

第四章习题

问题 7

为什么要引入动态重定位？如何实现？

在程序执行过程中，每当访问指令或数据时，将要访问的程序或数据的逻辑地址转换成物理地址，即动态重定位；动态运行时的装入程序在把装入模块装入内存后，并不立即把装入模块中的逻辑地址转换为物理地址，而是把这种地址转换推迟到程序真正要执行时才进行。

实现方法可以在系统中增加一个重定位寄存器，用来装入程序在内存中的起始地址，程序执行时，真正访问的内存地址是相对地址与重定位寄存器中的地址相加之和，从而实现动态重定位。

问题 11

令 $buddy_k(x)$ 为大小为 2^k 、地址为 x 的块的伙伴系统地址，试写出 $buddy_k(x)$ 的通用表达式。

$$buddy_k(x) = x + 2^k \quad (x \bmod 2^{k+1} = 0)$$

$$buddy_k(x) = x - 2^k \quad (x \bmod 2^{k+1} = 2^k)$$

使用了模运算（mod）来判断 x 是否为 2^{k+1} 的倍数。如果是的话，那么伙伴块的地址就是 $x + 2^k$ ；如果不是的话，那么伙伴块的地址就是 $x - 2^k$ 。

问题 22

具有快表时是如何实现地址变换的？

地址变换是将逻辑地址转换为物理地址的过程。这通常是通过使用段页式存储管理方式中的段表和页表来实现的。

在 CPU 给出有效地址后，由地址变换机构自动将页号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。

第五章习题

问题 4

实现虚拟存储器需要哪些硬件支持？

1. 请求分页（段）的页（段）表机制。
2. 缺页（段）终端机构。
3. 地址变换机构

问题 5

实现虚拟存储器需要那几个关键技术？

1. 如果使用分页请求系统，需要在分页系统的基础上，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. 如果使用分段请求系统，需要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。

问题 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，是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。

当 $M = 3$ 时，情况如表 1 所示，缺页次数为 9，缺页率为 75%。

表 1: FIFO 置换算法 ($M = 3$)

访问页面	4	3	2	1	4	3	5	4	3	2	1	5
物理块 1	4	4	4	1	1	1	5			5	5	
物理块 2		3	3	3	4	4	4			2	2	
物理块 3			2	2	2	3	3			3	1	
是否缺页	✓	✓	✓	✓	✓	✓	✓			✓	✓	

当 $M = 4$ 时，情况如表 2 所示，缺页次数为 10，缺页率为 83.3%。

表 2: FIFO 置换算法 ($M = 4$)

访问页面	4	3	2	1	4	3	5	4	3	2	1	5
物理块 1	4	4	4	1			5	5	5	5	1	1
物理块 2		3	3	3			3	4	4	4	4	5
物理块 3			2	2			2	2	3	3	3	3
物理块 4				1			1	1	1	2	2	2
是否缺页	✓	✓	✓	✓			✓	✓	✓	✓	✓	✓

对比发现，此处使用 FIFO 算法增大 M 后，缺页次数与缺页率反而增大。

补充练习

补充练习 1

给定内存空闲分区，按地址从小到大为:100K、500K、200K、300K 和 600K。现有用户进程一次分别为 212K、417K、112K 和 426K。回答以下问题:

- (1) 分别用首次适应算法、循环首次适应算法、最佳适应算法和最坏适应算法将它们装入到内存的那个分区?
- (2) 哪些算法能适合该进程序列?

对于该进程，最佳适应算法更适合，对于 426K 的请求成功分配。具体情况如表 3 所示。

表 3: 基于顺序搜索的动态分区算法分配算法对比

地址由低到高排序	分区长度	首次适应算法	循环首次适应算法	最佳适应算法	最坏适应算法
1	100K				
2	500K	212K 112K	212K 112K	417K	417K
3	200K			112K	
4	300K			212K	
5	600K	417K	417K	426K	212K 112K
		426K 无法满足	426K 无法满足		426K 无法满足

补充练习 3

某系统采用页式存储管理策略，拥有逻辑空间 32 页，每页 2K，拥有物理空间 1M。

- (1) 写出逻辑地址的格式。
- (2) 若不考虑访问权限等，进程的页表有多少项? 每项至少有多少位?
- (3) 如果物理空间减少一半，页表结构应相应作怎样的改变?

- (1) 逻辑空间有 32 页，则页号有 5 页， $2^5 = 32$ ；每页 2KB，则页内地址有 11 位，逻辑地址格式为:

页号 (15-11)	页内地址 (10-0)
------------	-------------

- (2) 进程页表最多有 32 个表项，1MB 的物理空间可以分为 $2^{20}/2^{11} = 2^9$ 个内存块，则每个页表项至少 9 位。
- (3) 项数不变，每一项长度减去 1。

补充练习 2

已知某分页系统，主存容量为 64K，页面大小为 1K，对一个 4 页大的作业，其 0、1、2、3 页分别被分配到主存的 2、4、6、7 块中。

- (1) 将十进制的逻辑地址 1023、2500、3500、4500 转换成物理地址。
- (2) 以十进制的逻辑地址 1023 为例画出地址变换过程图。

- (1) 1023:

$1023/1024 = 0$ ，即在第 0 页，第 2 块。

$1023\%1024 = 1023$ 物理地址为 $2 * 1024 + 1023 = 3071$ 。

- (2) 2500:
 $2500/1024 = 2$ ，即在第 2 页，第 6 块。
 $2500\%1024 = 452$ 物理地址为 $6 * 1024 + 452 = 6596$ 。
- (3) 3500:
 $3500/1024 = 3$ ，即在第 3 页，第 7 块。
 $3500\%1024 = 428$ 物理地址为 $7 * 1024 + 428 = 7596$ 。
- (4) 4500:
 $4500/1024 = 4$ ，越界
- 十进制逻辑地址 1023 的地址变换过程图如图 1 所示。

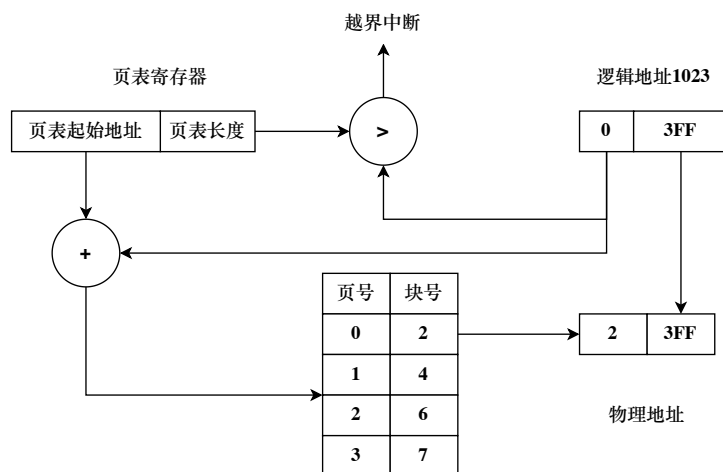


图 1: 地址变换过程图

补充练习 4

某段式存储管理系统中，有一作业的段表如下表所示，求逻辑地址 [0, 65], [1, 55], [2, 90], [3, 20] 对应的主存地址 (按十进制)。(其中方括号中的第一个元素为段号，第二个元素为段内地址)

段号	段长 (容量)	主存起始地址	状态
0	200	600	1
1	50	850	1
2	100	1000	1
3	150	—	0

- (1) [0,65]:
 $65 < 200$ ，主存地址为 $600 + 65 = 665$ 。
- (2) [1,55]:
 $55 > 50$ ，产生段地址越界中断。
- (3) [2,90]:
 $90 < 1000$ ，主存地址为 $1000 + 90 = 1090$ 。
- (4) [3,20]:
 状态为 0，此段在外存中，缺页中断。

补充练习 5

某进程页面走向为:1, 2, 3, 4, 2, 1, 5, 6, 2, 1, 2, 3, 7, 6, 3, 2, 1, 2, 3, 6。系统为该进程分配块的大小分别为 3 和 4 时, 试计算 FIFO, LRU 和 OPT 页面淘汰算法的缺页中断数及缺页中断率各是多少?

FIFO 置换算法 ($M = 3$)

访问页面	1	2	3	4	2	1	5	6	2	1	2	3	7	6	3	2	1	2	3	6
物理块 1	1	1	1	4		4	4	6	6	6		3	3	3		2	2		2	6
物理块 2		2	2	2		1	1	1	2	2		2	7	7		7	1		1	1
物理块 3			3	3		3	5	5	5	1		1	1	6		6	6		3	3
是否缺页	✓	✓	✓	✓		✓	✓	✓	✓	✓		✓	✓	✓		✓	✓		✓	✓

缺页次数为 16, 缺页率为 80%。

LRU 置换算法 ($M = 3$)

访问页面	1	2	3	4	2	1	5	6	2	1	2	3	7	6	3	2	1	2	3	6
物理块 1	1	1	1	4		4	5	5	5	1		1	7	7		2	2			2
物理块 2		2	2	2		2	2	6	6	6		3	3	3		3	3			3
物理块 3			3	3		1	1	1	2	2		2	2	6		6	1			6
是否缺页	✓	✓	✓	✓		✓	✓	✓	✓	✓		✓	✓	✓		✓	✓			✓

缺页次数为 15, 缺页率为 75%。

OPT 置换算法 ($M = 3$)

访问页面	1	2	3	4	2	1	5	6	2	1	2	3	7	6	3	2	1	2	3	6
物理块 1	1	1	1	1			1	1				3	3			3	3			6
物理块 2		2	2	2			2	2				2	7			2	2			2
物理块 3			3	4			5	6				6	6			6	1			1
是否缺页	✓	✓	✓	✓			✓	✓				✓	✓			✓	✓			✓

缺页次数为 11, 缺页率为 55%。

FIFO 置换算法 ($M = 4$)

访问页面	1	2	3	4	2	1	5	6	2	1	2	3	7	6	3	2	1	2	3	6
物理块 1	1	1	1	4			1	1				1	1	6			6			
物理块 2		2	2	2			2	2				2	2	2			2			
物理块 3			3	3			5	5				3	3	3			3			
物理块 4				4			4	6				6	7	7			1			
是否缺页	✓	✓	✓	✓			✓	✓				✓	✓	✓			✓			

缺页次数为 6，缺页率为 30%。

LRU 置换算法 ($M = 4$)

访问页面	1	2	3	4	2	1	5	6	2	1	2	3	7	6	3	2	1	2	3	6
物理块 1	1	1	1	4			5	5	5	5		3	3	3		3	1		1	
物理块 2		2	2	2			2	6	6	6		6	7	7		7	7		3	
物理块 3			3	3			3	3	2	2		2	2	6		6	6		6	
物理块 4				4			4	4	4	1		1	1	1		2	2		2	
是否缺页	✓	✓	✓	✓			✓	✓	✓	✓		✓	✓	✓		✓	✓		✓	

缺页次数为 10，缺页率为 50%。

OPT 置换算法 ($M = 4$)

访问页面	1	2	3	4	2	1	5	6	2	1	2	3	7	6	3	2	1	2	3	6
物理块 1	1	1	1	1			1	1					1	1						
物理块 2		2	2	2			2	2					2	2						
物理块 3			3	3			3	3					3	3						
物理块 4				4			5	6					7	6						
是否缺页	✓	✓	✓	✓			✓	✓					✓	✓						

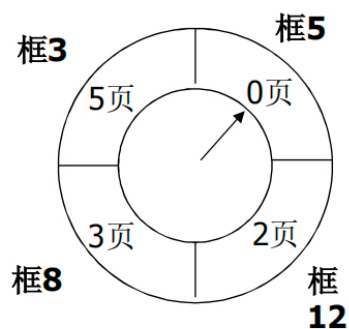
缺页次数为 4，缺页率为 20%。

补充练习 6

某虚拟页式系统，进程空间和内存空间都是 64k，页长 1K，某进程 6 个页，内存分配 4 个页框，采用局部置换，280 时刻页表和 Clock 数据结构如下：

- (1) 280 时刻访问 13B7H，逻辑页号是多少？
- (2) 采用 FIFO 置换算法，物理页框号是多少？物理地址是多少？
- (3) 采用 CLOCK 置换算法，页框号是多少？物理地址是多少？

逻辑页号	页框号	装入时间	访问标志
0	5	110	1
1	-	-	-
2	12	160	1
3	8	230	1
4	-	-	-
5	3	80	1



- (1) 13B7H \longleftrightarrow 0001 0011 1011 0111，前六位 000100 为逻辑页号，对应十进制 4。
- (2) 4 号页发生缺页中断，置换第 5 页，物理页框号为 3，物理地址为 0FB7H。
- (3) 淘汰第 0 页，页框号为 5，物理地址为 17B7H。