

严锦 1700011049

1.

(1). FIFO

2	3	1	4	0	3	2	4	1	2	4	3
		1	4	0	3	2	4	1	1	1	3
	3	3	1	4	0	3	2	4	4	4	1
2	2	2	3	1	4	0	3	2	2	2	4

LRU

2	3	1	4	0	3	2	4	1	2	4	3
		1	4	0	3	2	4	1	1	1	3
	3	3	1	4	0	3	2	4	4	4	4
2	2	2	3	1	4	0	3	2	2	2	2

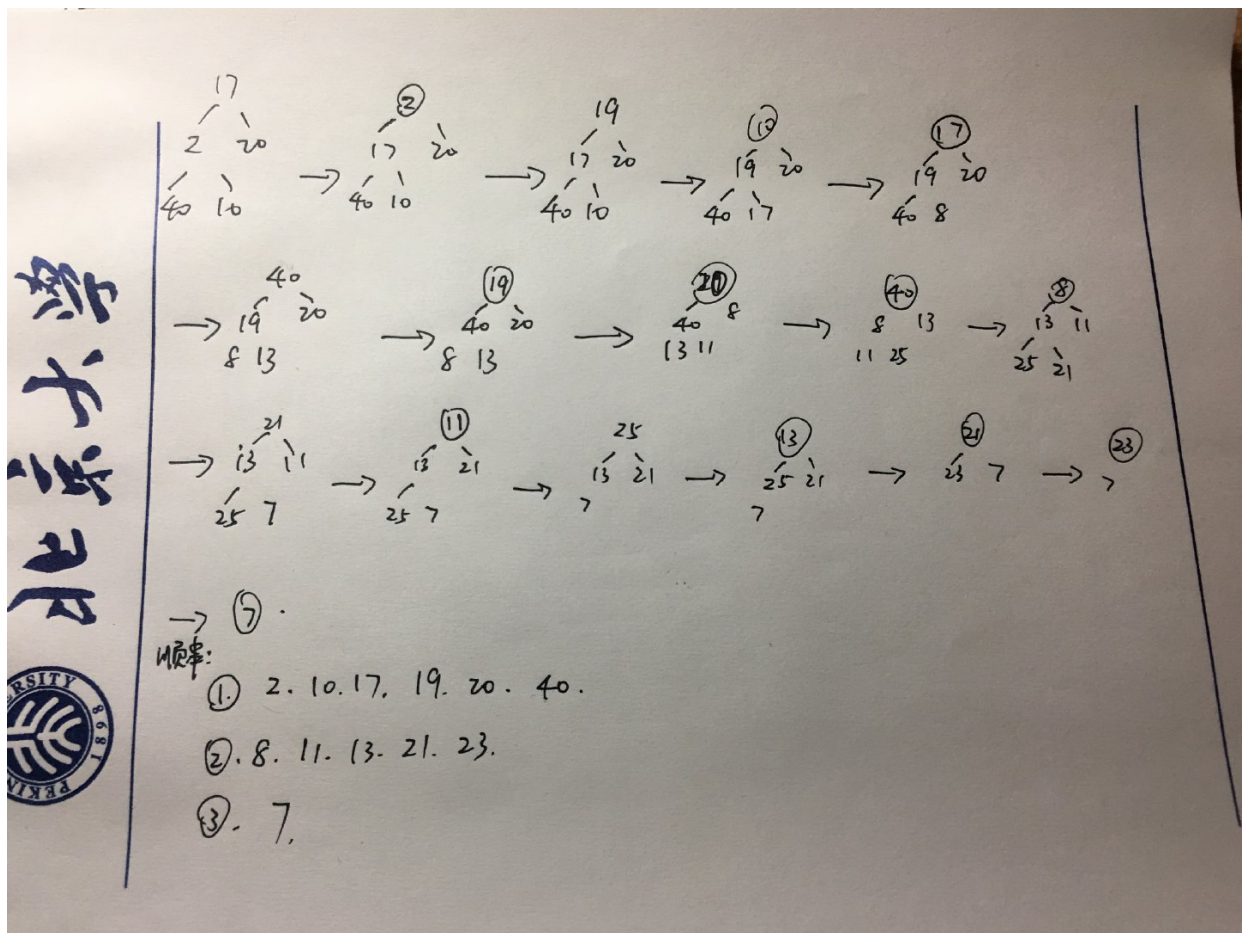
FIFO可能会将较常用的元素替换出缓冲区，而LRU每次替换的都是较不常用的元素

(2). 以(1)为例，当缓冲区容量为5时，FIFO和LRU的未命中次数均达到最小值5，因此增加缓冲区容量可以减少miss rate

(3). LFU. 每次将访问次数最少的元素替换出去，可以在缓冲区中保留最常访问的元素，但需要记录访问次数，增加了额外开销

2.

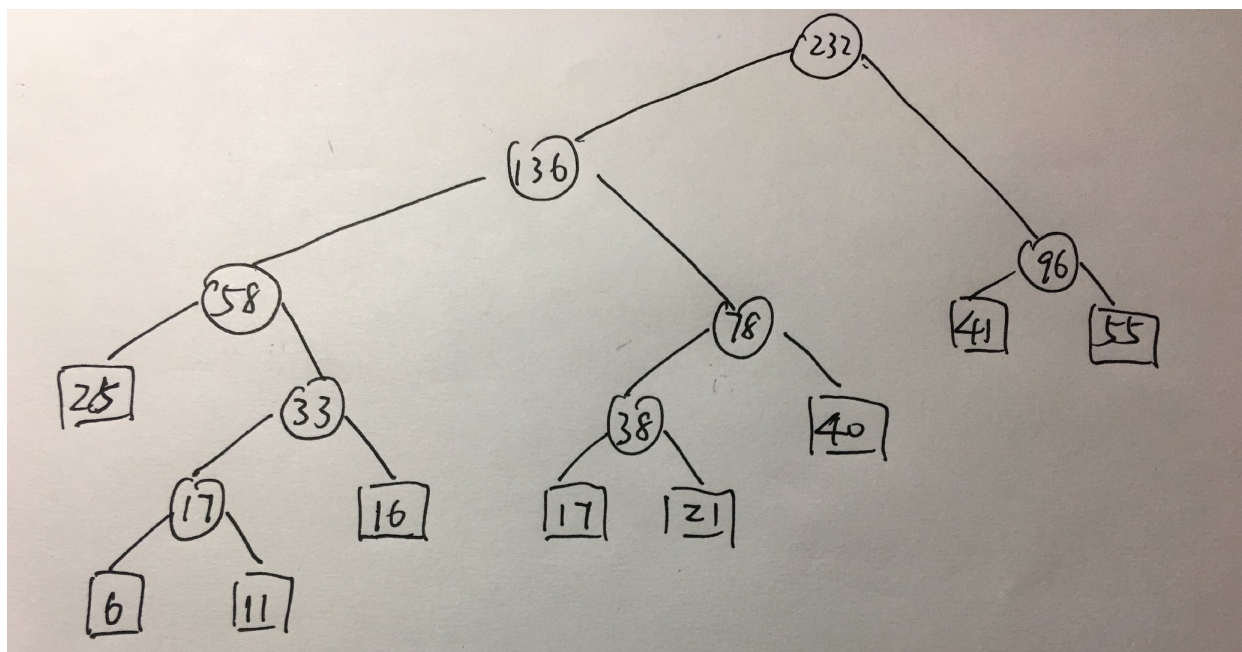
(1).



(2). 可以每次取M个元素进行排序，再将排序结果输出。这种方法时间复杂度与置换选择排序相近，但最多只能生成M个元素的顺串，而置换选择排序平均可生成2M个元素的顺串。

3.

(1).



(2).

每一趟读记录次数为圆形结点中的数值，总读记录次数为

$$3 * 25 + 5 * 6 + 5 * 11 + 4 * 16 + 4 * 17 + 4 * 21 + 3 * 40 + 2 * 41 + 2 * 55 = 688$$

(3).

n 个文件、 m 个初始顺串进行 k 路归并的总比较次数为 $\lceil \log_2 m / \log_2 k \rceil * (k - 1) * (n - 1)$, m 不变时, k 增大会导致比较次数增大, 减弱了归并次数减少带来的好处

4.

败者树只需要往根的方向与上层父节点所索引的关键码进行比较即可, 而赢者树需要涉及到相应兄弟节点的比较。与败者树相比, 赢者树需要从外存中额外读取兄弟结点的数据, 因此败者树访问外存次数比赢者树少。

5.

- 假设1MB内存全部用于堆空间, 则平均顺串长度为 $1\text{MB} * 2 = 2\text{MB}$
- 1MB内存可容纳1024个块, 因此可以进行1024路归并。至多进行两趟归并, 因此可以处理 $1024^2 = 2^{20}$ 个顺串
- 能够处理的最大文件长度为 $2\text{MB} * 2^{20} = 2^{31}\text{B}$