

第 4 章 应用题参考答案

1. 在一个请求分页虚拟存储管理系统中，一个程序运行的页面走向是：

1、2、3、4、2、1、5、6、2、1、2、3、7、6、3、2、1、2、3、6。

分别用 FIFO、OPT 和 LRU 算法，对分配给程序 3 个页框、4 个页框、5 个页框和 6 个页框的情况下，分别求出缺页异常次数和缺页中断率。

解：

页框数	FIFO	LRU	OPT
3	16	15	11
4	14	10	8
5	12	8	7
6	9	7	7

只要把表中缺页异常次数除以 20，便得到缺页中断率。

2. 在一个请求分页虚拟存储管理系统中，一个作业共有 5 页，执行时其访问页面次序为：(1) 1、4、3、1、2、5、1、4、2、1、4、5。

(2) 3、2、1、4、4、5、5、3、4、3、2、1、5。

若分配给该作业三个页框，分别采用 FIFO 和 LRU 页面替换算法，求出各自的缺页异常次数和缺页中断率。

解：

(1) 采用 FIFO 为 9 次， $9/12=75\%$ 。采用 LRU 为 8 次， $8/12=67\%$ 。

(2) 采用 FIFO 和 LRU 均为 9 次， $9/13=69\%$ 。

3. 一个页式存储管理系统使用 FIFO、OPT 和 LRU 页面替换算法，如果一个作业的页面走向为：

(1) 2、3、2、1、5、2、4、5、3、2、5、2。

(2) 4、3、2、1、4、3、5、4、3、2、1、5。

(3) 1、2、3、4、1、2、5、1、2、3、4、5。

当分配给该作业的物理块数分别为 3 和 4 时，试计算访问过程中发生的缺页异常次数和缺页中断率。

解：

(1) 作业的物理块数为 3 块，使用 FIFO 为 9 次， $9/12=75\%$ 。使用 LRU 为 7 次， $7/12=58\%$ 。使用 OPT 为 6 次， $6/12=50\%$ 。

作业的物理块数为 4 块，使用 FIFO 为 6 次， $6/12=50\%$ 。使用 LRU 为 6 次， $6/12=50\%$ 。使用 OPT 为 5 次， $5/12=42\%$ 。

(2) 作业的物理块数为 3 块，使用 FIFO 为 9 次， $9/12=75\%$ 。使用 LRU 为 10 次， $10/12=83\%$ 。使用 OPT 为 7 次， $7/12=58\%$ 。

作业的物理块数为 4 块，使用 FIFO 为 10 次， $10/12=83\%$ 。使用 LRU 为 8

次， $8/12=66\%$ 。使用 OPT 为 6 次， $6/12=50\%$ 。

其中，出现了 Belady 现象，增加分给作业的内存块数，反使缺页中断率上升。

4. 在可变分区存储管理下，按地址排列的内存空闲区为：10K、4K、20K、18K、7K、9K、12K 和 15K。对于下列的连续存储区的请求：(1)12K、10K、9K，(2)12K、10K、15K、18K 试问：使用首次适应算法、最佳适应算法、最差适应算法和下次适应算法，哪个空闲区被使用？

解：

(1) 空闲分区如图所示。

分区号	分区长
1	10KB
2	4KB
3	20KB
4	18KB
5	7KB
6	9KB
7	12KB
8	15KB

1)首次适应算法

12KB 选中分区 3，这时分区 3 还剩 8KB。10KB 选中分区 1，恰好分配故应删去分区 1。9KB 选中分区 4，这时分区 4 还剩 9KB。

2)最佳适应算法

12KB 选中分区 7，恰好分配故应删去分区 7。10KB 选中分区 1，恰好分配故应删去分区 1。9KB 选中分区 6，恰好分配故应删去分区 6。

3)最差适应算法

12KB 选中分区 3，这时分区 3 还剩 8KB。10KB 选中分区 4，这时分区 4 还剩 8KB。9KB 选中分区 8，这时分区 8 还剩 6KB。

4)下次适应算法

12KB 选中分区 3，这时分区 3 还剩 8KB。10KB 选中分区 4，这时分区 4 还剩 8KB。9KB 选中分区 6，恰好分配故应删去分区 6。

(2) 原始分区情况同上图。

1)首次适应算法

12KB 选中分区 3，这时分区 3 还剩 8KB。10KB 选中分区 1，恰好分配故应删去分区 1。15KB 选中分区 4，这时分区 4 还剩 3KB。最后无法满否 18KB 的申请，应该等待。

2)最佳适应算法

12KB 选中分区 7，恰好分配故应删去分区 7。10KB 选中分区 1，恰好分配故应删去分区 1。15KB 选中分区 8，恰好分配故应删去分区 8。18KB 选中分区 4，恰好分配故应删去分区 4。

3)最差适应算法

12KB 选中分区 3，这时分区 3 还剩 8KB。10KB 选中分区 4，这时分区 4 还剩 8KB。15KB 选中分区 8，恰好分配故应删去分区 8。最后无法满否 18KB 的申请，应该等

待。

4)下次适应算法

12KB 选中分区 3, 这时分区 3 还剩 8KB。10KB 选中分区 4, 这时分区 4 还剩 8KB。
15KB 选中分区 8, 恰好分配故应删去分区 8。最后无法满否 18KB 的申请, 应该等待。

5. 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 算法将它们装入到内存的哪个分区?(2) 哪个算法能最有效利用内存?

解:

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 算法能最有效利用内存

6. 一个 32 位地址的计算机系统使用二级页表, 虚地址被分为 9 位顶级页表, 11 位二级页表和偏移。试问: 页面长度是多少? 虚地址空间共有多少个页面?

解: 由于 $32-9-11=12$, 所以, 页面大小为 4KB, 页面的个数为 2^{20} 个。

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, 在内存有 3 个和 4 个空闲页框的情况下, 分别给出页面替换次数。

解: 内存有 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, 增加分给作业的内存块数, 反使缺页中断率上升。

8. 某计算机有缓存、内存、外存来实现虚拟存储器。如果数据在缓存中, 访问它需要 **Ans**; 如果在内存但不在缓存, 需要 **Bns** 将其装入缓存, 然后才能访问; 如果不在内存而在外存, 需要 **Cns** 将其读入内存, 然后, 用 **Bns** 再读入缓存, 然后才能访问。假

设缓存命中率为 $(n-1)/n$, 内存命中率为 $(m-1)/m$, 则数据平均访问时间是多少?

解:

数据在缓存中的比率为: $(n-1)/n$

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

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

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

关于本题有同学提问, 及作进一步讨论如下:

数据在缓存中的比率为: $(n-1)/n$

批注 1:

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%。

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

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

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

同学提问:

第4章第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?

批注 2:

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%。

批注 3: 因为这个题目没有明确两个概率缓存命中率为 $(n-1)/n$, 主存命中率为 $(m-1)/m$ 是全局性的, 还是局部的条件概率。

(1) 参考答案的前提是缓存命中率为 $(n-1)/n$ 是全局的, 主存命中率为 $(m-1)/m$ 是局部的条件概率。

(2) 因此, 结合同学的提问, 给出的解的前提是缓存命中率为 $(n-1)/n$, 主存命中率为 $(m-1)/m$ 都是全局性的。

要 20ns；如果在内存但不在 cache，需要 60ns 将其装入缓存，然后才能访问；如果不在内存而不在外存，需要 12μs 将其读入内存，然后，用 60ns 再读入 cache，然后才能访问。假设 cache 命中率为 0.9，内存命中率为 0.6，则数据平均访问时间是多少(ns)？

解：

506ns。

10. 有一个分页系统，其页表存放在内存里，(1)如果对内存的一次存取要 1.2 微秒，试问实现一次页面访问的存取需花多少时间？(2)若系统配置了相联存储器，命中率为 80%，假定页表表目在相联存储器的查找时间忽略不计，试问实现一次页面访问的存取时间是多少？

解：(1)2.4 微秒 (2) $0.8 \times 1.2 + 0.2 \times 2.4 = 0.76 + 0.48 = 1.24$ 微秒

11. 给定段表如下：

段 号	段 首 址	段 长
0	219	600
1	2300	14
2	90	100
3	1327	580
4	1952	96

给定地址为段号和位移：1) [0, 430]、2) [3, 400]、3) [1, 1]、4) [2, 500]、5) [4, 42]，试求出对应的内存物理地址。

解：1)649 2)1727 3)2301 4)越界 5)1994

12. 某计算机系统提供 24 位虚存空间，内存为 2^{18}B ，采用分页式虚拟存储管理，页面尺寸为 1KB。假定用户程序产生了虚拟地址 11123456（八进制），而该页面分得块号为 100(八进制)，说明该系统如何产生相应的物理地址且写出物理地址。

解：虚拟地址 11123456（八进制）转化为二进制为：

001 001 001 010 011 100 101 110

其中前面为页号，而后 10 位为位移：001 001 001 010 01-----1 100 101 110。由于内存大小为 2^{18}B ，页面尺寸为 1KB，所以，内存共有 256 块。所以，块号为 100(八进制)是合法地址，于是，物理地址为 100（八进制）与位移 1 100 101 110 并接，得到：八进制物理地址 001000000 1 100 101 110=201456（八进制）。

13. 内存中有两个空闲区如图所示，

0K	
15K	100K
125K	50K

现有作业序列依次为：Job1 要求 30K；Job2 要求 70K；Job3 要求 50K；使用首次适应、最坏适应和最佳适应算法处理这个作业序列，试问哪种算法可以满足分配？为什么？

解：首次适应、最坏适应算法处理这个作业序列可以满足分配，最佳适应算法不行。因为后者会分割出无法使用的碎片，浪费内存，从而，不能满足所有作业的内存需求。

14. 设有一页式存储管理系统，向用户提供的逻辑地址空间最大为 16 页，每页 2048 字节，内存总共有 8 个存储块。试问逻辑地址至少应为多少位？内存空间有多大？

解：逻辑地址 $2^{11} \times 2^4$ ，故为 15 位。内存大小为 $2^3 \times 2^{11} = 2^{14} \text{B} = 16 \text{KB}$ 。

15. 在一分页存储管理系统中，逻辑地址长度为 16 位，页面大小为 4096 字节，现有一逻辑地址为 2F6AH，且第 0、1、2 页依次存在物理块 10、12、14 号中，问相应的物理地址为多少？

解：因为逻辑地址长度为 16 位，而页面大小为 4096 字节，所以，前面的 4 位表示页号。把 2F6AH 转换成二进制为：0010 1111 0110 1010，可知页号为 2。故放在 14 号物理块中，写成十六进制为：EF6AH。

16. 有数组 `int A[100][100]`；元素按行存储。在一虚存系统中，采用 LRU 淘汰算法，一个进程有 3 页内存空间，每页可以存放 200 个整数。其中第 1 页存放程序，且假定程序已在内存。

程序 A:

```
for(int i=0; i<100; i++)
    for(int j=0; j<100; j++)
        A[i, j] = 0;
```

程序 B:

```
for(int j=0; j<100; j++)
    for(int i=0; i<100; i++)
        A[i, j] = 0;
```

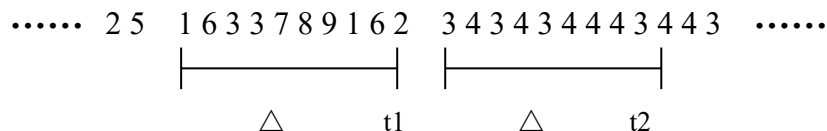
分别就程序 A 和 B 的执行进程计算缺页次数。

解：题中 $100 \times 100 = 10000$ 个数据，每页可以存放 200 个整数，故一共存放在 50 个页面中。由于元素按行存储，第 1 行、第 2 行放在第 1 页， \dots ，第 99 行、第 100 行放在第 50 页。故对于程序 A，缺页异常为 50 次。对于程序 B，缺页异常为 5000 次。

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?

解：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。

18. 在页式虚拟存储管理中,为解决抖动问题,可采用工作集模型以决定分给进程的物理块数,有如下页面访问序列:



窗口尺寸 $\Delta=9$ ，试求 t_1 、 t_2 时刻的工作集。

解: t1 时刻的工作集为: {1, 2, 3, 6, 7, 8, 9}。t2 时刻的工作集为: {3, 4}。

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% 。

解: (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。

20. 在一个分页虚存系统中，用户编程空间 32 个页，页长 1KB，内存为 16KB。如果用户程序有 10 页长，若已知虚页 0、1、2、3，已分到页框 8、7、4、10，试把虚地址 0AC5H 和 1AC5H 转换成对应的物理地址。

解：虚地址 0AC5H 对应的物理地址为：12C5H。而执行虚地址 1AC5H 会发现页表中尚未有分配的页框而发生缺页异常，由系统另行分配页框。

21. 一个进程已分配到 4 个页框，每页的装入时间、最后访问时间、访问位 R、修改位 D 如表所示（所有数字为十进制，且从 0 开始），当进程访问第 4 页时，产生缺页异常。请分别用 FIFO、LRU 和 NRU 算法，决定缺页异常服务程序选择换出的页面。

page	Page frame	loaded	last reference	R	D
2	0	60	161	0	1
1	1	130	160	0	0
0	2	26	162	1	0
3	3	20	163	1	1

解：装入时间越小，则装入越早。同样，最近访问时间越小，则越早。于是，

FIFO: 换出进入内存时间最久的页面, 表中第 4 项(即 page3)装入内存最久, 所以被替换。

LRU: 换出最近最久时间没有使用的页面, 第 1 和第 2 项(即 page1 和 page2)的 R 位为 0, 最近没有被访问, 但 page1 的访问比 page2 时间早, 所以换出 page1。

NRU: 选择在最近一段时间内未使用过的一页换出。表中第 1 项和第 2 项的 R 位为 0, 但 page2 的 D 位为 1, 故仅 page1 未被访问, 所以被换出。

22. 一台计算机中, 4 个页框中的页面当前状态: 每页装入时间、最后访问时间、访问位 R、修改位 D 如下表所示, 试分别用 FIFO、LRU、NRU 和 SCR 算法, 决定缺页异常服务程序选择换出的页面。

page	loaded	last reference	R	D
0	126	269	0	0
1	230	250	1	0
2	110	273	1	1
3	160	280	1	1

解:

FIFO—淘汰最先进入内存的页面。P2 最先进入(装入时间最早 110), 故淘汰 P2。

LRU--淘汰最近最久未使用的页面。P1 最近最少使用(上次引用时间最早 250), 故淘汰 P1。

NRU--淘汰最近未被访问的页面。只有 P0 的 R 及 D 均为 0, 故淘汰 P0。

SCR--淘汰自上次检查过以来未被访问过的页面。只有 P0 未被访问过, 故淘汰 P0。

23. 考虑下面的程序:

```
for(int i=0;i<20; i++)
    for(int j=0;j<10;j++)
        a[i]=a[i]×j;
```

试举例说明该程序的空间局部性和时间局部性。

解: 当数组元素 $a[0]$, $a[1]$, \dots , $a[19]$ 存放在一个页面中时, 其空间局部性和时间局部性较好, 也就是说, 在很短时间内执行都挂行循环乘法程序, 而且数组元素分布在紧邻连续的存储单元中。当数组元素存放在不同页面中时, 其时间局部性虽相同, 但空间局部性较差, 因为处理的数组元素分布在不连续的存储单元中。

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

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

根据上述分析, 平均有效访问时间是:

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

25. 一个有快表的请页式虚存系统, 设内存访问周期为 1 微秒, 内外存传送一个页面的平均时间为 5 毫秒。如果快表命中率为 75%, 缺页中断率为 10%。忽略快表访问时间, 试求内存的有效存取时间。

解: 快表命中率为 75%, 缺页中断率为 10%, 所以, 内存命中率为 15%。故内存的有效存取时间= $1 \times 75\% + 2 \times 15\% + (5000 + 2) \times 10\% = 501.25$ 微秒。

批注:

按照第 25 题解法, 快表的命中率 75% 是全局性的, 即假设有 100 次页面访问, 有 75 次在快表中直接命中。有 10 次发生缺页中断, 有 15 次在页表(主存)中命中。

26. 假设某虚存的用户空间为 1024KB, 页面大小为 4KB, 内存空间为 512KB。已知用户的虚页 10、11、12、13 页分得内存页框号为 62、78、25、36, 求出虚地址 0BEBC(16 进制)的实地址(16 进制)是多少?

解: 虚地址 0BEBC(16 进制)的二进制形式为: 0000 1011 1110 1011 1100。由于页面大小为 4KB, 故其中后 12 位是位移, 所以, 虚地址的页号为: 11。查页表分得内存对应页框号为: 78。已知内存空间为 512KB, 故内存共有 128 个页框, 78 是合法物理块。把 78 化为 16 进制是 4E, 虚地址 0BEBC(16 进制)的实地址(16 进制)是: 4EEBC。

27. 某请求分页存储系统使用一级页表, 假设页表全部放在内存内, :

- 1) 若一次访问内存花 120ns, 那么, 访问一个数据的时间是多少?
- 2) 若增加一个快表, 在命中或失误时需有 20ns 开销, 如果快表命中率为 80%, 则访问一个数据的时间为多少?

解: 1) $120\text{ns} \times 2 = 240\text{ns}$ 。

2) $(120 + 20) \times 80\% + (120 + 120 + 20) \times 20\% = 174\text{ns}$ 。

28. 设某系统中作业 J_1 , J_2 , J_3 占用内存的情况如图。今有一个长度为 20k 的作业 J_4 要装入内存, 当采用可变分区分配方式时, 请回答:

- (1) J_4 装入前的内存已分配表和未分配表的内容。
- (2) 写出装入 J_4 时的工作流程, 并说明你采用什么分配算法。

0	OS
10k	J_1
18k	
30k	J_2
40k	
54k	J_3
70k	

解: (1) 内存已分配表共有三项, 由作业 J_1 、 J_2 、 J_3 占用, 长度依次为: 10k、30k 和 54k。未分配表共有三项: 空闲区 1、空闲区 2 和空闲区 3, 长度依次为 18k、40k 和 70k。

(2) 作业 J_4 装入时, 采用直接分配, 搜索未分配表, 空闲区 1 不能满足。所以, 要继续搜索未分配表, 空闲区 2 可以满足 J_4 的装入要求。

29. 考虑下列的段表：

段号	始址	段长
0	200	500
1	890	30
2	120	100
3	1250	600
4	1800	88

对下面的逻辑地址，求物理地址，如越界请指明。1) <0,480> 2) <1,25> 3) <1,14> 4) <2,200> 5) <3,500> 6) <4,100>。

解：1)680 2)915 3)904 4)越界 5)1750 6) 越界。

30. 请页式存储管理中，进程访问地址序列为：

10,11,104,170,73,305,180,240,244,445,467, 366。

试问 1)如果页面大小为 100，给出页面访问序列。2)进程若分得 3 个页框，采用 FIFO 和 LRU 替换算法，求缺页中断率？

解：1) 页面访问序列为 1, 1, 2, 2, 1, 4, 2, 3, 3, 5, 5, 4。

2)FIFO 为 5 次，缺页中断率为 $5/12=41.6\%$ 。LRU 为 6 次，缺页中断率为 $6/12=50\%$ 。LRU 反比 FIFO 缺页中断率高。

31. 设程序大小为 460 个字，考虑下列访问序列：

55、20、108、180、79、310、170、255、246、433、488、369

(1) 设页面大小为 100 个字，试给出访问序列页面走向。(2) 假设程序可用内存为 200 个字，采用 FIFO、LRU 和 OPT 淘汰算法，试求出缺页中断率。

解：(1) 0、0、1、1、0、3、1、2、2、4、4、3。

(2)

算法	FIFO	LRU	OPT
缺页次数	6	7	5
缺页率	6/12	7/12	5/12

32. 假设计算机有 2M 内存，其中，操作系统占用 512K，每个用户程序也使用 512K 内存。如果所有程序都有 70%的 I/O 等待时间，那么，再增加 1M 内存，吞吐率增加多少？

答：由题意可知，内存中可以存放 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 内存，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 内存，吞吐率增加了： $83.2\% \div 65.7\% - 100\% = 27\%$ 。

33. 一个计算机系统有足够的内存空间存放 4 道程序，这些程序有一半时间在空闲等待 I/O 操作。问多大比例的 CPU 时间被浪费掉了？

解： $(50\%)^4=(1/2)^4=1/16$ 。

34. 如果执行一条指令平均需 1 微秒, 处理一个缺页异常另需 n 微秒, 给出当缺页异常每 k 条指令发生一次时, 指令的实际执行时间。

解: $(1+n/k)$ 微秒。

35. 一台计算机的内存空间为 1024 个页面, 页表放在内存中, 从页表中读一个字的开销是 500ns。为了减少开销, 使用了有 32 个字的快表, 查找速度为 100ns。要把平均开销降到 200ns 需要的快表命中率是多少?

解: 设快表命中率是 x , 则内存命中率为 $1-x$ 。于是: $500(1-x)+100x=200$, 解方程得 $x=75\%$ 。

36. 假设一条指令平均需花 1 微秒, 但若发生了缺页异常就需 2001 微秒。如果一个程序运行了 60 秒, 期间发生了 15000 次缺页异常, 若可用内存是原来的两倍, 这个程序运行需要多少时间?

解: 一个程序运行期间发生了 15000 次缺页异常, 由于缺页异常处理花 2000 微秒(1 微秒是指令执行时间, 于是这个程序缺页异常处理花了: $2000 \text{ 微秒} \times 15000 = 30 \text{ 秒}$ 。占了运行时间 60 秒的一半。当可用内存是原来的两倍时, 缺页异常次数减为一半, 故有 15 秒就能处理完。所以, 这个程序运行需要时间为: 45 秒。

37. 在分页式虚存管理中, 若采用 FIFO 替换算法, 会发生: 分给作业页面越多, 进程执行时缺页中断率越高的奇怪现象。试举 2 个例子说明这个现象。

解: 考查页面走向----(1)4,3,2,1,4,3,5,4,3,2,1,5。(2)0,1,2,3,0,1,4,0,1,2,3,4。

当分配页框数为 3 时, 均缺页 9 次。当分配页框数为 4 时, 均缺页 10 次。

38. 假设一个任务被划分成 4 个大小相等的段, 每段有 8 项的页描述符表, 若页面大小一为 2KB。试问段页式存储系统中: (a)每段最大尺寸是多少?(b)该任务的逻辑地址空间最大为多少?(c)若该任务访问到逻辑地址空间 5ABCH 中的一个数据, 试给出逻辑地址的格式。

解: 段数 $2^2=4$, 每段有 $2^3=8$ 页, 页大小为 $2^{11}=2\text{KB}$ 。(a) 故每段最大为 $2^{14}\text{B}=16\text{KB}$ 。(b) 逻辑地址空间最大 $4 \times 16\text{KB}=64\text{KB}$ 。

(c) 若该任务访问到逻辑地址空间 5ABCH, 其二进制表示为:

0101 1010 1011 1100

所以, 逻辑地址表示为: 01 011 010 1011 1100

5ABCH 的逻辑地址为: 第 1 段 第 3 页, 位移由后 11 位给出。

39. 进程在某时刻的页表如下, 设页面大小为 1KB, 表中所有数字为十进制)。

页号	有效位	访问位	修改位	页框号
0	1	1	0	4
1	1	1	1	7
2	0	0	0	
3	1	0	0	2
4	0	0	0	
5	1	0	1	0

下列虚地址转换为物理地址的值是多少?(1)1052, (2)2221, (3)5499。

解: 1) $1052 \div 1024=1$, $1052 \bmod 1024=28$

查页表可知 1 页对应的内存块号为 4, 所以物理地址为 $4 \times 1024+28=4124$

2) $2221 \div 1024=2$, $2221 \bmod 1024=173$

查页表可知第 2 页不在内存, 将产生缺页异常。

3) $5499 \div 1024=5$, $5499 \bmod 1024=379$

查页表可知其对应块号为 0, 所以物理地址为 379。

40. 已知某系统页面长 4KB, 页表项 4B, 采用多级页表映射 64 位虚地址空间。若限定最高层页表占 1 页, 问它可以采用几级页表?

解: 由于页面长 4KB, 页表项 4B, 故每页可包含 1KB 个页表项。由于限定最高层页表占 1 页, 即它的页表项为 2^{10} 个; 而每个页表项指向一页, 每页又存放页表项个数为 2^{10} 个, 依此类推, 最多可以采用 $64/6$ 取整为 6 级页表。

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, 内存访速度为 100ns, 快表访问速度为 20ns, 缺页异常处理耗时为 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?

解: 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, 所以每访问一次内存中的数据, 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内存中的页表。

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 访问内存就可以了。如果没有命中, 系统还需要访问两次内存。第 1 次是访问内存中的页表, 第 2 次是访问内存中的数据。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 ma 是:

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

批注 1:

按照这个公式的计算思路, 快表命中率 20% 是相对的, 即设有 100 次页面访问, 如果缺页中断率为 60%, 那么有 40 次命中(含快表命中和页表命中), 其中快表为 $40 \times 20\% = 8$ 次, 页表为 $40 \times (1-20\%) = 32$ 次。这与第 25 题解法的思路不一样。

批注 2:

严格地说, 这其中存在歧义, 在于快表命中率看成全局的还是看成局部的, 两个题目均未明确指明。

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 $T = (1-p)ma + p \times \text{缺页异常耗时}$

填入本题中的已知条件后, 得:

$$\begin{aligned} T &= (1-p) \times ma + p \times 25 \text{ (ms)} \\ &= 0.4 \times 200 \text{ (ns)} + 0.6 \times 25 \text{ 000000} \\ &= 15000080 \text{ (ns)}. \end{aligned}$$

批注 3:

如果按照第 25 题的公式的计算思路求解, 本题结果应该为:

$$T = (20+100) \times 20\% \text{ (ns)} + 220 \times 20\% \text{ (ns)} + 25 \text{ 000000} \times 60\% \text{ (ns)} = 15000068 \text{ (ns)}$$

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, 试给出缺页异常的上限和下限, 并举例说明。

解: 对于 FIFO、LRU 两种页面替换算法, 缺页异常的上限和下限: 为 p 和 n 。因为有 n 个不同页面, 无论怎样安排, 不同页面进入内存至少要产生一次缺页异常, 故下限为 n 次。由于 $m < n$, 引用串中有些页可能进入内存后又被调出, 而多次发生缺页异常。极端情况, 访问的页都不在内存, 这样共发生了 p 次缺页异常。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。缺页异常为下限 4 次。而访问串: 2, 3, 4, 1, 2, 3, 4, 1, 2, 3, 4, 1。缺页异常为上限 12 次。

43. 在请求分页虚存管理系统中, 页表保存在寄存器中。若替换一个未修改过页面的缺页异常处理需 8 毫秒, 若替换一个已修改过页面的缺页异常处理需另加写盘时间 12 毫秒, 内存存取周期为 1 微秒。假定 70% 被替换的页面被修改过, 为保证有效存取时间不超过 2 微秒, 允许的最大缺页中断率为多少?

解: 设最大缺页中断率为 x , 则有:

$$(1-x) \times 1 \text{ 微秒} + (1-70\%) \times x \times 8 \text{ 毫秒} + 70\% \times x \times (8+12) = 2 \text{ 微秒}$$

即得到: $-x + 2400x + 14000x = 1$, 解得: x 约为 0.00006。

44. 若内存中按地址递增次序有三个不邻接的空闲区 F1、F2、F3, 它们的大小分别是: 50K、120K 和 25K。请给出后备作业序列, 使得实施分配时: (1) 采用最佳适应算法效果好, 但采用首次适应与最坏适应算法效果不好。(2) 采用最坏适应算法效果好, 但采用首次适应与最佳适应算法效果不好。

解:

F1(50)
F2(120)
F3(25)

(1)采用最佳适应算法效果好,但采用首次适应与最坏适应算法效果不好。作业序列: 25, 120, 50。

(2)采用最坏适应算法效果好,但采用首次适应与最佳适应算法效果不好。作业序列: 40, 80, 50, 25。

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, 且各有一个内存储器 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?

解: 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 为内存储器存取周期, 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: $0.7 \times 60+(1-0.7) \times 1 \mu s=342ns$

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

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: $0.9 \times 60+(1-0.9) \times 1 \mu s=154ns$

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

故计算机 P1 比 P2 处理速度快。

46. Linux 中采用多种 cache 来改善系统运行性能, 试解释所用的缓存 cache、页面 cache、交换 cache 和硬件 cache 的作用。

解: 1) 缓存 cache—存储块设备驱动模块使用的缓存数据, 这些数据是从设备上读取的或要写入设备的, 采用设备标识和块号进行标识, 提高对块设备的访问速度。

2) 页面 cache—用来加快对磁盘数据的访问速度, 它缓存一个文件中的页面内容, 利用文件和及偏移进行标识。

3) 交换 cache—被修改的页面写入交换区, 提高虚存交换页面的速度。

4) 硬件 cache—用在进程地址转换中, 作为快表。

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, 假设所有对页面 p2 的访问都是写请求)。试用下列算法求出缺页异常次数, (a)OPT, (b)FIFO, (c)SCR, (d)改进的 CLOCK, (e)LRU, (f)MIN(滑动窗口 $\tau = 3$), (g)WS(工作集窗口尺寸 $\Delta = 2$)。

解:

(a) 最优置换算法 OPT

F	F	F		F	F(3)						F(1)		
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。

(b) 先进先出算法 FIFO

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

(c) 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(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。

(d) 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)
				→							→		
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(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。

(e) 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
		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。

(f) 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。

(g) 工作集算法 (WS)

设工作集窗口尺寸 $\Delta = 2$

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。

说明: t_4 时, 由于 p_1 不在工作集, 故 p_1 应被调出, 但 t_4 时又需引用 p_1 , 故又应调入, 这时应由 os 控制既不移出 p_1 , 又不移入 p_1 , 即保持 p_1 在工作集中。可以看作没有产生缺页。类似情况均可如此处理。

48. 考虑下面的引用串:

P1、p2、p3、p4、p1、p2、p5、p1、p2、p3、p4、p5

对范围从 1~6 的页框, 使用 FIFO 页面置换方案, 确定其产生的缺页数。画图表示缺页次数和页框数的关系, 以说明 Belady 异常。

解: 1) 页框为 1 时: 若非连续访问相同页面每次都缺页, 易知缺页 12 次。

2) 页框为 2 时: 缺页 12 次。

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

3) 页框为 3 时: 缺页 9 次

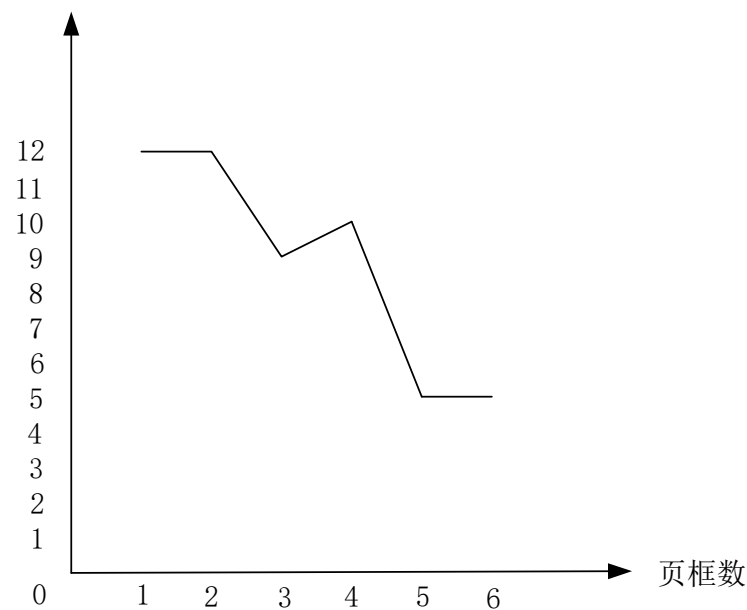
F	F	F	F(1)	F(2)	F(3)	F(4)			F(1)	F(2)	
1	1	1	2	3	4	1	1	1	2	5	5
	2	2	3	4	1	2	2	2	5	3	3
		3	4	1	2	5	5	5	3	4	4

4) 页框为 4 时: 缺页 10 次

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

5) 页框为 5, 6 时, 足够容纳所有页面, 故只有第一次访问产生缺页, 即 5 次

缺页次数



49. 考虑 $\Delta=3$ 的工作集模型。给出进程 p 的引用串如下:

y x x x x x x y y u x x x y z y z w w z x x w w

(a) 进程 p 能够拥有的最大工作集是什么? (b) 进程 p 能够拥有的最小工作集是什么?

解: (a) 由于 $\Delta=3$, 最大工作集即长度为 4 的子串中最多不同页面数, 考察引用串知最大工作集尺寸为 3, 为 $\{x,y,u\}$, $\{x,y,z\}$, $\{y,z,w\}$ 或 $\{w,z,x\}$ 。

(b) 若考虑初始情况最小工作集必为 1, 即 $\{y\}$; 否则由于 $\Delta=3$, 最小工作集即长度为 4 的子串中最少不同页面数, 考察引用串知最小工作集尺寸为 1, 即 $\{x\}$ 。

50. 考虑引用串 abcdebcdbdbddddd。假设使用工作集置换策略, 确定最小的窗口大小, 以保证上面的引用串至多产生 5 次缺页。说明每次引用时哪些页面驻留在内存中。用一个星号标记缺页。

解: 由于引用串中有 5 个不同页面, 初次访问必产生缺页, 所以至少产生 5 次缺页。故要保证移出工作集的页面不再访问。考察引用串, 当访问到 e 时已产生 5 次缺页, 最小窗口即包含 e 后面所有页号的 e 之前的最小子串长度, 即最小窗口为 3。

时刻	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
引用串		a	b	c	d	e	b	c	d	c	b	d	d	b	d	d	d
a		√	√	√	√												
b			√	√	√	√	√	√	√	√	√	√	√	√	√	√	√
c				√	√	√	√	√	√	√	√	√	√				
d					√	√	√	√	√	√	√	√	√	√	√	√	√
e						√	√	√	√								
缺页		*	*	*	*	*											

51. 设进程分得三个页框, 其执行访问序列为: 0 1 2 3 0 1 2 3 0 1 2 3 4 5 6 7, 试采用 (1)Belady, (2)LRU, (3)LFU, (4)FIFO 算法来分别计算缺页异常次数, 并给出缺页时加进内存的页号。

解: (1)Belady 算法共 10 次, 缺页时加进内存的页面见表中带星的页号。

页框	0	1	2	3	0	1	2	3	0	1	2	3	4	5	6	7
0	0*	0	0	0	0	0	0	0	0	1*	1	1	4*	4	4	7*
1		1*	1	1	1	1	2*	2	2	2	2	2	2	5*	5	5
2			2*	3*	3	3	3	3	3	3	3	3	3	3	6*	6

(2)LRU 算法共 16 次, 缺页时加进内存的页面见表中带星的页号。

页框	0	1	2	3	0	1	2	3	0	1	2	3	4	5	6	7
0	0*	0	0	3*	3	3	2*	2	2	1*	1	1	4*	4	4	7*
1		1*	1	1	0*	0	0	3*	3	3	2*	2	2	5*	5	5
2			2*	2	2	1*	1	1	0*	0	0	3*	3	3	6*	6

(3)LFU 算法共 12 次, 缺页时加进内存的页面见表中带星的页号。

页框	0	1	2	3	0	1	2	3	0	1	2	3	4	5	6	7
0	0*	0	0	0	0	0	0	0	0	0	0	3*	3	3	3	3
1		1*	1	1	1	1	1	3*	3	1*	1	1	1	1	1	1
2			2*	3*	3	3	2*	2	2	2	2	2	4*	5*	6*	7*

注意, 如果一个页过去没有被经常使用, 它就会被选替换, 当有多个页满足条件时, 系统可任选一个进行替换。

(4)FIFO 算法共 16 次，缺页时加进内存的页面见表中带星的页号。

页 框	0	1	2	3	0	1	2	3	0	1	2	3	4	5	6	7
0	0*	0	0	3*	3	3	2*	2	2	1*	1	1	4*	4	4	7*
1		1*	1	1	0*	0	0	3*	3	3	2*	2	2	5*	5	5
2			2*	2	2	1*	1	1	0*	0	0	3*	3	3	6*	6