《操作系统》作业汇报与展示

OS-HW6 (Ch 8. Virtual Memory)

崔冠宇

2018202147

cuiguanyu@ruc.edu.cn

信息学院
中国人民大学

2020 年 6 月 11 日

目录

P421, 8.1 题目

Suppose the page table for the process currently executing on the processor looks like the following. All numbers are decimal, everything is numbered starting from zero, and all addresses are memory byte addresses. The page size is 2,048 bytes.

Virtual page number	Valid bit	Reference bit	Modify bit	Page frame number
0	1	1	1	7
1	0	0	0	-
2	0	0	0	-
3	1	0	0	6
4	1	1	0	0
5	1	0	1	3

a. Describe exactly how, in general, a virtual address generated by the CPU is translated into a physical main memory address.

b. What physical address, if any, would each of the following virtual addresses correspond to? (Do not try to handle any page faults, if any.)

(i) 6,204 (ii) 3,021 (iii) 9,000

P421, 8.1 解析

解. **a.** 可参考教材图 8.2:

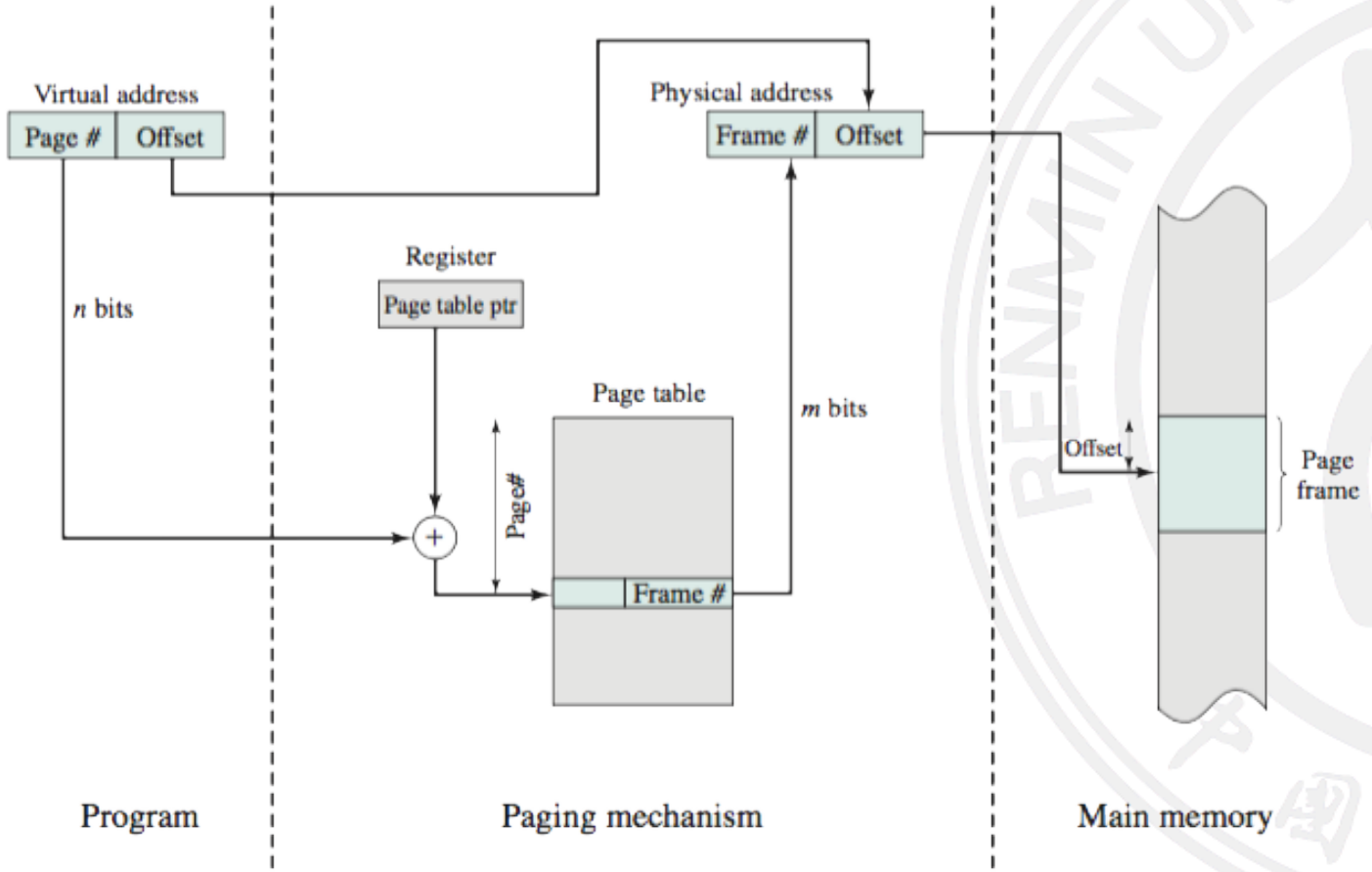


Figure 8.2 Address Translation in a Paging System

P421, 8.1 解析

a. 根据页的大小, 将虚地址分为两部分: 页号和页内偏移量 (此处页大小为 2048 Bytes, 故低 11 位为页内偏移量). 将页表寄存器内的页表始址加上页号, 得到页表中对应的页表项, 然后取出页框号, 再与页面偏移量做拼接, 就得到了物理地址.

b. (i) $6204 = 3 \times 2048 + 60$, 查表得虚页号 3 对应的页框号为 6, 所以物理地址为 $6 \times 2048 + 60 = 12348$.

(ii) $3021 = 1 \times 2048 + 973$, 查表得虚页号 1 有效位为 0, 出现页错误.

(iii) $9000 = 4 \times 2048 + 808$, 查表得虚页号 4 对应的页框号为 0, 所以物理地址为 $0 \times 2048 + 808 = 808$.

P422, 8.4 题目

Consider the following page-reference string: a, b, d, c, b, e, d, b, d, b, a, c, b, c, a, c, f, a, f, d. Assume that there are 3 frames available and that they are all initially empty. Complete a figure, similar to Figure 8.14, showing the frame allocation for each of the following page replacement policies:

a. First-in-first-out

b. Optimal

c. Least recently used

Then, find the relative performance of each policy with respect to page faults.

P422, 8.4 解析

解. 可参考教材图 8.14:

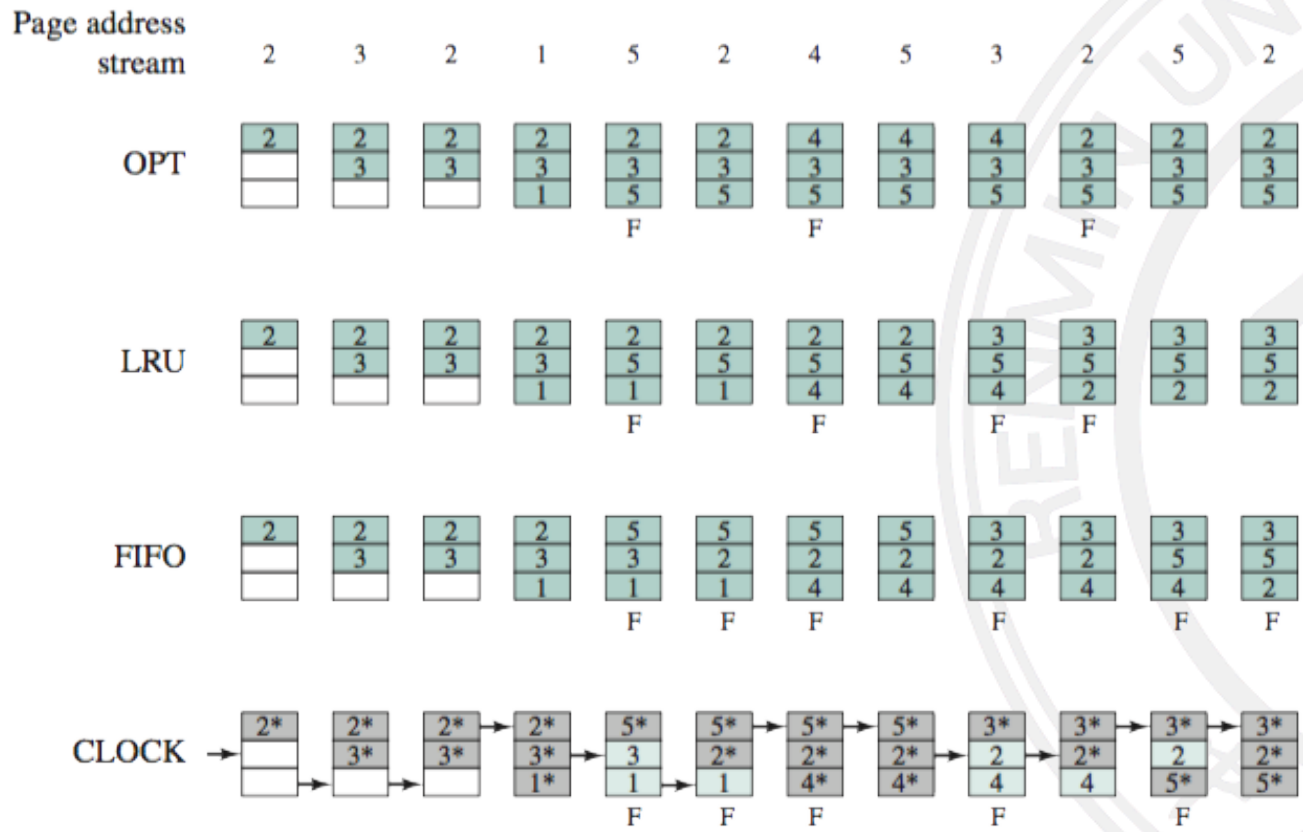
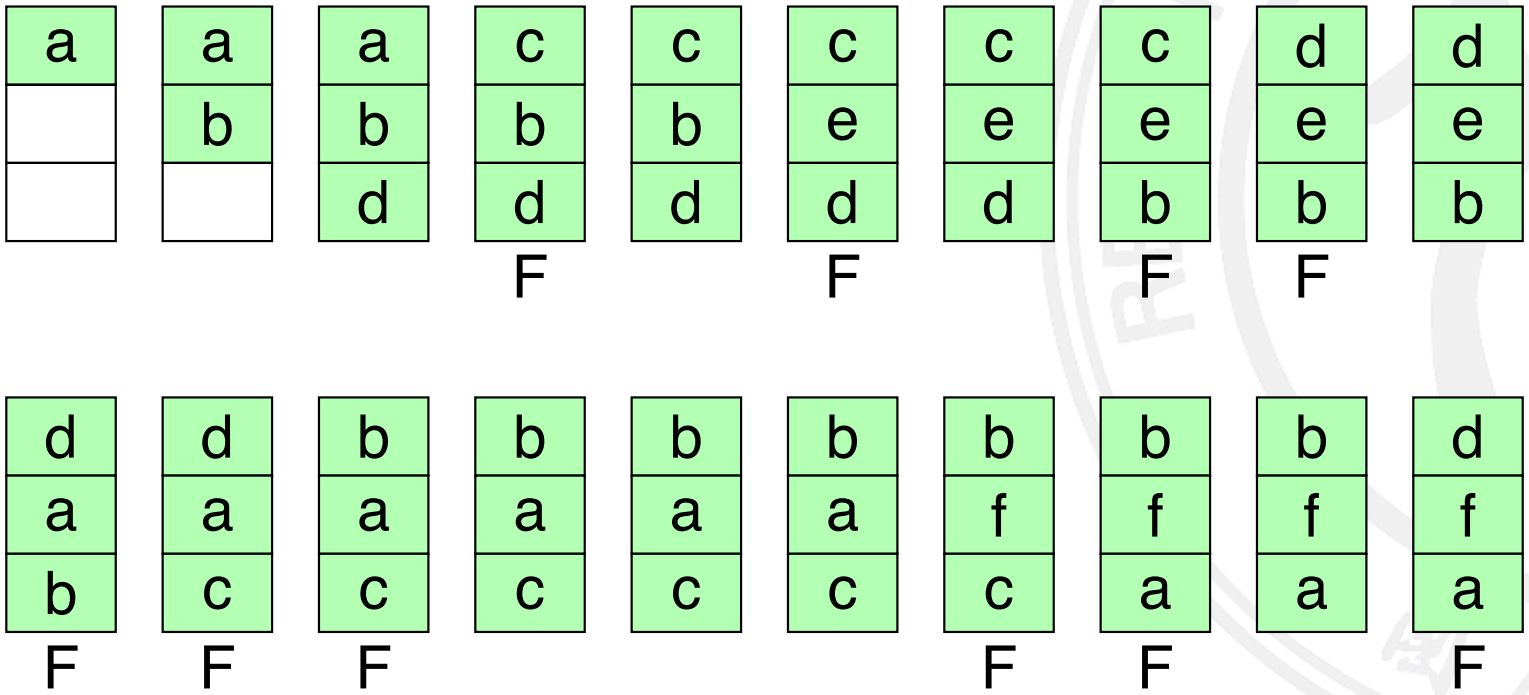


Figure 8.14 Behavior of Four Page Replacement Algorithms

P422, 8.4 解析

(访问顺序: a, b, d, c, b, e, d, b, d, b, a, c, b, c, a, c, f, a, f, d)

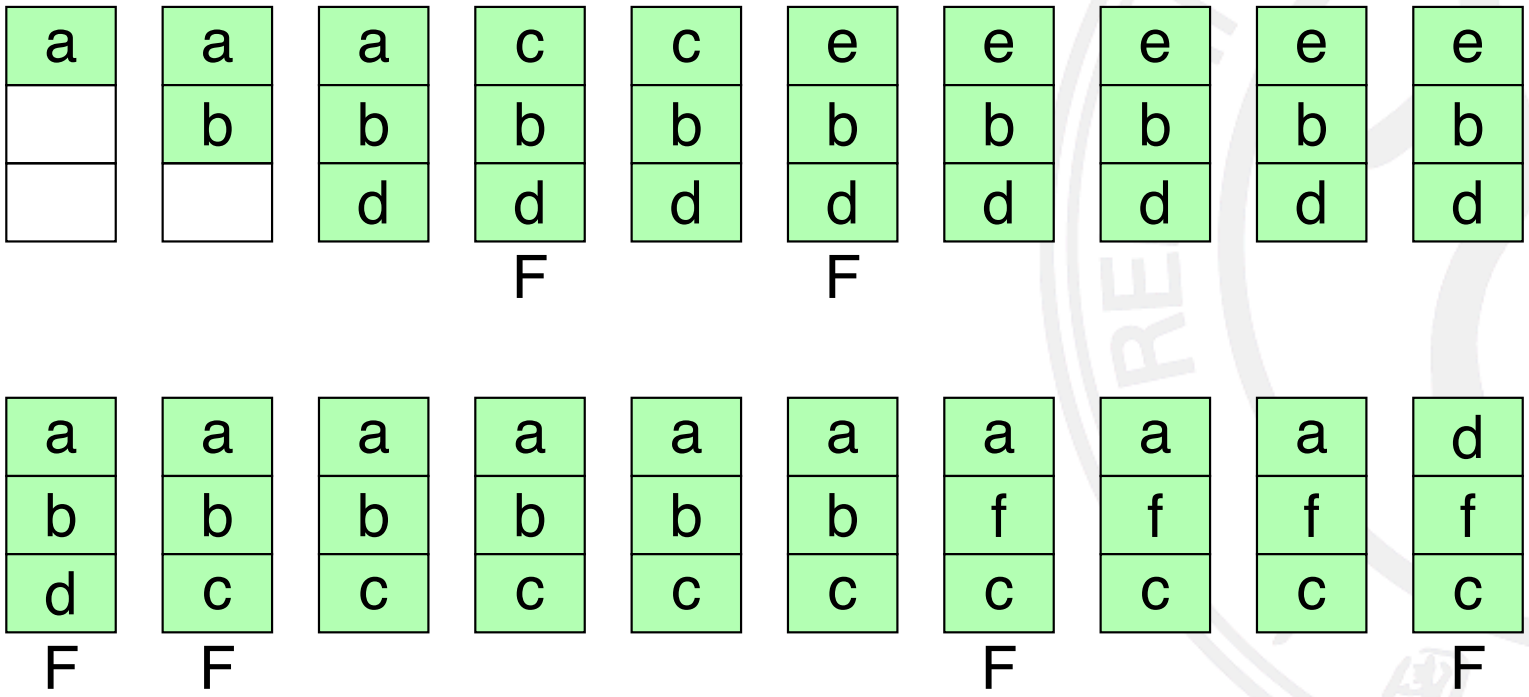
a. First-in-first-out: 先进先出, 即置换上一状态时进程页框中最先被调入的页.



P422, 8.4 解析

(访问顺序: a, b, d, c, b, e, d, b, d, b, a, c, b, c, a, c, f, a, f, d)

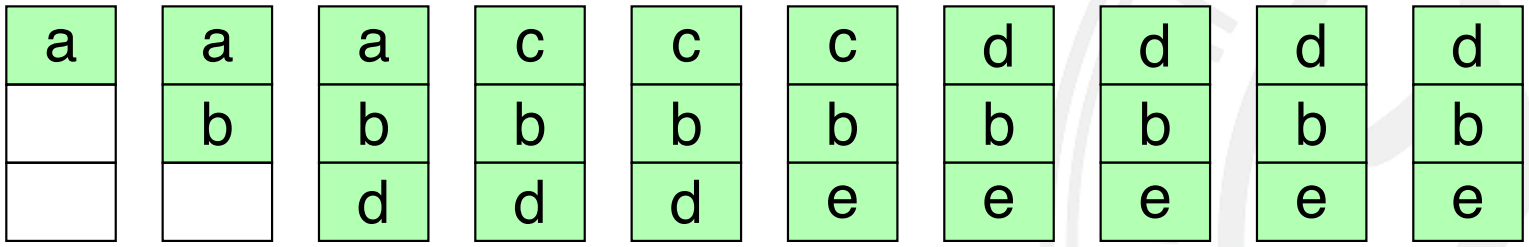
b. Optimal: 选择置换下次访问距当前时间最长的页.

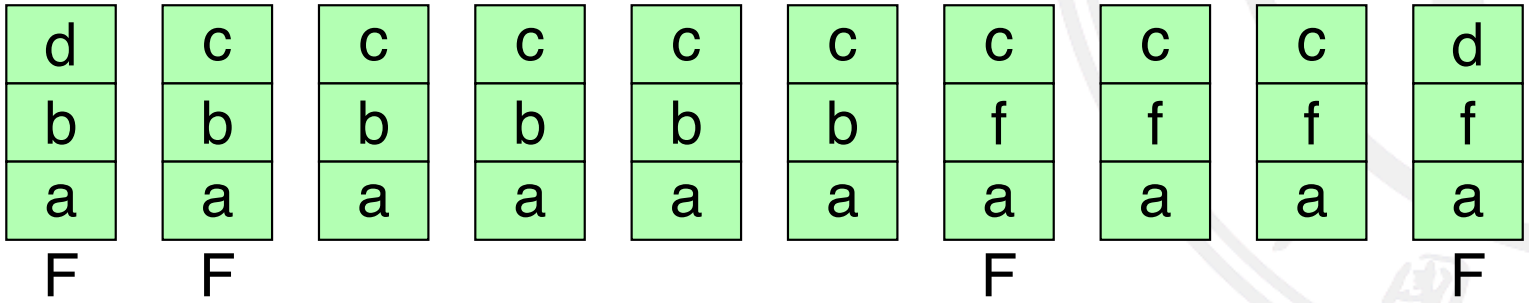


P422, 8.4 解析

(访问顺序: a, b, d, c, b, e, d, b, d, b, a, c, b, c, a, c, f, a, f, d)

c. Least recently used: 置换内存中最长时间未被引用的页.





P422, 8.4 解析

从缺页错误的角度看, FIFO 方式在这种情况下产生了 10 次缺页, 最多; OPT 方式在这种情况下产生了 6 次缺页, 最少; LRU 方式在这种情况下产生了 7 次缺页, 总体较好.