

第四章 应用题参考答案

布置作业

第四章（应用题部分） 5, 7, 8, 17, 19, 24, 32, 41, 42, 47。

- 5 给定主存空闲分区，按地址从小到大为：100K、500K、200K、300K 和 600K。现有用户进程依次分别为 212K、417K、112K 和 426K，(1)分别用 first-fit、best-fit 和 worst-fit 算法将它们装入到主存的哪个分区？(2) 哪个算法能最有效利用主存？

答：按题意地址从小到大进行分区如图所示。

分区号	分区长
1	100KB
2	500KB
3	200KB
4	300KB
5	600KB

- (1) 1)first-fit 212KB 选中分区 2，这时分区 2 还剩 288KB。417KB 选中分区 5，这时分区 5 还剩 183KB。112KB 选中分区 2，这时分区 2 还剩 176KB。426KB 无分区能满足，应该等待。

2)best-fit 212KB 选中分区 4，这时分区 4 还剩 88KB。417KB 选中分区 2，这时分区 2 还剩 83KB。112KB 选中分区 3，这时分区 3 还剩 88KB。426KB 选中分区 5，这时分区 5 还剩 174KB。

3)worst-fit 212KB 选中分区 5，这时分区 5 还剩 388KB。417KB 选中分区 2，这时分区 2 还剩 83KB。112KB 选中分区 5，这时分区 5 还剩 176KB。426KB 无分区能满足，应该等待。

- (2) 对于该作业序列，best-fit 算法能最有效利用主存

- 7 一进程以下列次序访问 5 个页：A、B、C、D、A、B、E、A、B、C、D、E；假定使用 FIFO 替换算法，在主存有 3 个和 4 个空闲页框的情况下，分别给出页面替换次数。

答：页面替换序列如下：

FIFO	A	B	C	D	A	B	E	A	B	C	D	E
	A	A	A	D	D	D	E	E	E	E	E	E
		B	B	B	A	A	A	A	A	C	C	C
			C	C	C	B	B	B	B	B	D	D
是否缺页	是	是	是	是	是	是	是			是	是	

FIFO	A	B	C	D	A	B	E	A	B	C	D	E
	A	A	A	A	A	A	E	E	E	E	D	D
		B	B	B	B	B	B	A	A	A	A	E
			C	C	C	C	C	C	B	B	B	B
				D	D	D	D	D	D	C	C	C
是否缺页	是	是	是	是			是	是	是	是	是	是

主存有 3 个和 4 个空闲页框的情况下, 页面替换次数为 9 次和 10 次。出现了 Belady 现象, 增加分给作业的主存块数, 反使缺页中断率上升。

8 某计算机有缓存、主存、辅存来实现虚拟存储器。如果数据在缓存中, 访问它需要 A ns; 如果在主存但不在缓存, 需要 B ns 将其装入缓存, 然后才能访问; 如果不在主存而在辅存, 需要 C ns 将其读入主存, 然后, 用 B ns 再读入缓存, 然后才能访问。假设缓存命中率为 $(n-1)/n$, 主存命中率为 $(m-1)/m$, 则数据平均访问时间是多少?

答:

数据在缓存中的比率为: $(n-1)/n$

数据在主存中的比率为: $(1-(n-1)/n) \times (m-1)/m = (m-1)/nm$

数据在辅存中的比率为: $(1-(n-1)/n) \times (1-(m-1)/m) = 1/nm$

故数据平均访问时间是 $= ((n-1)/n) \times A + ((1-(n-1)/n) \times (m-1)/m) \times (A+B) + ((1-(n-1)/n) \times (1-(m-1)/m)) \times (A+B+C) = A+B/n+C/nm$

【说明】该题目的已知条件不够明确, 容易产生歧义, 不同的理解, 会形成不同的结果。这里缓存命中率为 $(n-1)/n$ 是指全局性的, 即在缓存的概率为 $(n-1)/n$, 那么不在缓存的概率为 $(1-(n-1)/n)$, 而不在缓存分为两种情形, 一种是在不在缓存在主存命中, 其条件概率是 $(m-1)/m$, 这个概率是相对的, 不是全局的, 因此从全局看, 在主存命中的概率为 $(1-(n-1)/n) \times (m-1)/m = (m-1)/nm$; 另一种情形是不在缓存且主存没有命中(在辅存), 从全局看, 该概率为 $(1-(n-1)/n) \times (1-(m-1)/m) = 1/nm$ 。这个解确保三个概率合起来为 100%。

【可能答案】可能有同学的结果是: 第 4 章的第 8 题, 其中关于数据在辅存中的概率, 缓存是不是在主存中的? 答案中是用不在缓存中的概率和不在主存中的概率相乘, 即, $(1-(n-1)/n) \times (1-(m-1)/m)$, 如果缓存是在主存中的, 为什么不是直接用 $1-(m-1)/m$ 来表示在辅存中的概率呢? **【如果给出这样的答案也算正确】**

【解释】如果用 $1-(m-1)/m$ 来表示在辅存的概率, 那么意味着将主存命中率为 $(m-1)/m$ 理解为全局性的, 其包含了在缓存和不在缓存在主存两种情形, 且如果将缓存命中率 $(n-1)/n$ 看作全局性的, 那么: (1) 在缓存的概率为 $(n-1)/n$; (2) 从全局看, 不在缓存在主存概率为 $(m-1)/m - (n-1)/n$, 当然要求保证 $(m-1)/m > (n-1)/n$ 。这样, 确保三个概率合起来为 100%。

17 一台机器有 48 位虚地址和 32 位物理地址, 若页长为 8KB, 问页表共有多少个页表项? 如果设计一个反置页表, 则有多少个页表项?

答: 因为页长 8KB 占用 13 位, 所以, 页表项有 2^{35} 个。

反置页表项有 2^{19} 个。

19 有一个分页虚存系统, 测得 CPU 和磁盘的利用率如下, 试指出每种情况下的存在问题和可采取的措施: (1)CPU 利用率为 13%, 磁盘利用率为 97% (2)CPU 利用率为 87%, 磁盘利用率为 3% (3)CPU 利用率为 13%, 磁盘利用率为 3%。

答: (1)系统可能出现抖动, 可把暂停部分进程运行。(2)系统运行正常, 可增加运行进程数以进一步提高资源利用率。(3)处理器和设备利用率均很低, 可增加并发运行的进程数。

24 在某页式虚存系统中, 假定访问主存的时间是 2ms, 平均缺页中断处理时间为 25ms, 均缺页中断率为 5%, 试计算在该虚存系统中, 平均有效访问时间是多少?

答: 若被访问的页面在主存中, 则一次访问的时间为, $2\text{ms} + 2\text{ms} = 4\text{ms}$; 如果不在主存, 所花的时间是 2ms (访问主存页表) + 25ms (中断处理) + 2ms (访问主存页表) + 2ms (访问主存) = 31ms 。

根据上述分析, 平均有效访问时间是:

$$4\text{ms} \times (1-5\%) + 31\text{ms} \times 5\% = 5.35\text{ms}$$

32 假设计算机有 2M 主存, 其中, 操作系统占用 512K, 每个用户程序也使用 512K 主存。如果所有程序都有 70% 的 I/O 等待时间, 那么, 再增加 1M 主存, 吞吐率增加多少? 此处题目问【吞吐率增加多少?】应该改为【CPU 利用率增加多少?】

答: 由题意可知, 主存中可以存放 3 个用户进程, 而 CPU 的利用率为: $1-(70\%)^3 = 1-(0.7)^3 = 65.7\%$ 。再增加 1M 主存, 可增加 2 个用户进程, 这时 CPU 的利用率为: $1-(70\%)^5 = 1-(0.7)^5 = 83.2\%$ 。故再增加 1M 主存, 吞吐率增加了: $83.2\% \div 65.7\% - 100\% = 27\%$ 。

【如果】有同学计算出 $83.2\% - 65.7\% = 17.5\%$ 【也算正确】。

41 采用 LRU 置换算法的虚拟分页存储管理系统, 其页面尺寸为 4KB, 主存访速度为 100ns, 快表访问速度为 20ns, 缺页中断处理耗时为 25ms。今有一个长度为 30KB 的进程 P 进入系统, 分配给 P 的页框有 3 块, 进程的所有页面都在运行中动态装入。若 P 访问快表的命中率为 20%, 对于下述页面号访问序列:

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

试计算平均有效访存时间为多少 ns?

答:

分页机制中, 系统需从页表中获得指定页的页框号, 而页表的一部分被存储在快表中, 所以每访问一次主存中的数据, 需要先访问一次快表, 如果在快表中查不到指定页时再访问主存中的页表。

1) 系统不缺页的时间花费。

如果要访问的页已经在快表中, 系统只需要花费 20ns 的快表访问时间和 100ns 访问主存就可以了。如果没有命中, 系统还需要访问两次主存。第 1 次是访问主存中的页表, 第 2 次是访问主存中的数据。根据快表的命中率为 20% 的已知条件, 不缺页的有效访问时间 m_a 是:

$$ma=120 \times 20\% + 220 \times 80\% = 200 \text{ (ns)}$$

2) 计算缺页率。

应用程序长度为 30KB, 按每页 4KB 计算共计 8 个页面 (0#~7#)。按 LRU 算法可以得出缺页达 12 次。对于共计 20 次页面访问来说, 缺页率 $p=60\%$ 。

3) 计算平均有效访问时间。

平均有效访问时间 T 的计算公式由两部分组成:

平均有效访问时间 $T = (1-p)ma + p \times \text{缺页中断耗时}$

填入本题中的已知条件后, 得:

$$\begin{aligned} T &= (1-p) \times ma + p \times 25 \text{ (ms)} \\ &= 0.4 \times 200 \text{ (ns)} + 0.6 \times 25000000 \\ &= 15000080 \text{ (ns)}. \end{aligned}$$

42 在请求分页虚存管理系统中, 若驻留集为 m 个页框, 页框初始为空, 在长为 p 的引用串中具有 n 个不同页面 ($n > m$), 对于 FIFO、LRU 两种页面替换算法, 试给出缺页中断的上限和下限, 并举例说明。

答: 对于 FIFO、LRU 两种页面替换算法, 缺页中断的上限和下限: 为 p 和 n 。因为有 n 个不同页面, 无论怎样安排, 不同页面进入主存至少要产生一次缺页中断, 故下限为 n 次。由于 $m < n$, 引用串中有些页可能进入主存后又被调出, 而多次发生缺页中断。极端情况, 访问的页都不在主存, 这样共发生了 p 次缺页中断。例如, 当 $m=3, p=12, n=4$ 时, 有如下访问中: 1, 1, 1, 2, 2, 3, 3, 3, 4, 4, 4, 4。缺页中断为下限 4 次。而访问串: 2, 3, 4, 1, 2, 3, 4, 1, 2, 3, 4, 1。缺页中断为上限 12 次。

45 有两台计算机 P1 和 P2, 它们各有一个硬件高速缓冲存储器 C1 和 C2, 且各有一个主存储器 M1 和 M2。其性能为:

	C1	C2	M1	M2
存储容量	4KB	4KB	2MB	2MB
存取周期	60ns	80ns	1 μ s	0.9 μ s

若两台机器指令系统相同, 它们的指令执行时间与存储器的平均存取周期成正比。如果在执行某个程序时, 所需指令或数据在高速缓冲存储器中存取到的概率 P 是 0.7, 试问: 这两台计算机哪个速度快? 当 $P=0.9$ 时, 处理器哪个速度快?

答: CPU 平均存取时间为: $T = p \times T_1 + (1-p) \times T_2$, T_1 为高速缓冲存储器存取周期, T_2 为主存储器存取周期, p 为高速缓冲存储器命中率。

(1) 当 $p=0.7$ 时,

P1 平均存取时间为: $0.7 \times 60 + (1-0.7) \times 1 \mu s = 342 \text{ ns}$

P2 平均存取时间为: $0.7 \times 80 + (1-0.7) \times 0.9 \mu s = 326 \text{ ns}$

故计算机 P2 比 P1 处理速度快。

(2) 当 $p=0.9$ 时,

P1 平均存取时间为: $0.9 \times 60 + (1-0.9) \times 1 \mu s = 154 \text{ ns}$

P2 平均存取时间为: $0.9 \times 80 + (1-0.9) \times 0.9 \mu s = 162 \text{ ns}$

故计算机 P1 比 P2 处理速度快。

47 假设一个物理存储器, 有 4 个页框, 对下面每种策略, 给出引用串:

p1、p2、p3、p1、p4、p5、p1、p2、p1、p4、p5、p3、p4、p5

的缺页数目(所有页框最初都是空的)。试用下列算法求出缺页中断次数, (1)OPT, (2)FIFO (3)SCR, (4)改进的 CLOCK, (5)LRU, (6)MIN, (7)WS。

解:

(1) 最优置换算法 OPT

F	F	F		F	F(3)						F(1)		
1	2	3	1	4	5	1	2	1	4	5	3	4	5
1	1	1	1	1	1	1	1	1	1	1	3	3	3
	2	2	2	2	2	2	2	2	2	2	2	2	2
		3	3	3	5	5	5	5	5	5	5	5	5
				4	4	4	4	4	4	4	4	4	4

缺页 6 次。

(2) 先进先出算法 FIFO

F	F	F		F	F(1)	F(2)	F(3)				F(4)	F(5)	F(1)
1	2	3	1	4	5	1	2	1	4	5	3	4	5
1	1	1	1	1	2	3	4	4	4	4	5	1	2
	2	2	2	2	3	4	5	5	5	5	1	2	3
		3	3	3	4	5	1	1	1	1	2	3	4
				4	5	1	2	2	2	2	3	4	5

缺页 10 次。

(3) 第二次机会算法 SCR

图中()中为引用位

F	F	F		F	F(1)	F(2)	F(3)				F(4)	F(5)	F(1)
1	2	3	1	4	5	1	2	1	4	5	3	4	5
1(1)	1(1)	1(1)	1(1)	1(1)	2(0)	3(0)	4(0)	4(0)	4(1)	4(1)	5(0)	1(0)	2(0)
	2(1)	2(1)	2(1)	2(1)	3(0)	4(0)	5(1)	5(1)	5(1)	5(1)	1(0)	2(0)	3(1)
		3(1)	3(1)	3(1)	4(0)	5(1)	1(1)	1(1)	1(1)	1(1)	2(0)	3(1)	4(1)
				4(1)	5(1)	1(1)	2(1)	2(1)	2(1)	2(1)	3(1)	4(1)	5(1)

缺页 10 次。

(4) 改进的时钟算法 clock (假设所有对页面 p2 的访问都是写请求)

图中(r, m)为(引用位, 修改位)

F	F	F		F	F(1)	F(3)					F(4)	F(5)	F(1)
P1	p2	p3	p1	p4	p5	p1	p2	p1	p4	p5	p3	p4	p5
				→		5(1,0)					→		
1(1,0)	1(1,0)	1(1,0)	1(1,0)	1(1,0)	5(1,0)		5(1,0)	5(1,0)	5(1,0)	5(1,0)	5(0,0)	4(1,0)	4(1,0)
→				→							→		
	2(1,1)	2(1,1)	2(1,1)	2(1,1)	2(0,1)	2(0,1)	2(1,1)	2(1,1)	2(1,1)	2(1,1)	2(0,1)	2(0,1)	2(0,1)
	→												
		3(1,0)	3(1,0)	3(1,0)	3(0,0)	1(1,0)	1(1,0)	1(1,0)	1(1,0)	1(1,0)	1(0,0)	1(0,0)	5(1,0)
		→	→			→	→	→	→	→			→
				4(1,0)	4(0,0)	4(0,0)	4(0,0)	4(0,0)	4(1,0)	4(1,0)	3(1,0)	3(1,0)	3(1,0)

缺页 9 次。

(5) 最近最少使用算法 (LRU)

F	F	F		F	F(2)		F(3)				F(2)		
1	2	3	1	4	5	1	2	1	4	5	3	4	5
1	2	3	1	4	5	1	2	1	4	5	3	4	5
	1	2	3	1	4	5	1	2	1	4	5	3	4
		1	2	3	1	4	5	5	2	1	4	5	3
				2	3	3	4	4	5	2	1	1	1

缺页 7 次。

(6) 局部最优页面置换算法 (MIN)

设滑动窗口 $\tau = 3$

时刻 t	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
引用串		P1	P2	P3	P1	P4	P5	P1	P2	P1	P4	P5	P3	P4	P5
P1		√	√	√	√	√	√	√	√	√					
P2			√						√						
P3				√									√		
P4						√					√	√	√	√	
P5							√					√	√	√	√
IN		P1	P2	P3		P4	P5	P2			P4	P5	P3		
OUT				P2	P3		P4	P5		P2	P1			P3	P4

缺页 9 次。

(7) 工作集算法 (WS), $\Delta = 2$

时刻 t	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14
引用串		P1	P2	P3	P1	P4	P5	P1	P2	P1	P4	P5	P3	P4	P5
P1		√	√	√	√	√	√	√	√	√	√	√			
P2			√	√	√				√	√	√				
P3				√	√	√							√	√	√
P4						√	√	√			√	√	√	√	√
P5							√	√	√			√	√	√	√
IN		P1	P2	P3			P5		P2		P4	P5	P3		
OUT						P2	P3		P4	P5		P2	P1		

缺页 8 次。