第 1 步，并且如果有必要，重复第 2 步。这样将可以找到供替换的帧。

29

## 虚拟页式存储管理

【考点笔记】页面分配策略

**固定分配局部置换。**它为每个进程分配一定数量的物理块，在整个运行期间都不改变。若进程在运行中发现缺页，则只能从该进程在内存的页面中选出一个换出，然后再调入需要的页面。实现这种策略难以确定为每个进程应分配的物理块数量：太少会频繁出现缺页中断，太多又会使 CPU 和其他资源利用率下降。

**可变分配全局置换。**这是最易于实现的物理块分配和置换策略，为系统中的每个进程分配一定数量的物理块，操作系统自身也保持一个空闲物理块队列。当某进程发现缺页时，系统从空闲物理块队列中取出物理块分配给该进程，并将欲调入的页装入其中。

**可变分配局部置换。**它为每个进程分配一定数目的物理块，当某进程发现缺页时，只允许从该进程在内存的页面中选出一页换出，这样就不会影响其他进程的运行。如果进程在运行中频繁地缺页，系统需再为该进程分配若干附加物理块，直至该进程缺页率趋于适当程度为止；反之，若一个进程在运行过程中缺页率特别低，则此时可适当减少分配给该进程的物理块。

30

## 虚拟页式存储管理

### 【考点笔记】工作集（驻留集）

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驻留集中，而长期不被使用的页面要从驻留集中被丢弃。驻留集模型使用较为简单：操作系统跟踪每个进程的驻留集，并为进程分配大于其驻留集的空间。如果还有空闲，那么可启动另一个进程。如果所有驻留集之和增加以至于超过了可用物理块的总数，那么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。

正确选择驻留集的大小，对存储器的有效利用和系统吞吐量的提高，都将产生重要的影响。

31

## 抖动

### 【考点笔记】抖动现象

在进程的页面置换过程中，频繁的页面调度行为称为抖动或颠簸。如果一个进程在换页上用的时间多于执行时间，那么这个进程就在颠簸。

使用虚拟内存技术，操作系统中进程通常只有一部分块位于主存中，从而可以在内存中保留更多的进程以提高系统效率。产生抖动后，使得系统吞吐量下降，缺页中断率迅速增长，由于处理机差不多花费所有时间在进行内外存页面置换，因此几乎不能做任何工作。

引起系统抖动的原因是对换的信息量过大、内存容量不足、置换算法选择不当，所以解决的办法就是降低交换页面的数量、加大内存容量、改变置换选择算法。但是降低交换页面数量和改变置换选择算法对于一个应用系统来讲是不可能的，只能增加内存容量，或者降低进程数量（相对地增加内存）。而增加交换区容量并不能解决物理内存不足的问题。

王道考研/CSKAOYAN.COM

32