

索引

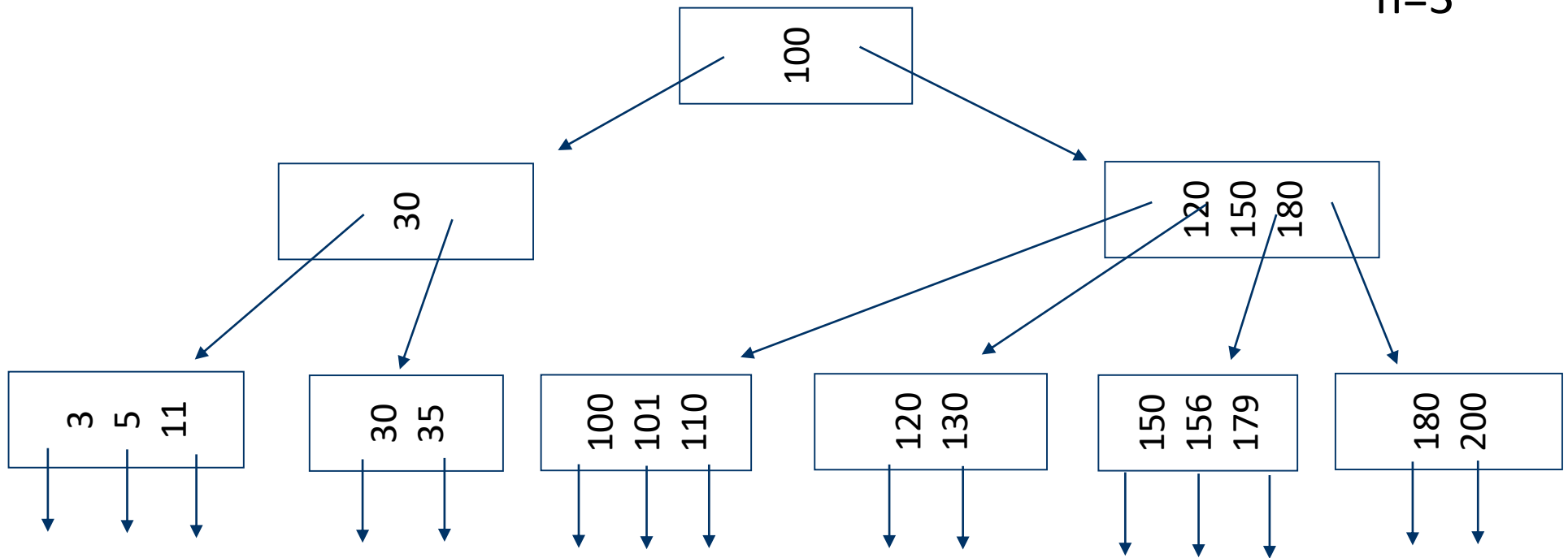


徐辰
华东师范大学
数据科学与工程学院
cxu@dase.ecnu.edu.cn

B-Tree

Root

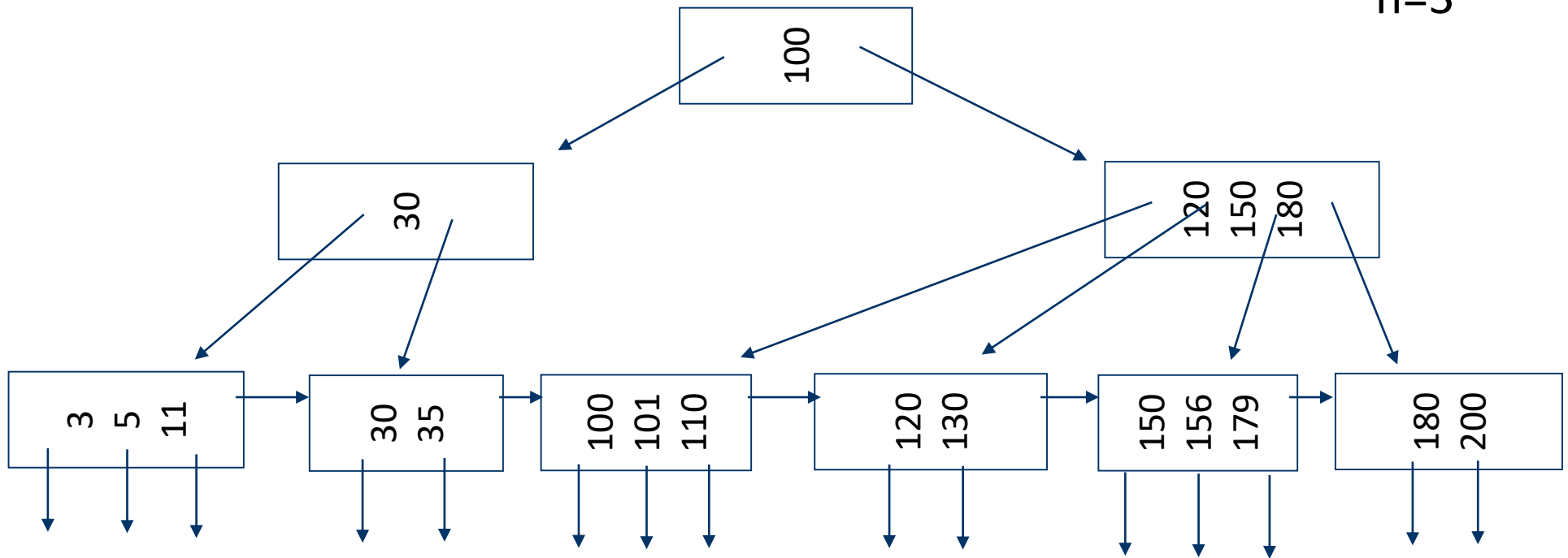
$n=3$



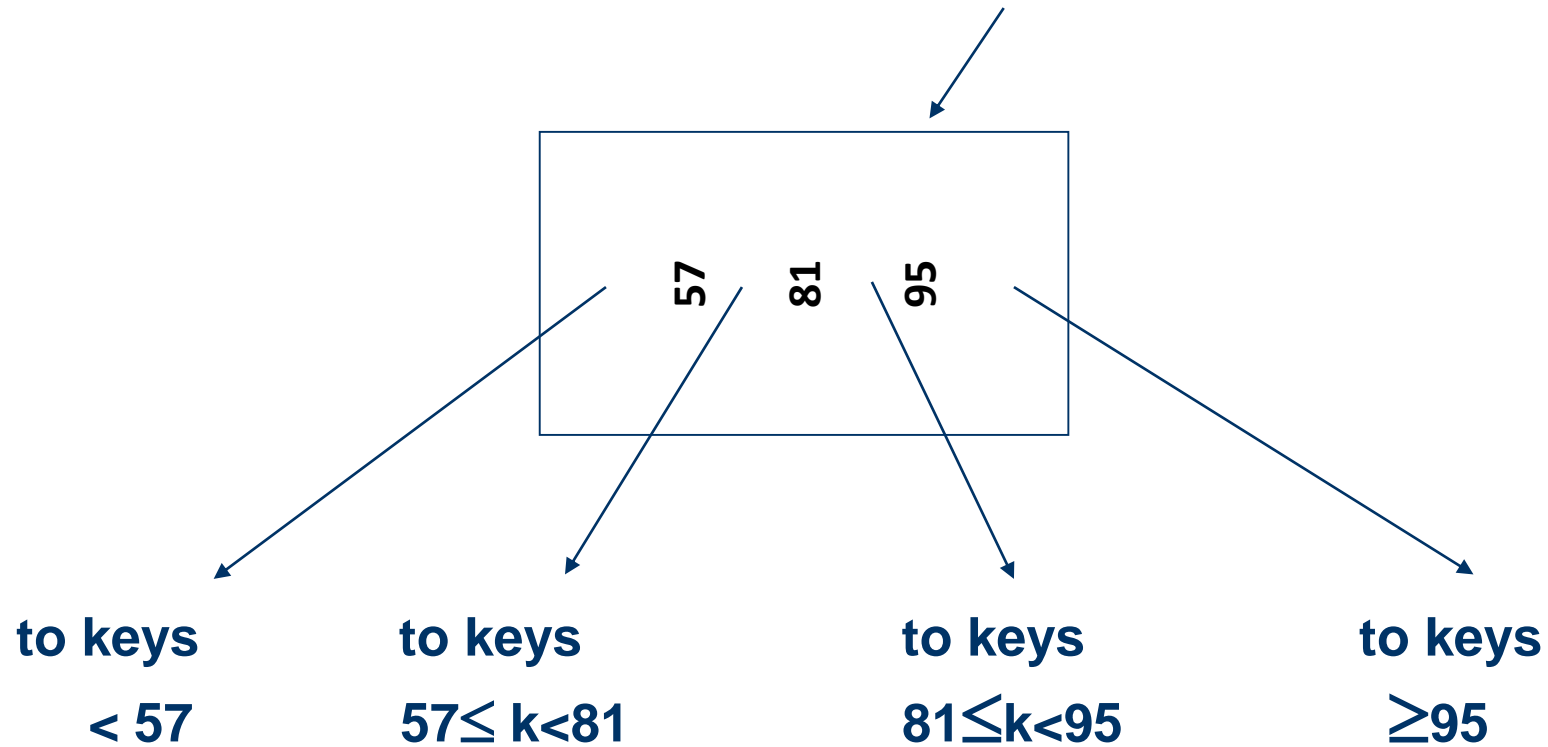
B-Tree

Root

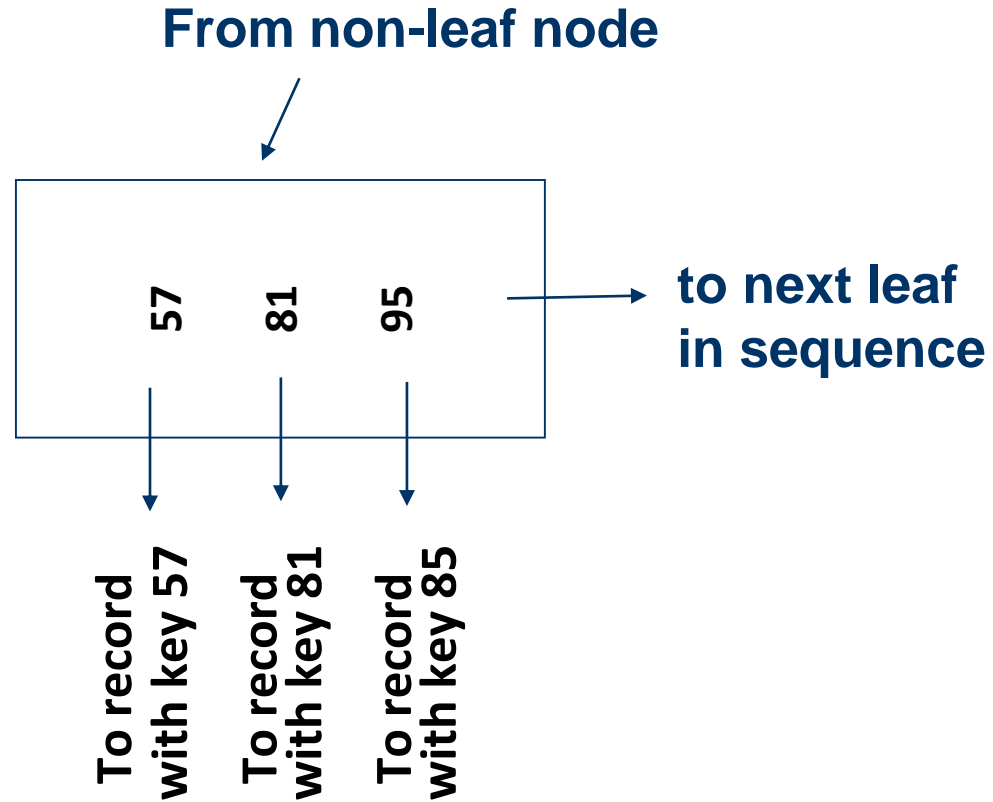
$n=3$



B-Tree的非叶节点



B-Tree的叶子节点



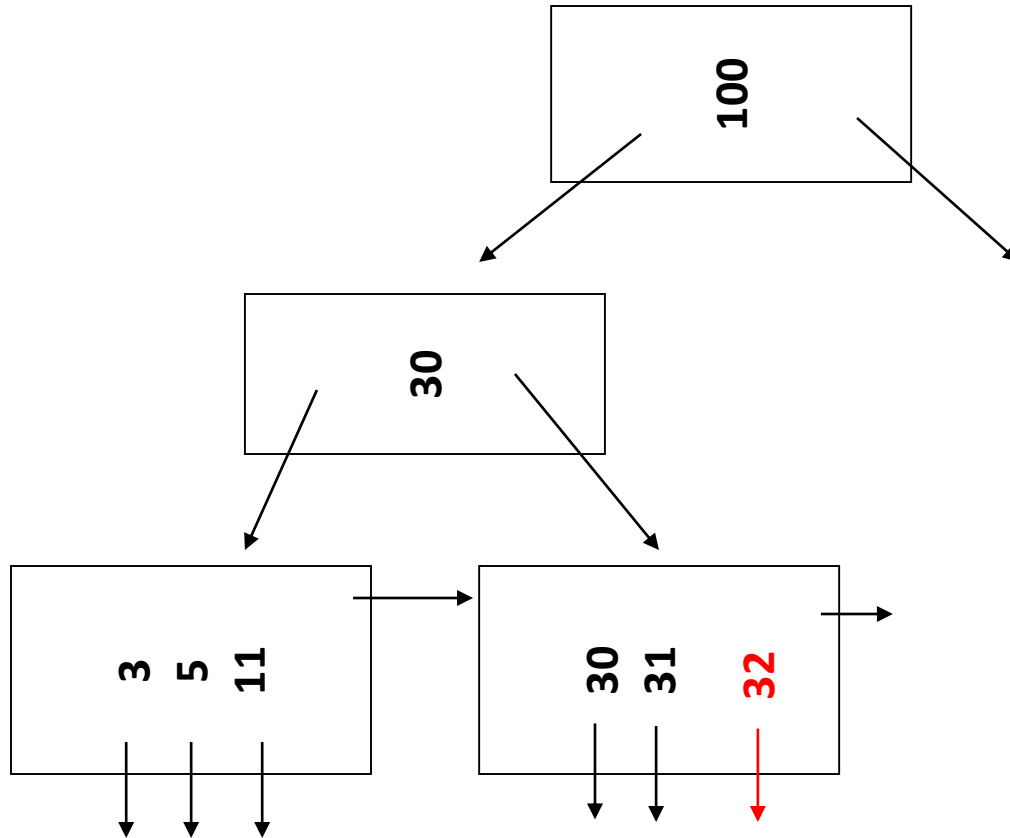
B-Tree的平衡性

- 每个节点的大小固定 (e.g. 4KB或8KB) , 因此最多只能容纳 n 个键和 $n+1$ 个指针。
 - 在添加数据的过程中, 如果超过 n , 则进行节点分裂。
- 除了根节点之外, 每个节点至少容纳 $\lceil (n+1)/2 \rceil - 1$ 个键和 $\lceil (n+1)/2 \rceil$ 个指针。
 - 在删除数据的过程中, 如果少于这个数, 则进行节点合并。

B-Tree 插入操作举例

(a) space available in leaf
- Insert key = 32

n=3

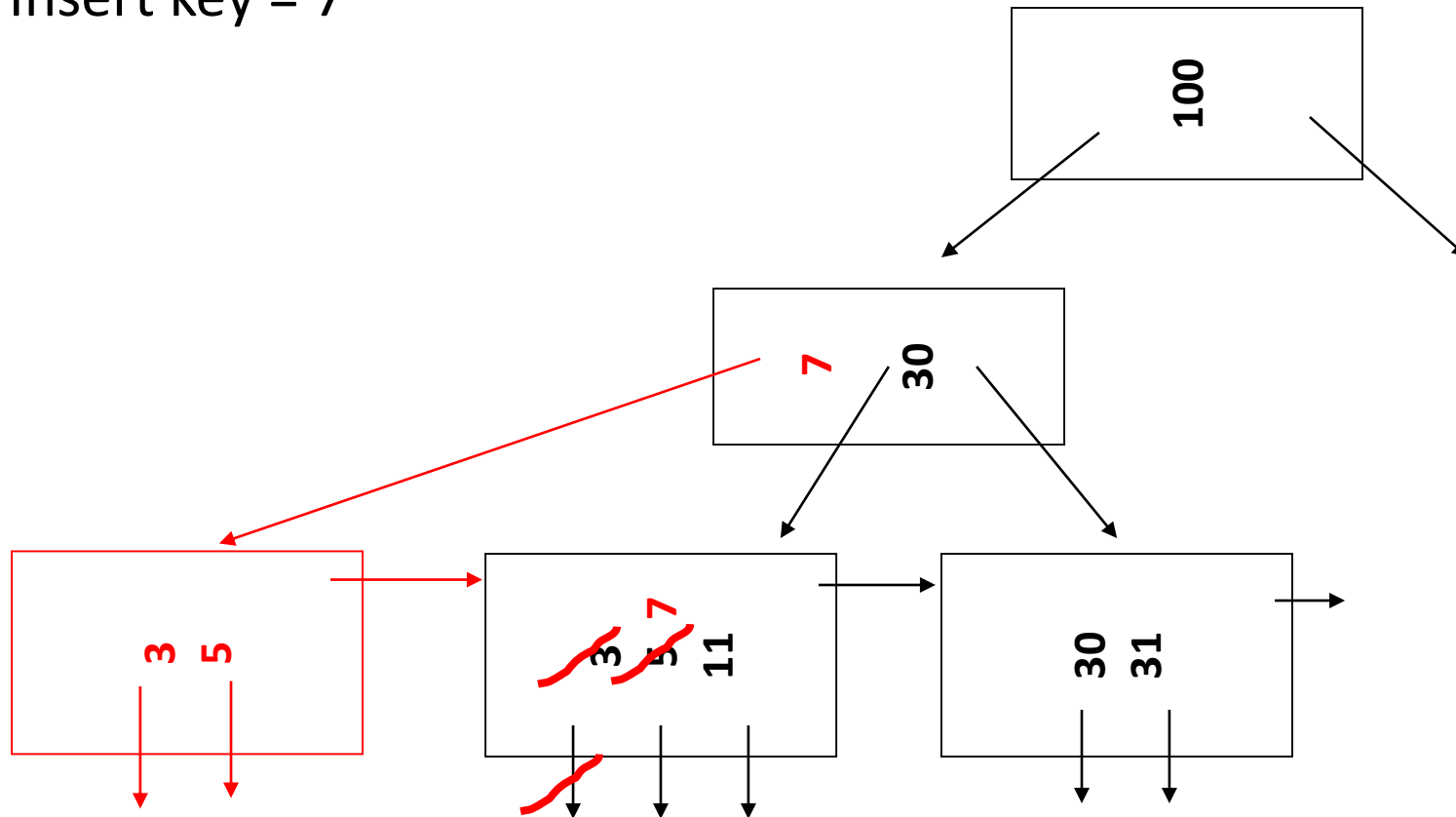


B-Tree 插入操作举例

(b) leaf overflow

n=3

- Insert key = 7

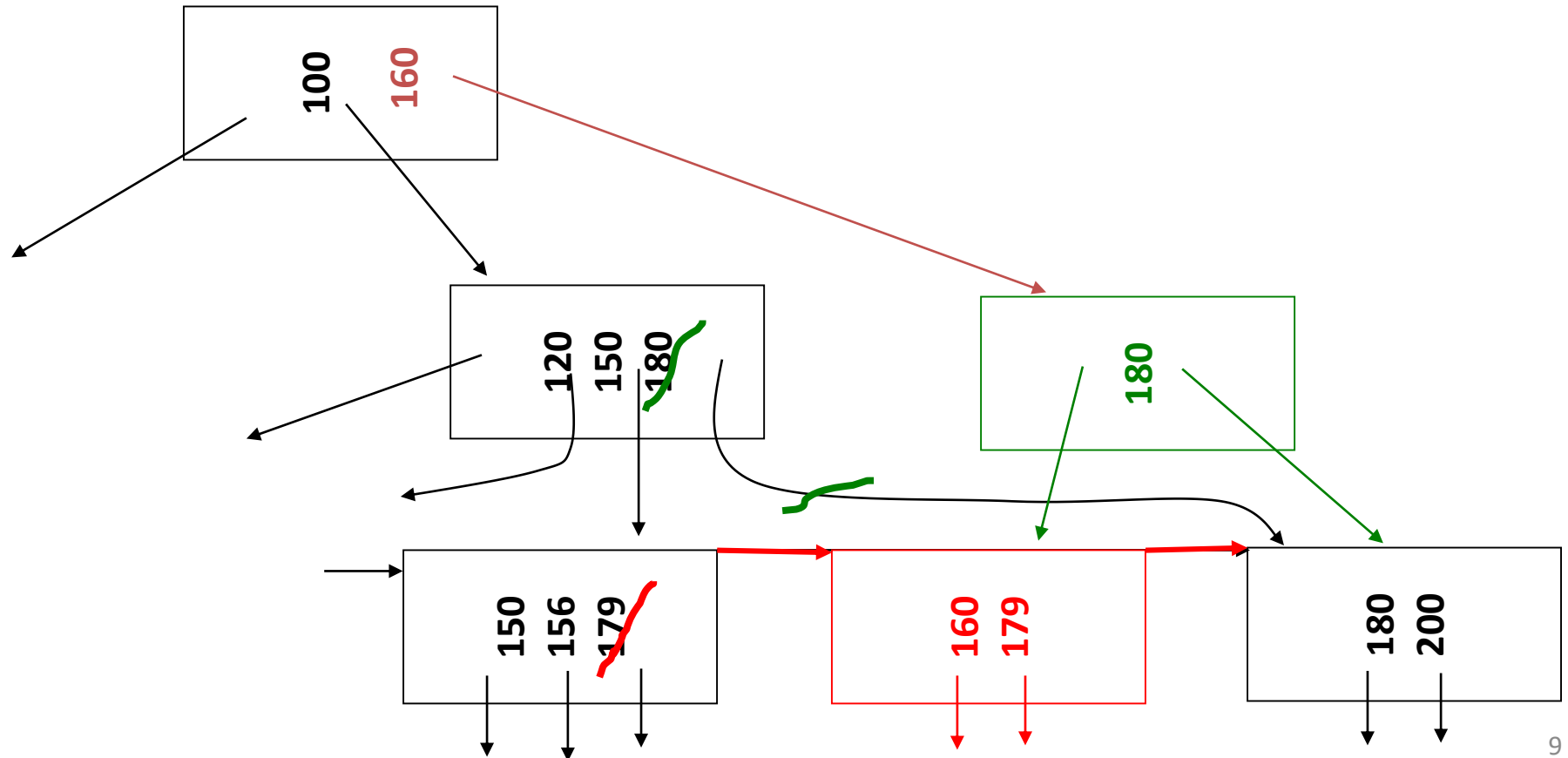


B-Tree 插入操作举例

(c) non-leaf overflow

- Insert key = 160

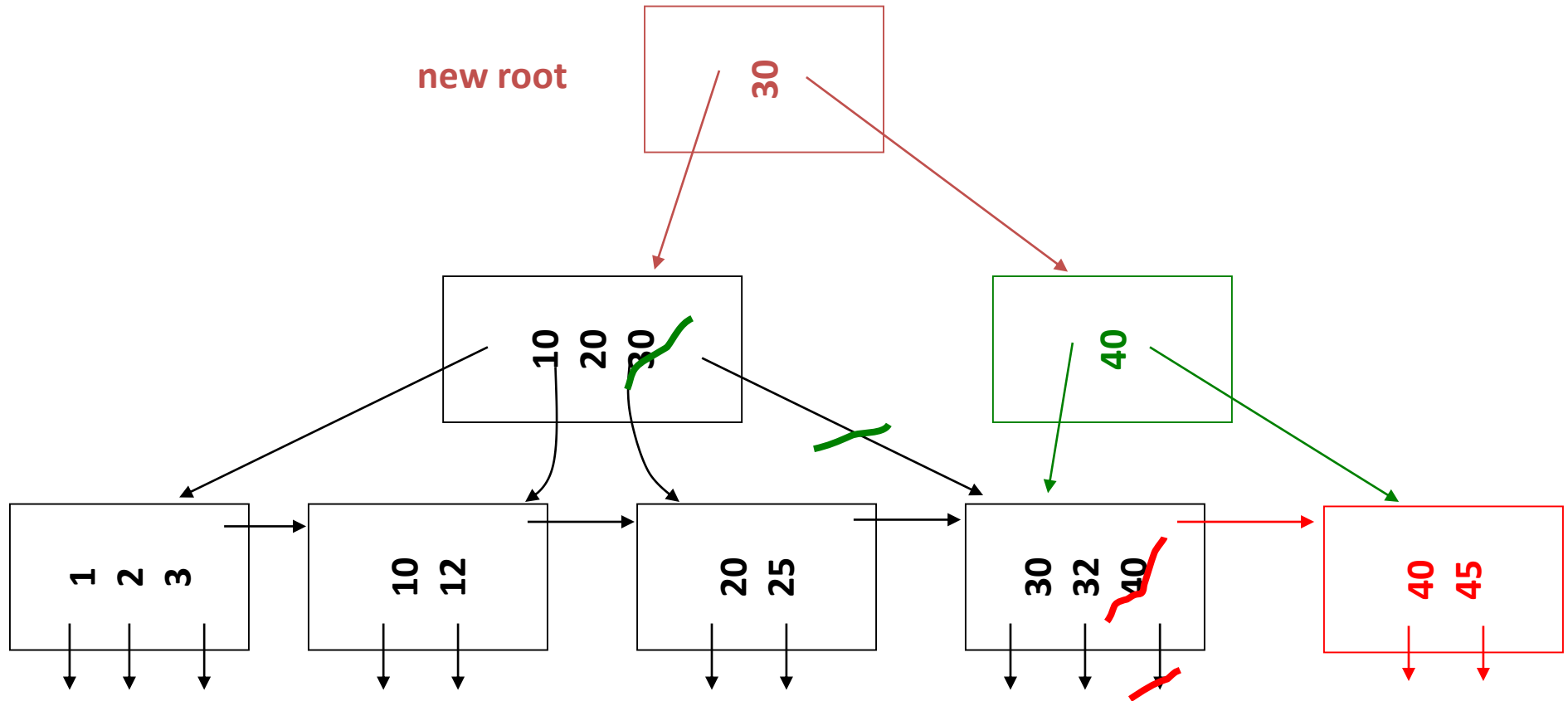
$n=3$



B-Tree 插入操作举例

(d) New root
- insert 45

n=3

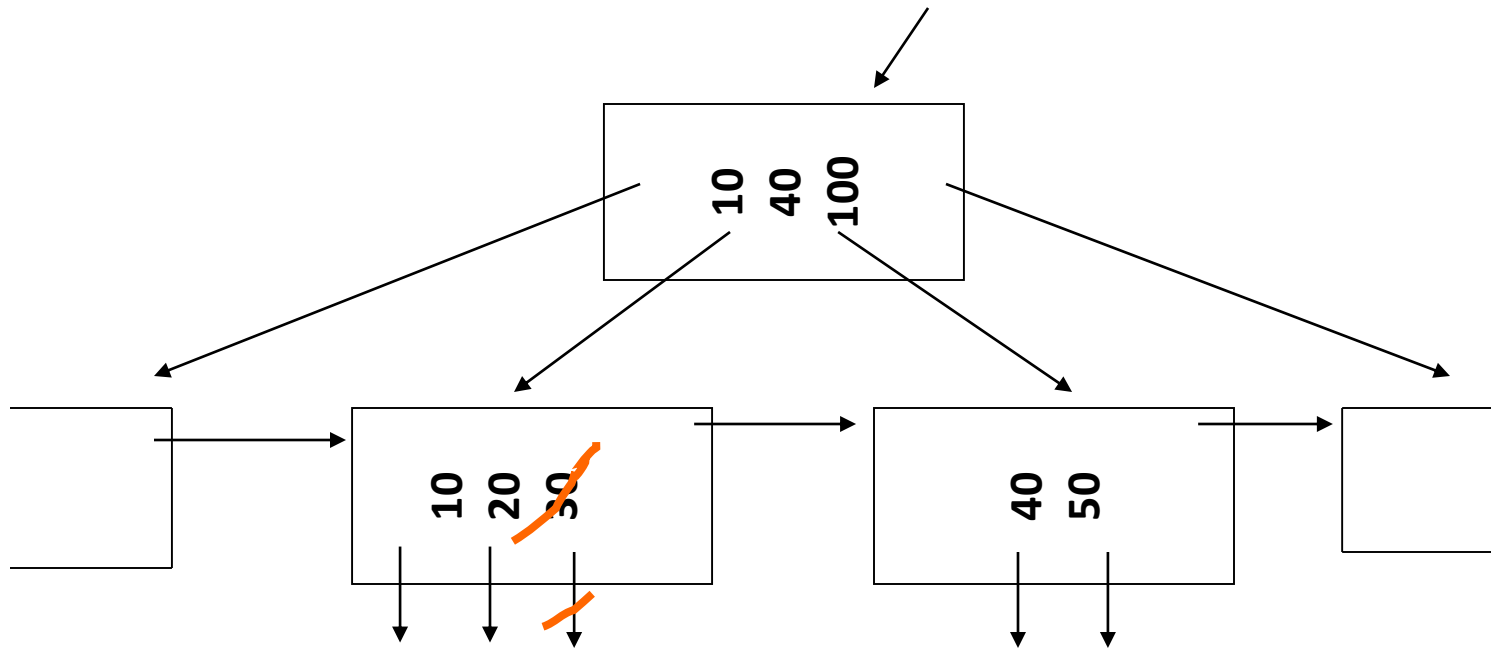


B-Tree 删除操作举例

(a) Simple case

– Delete 30

n=4

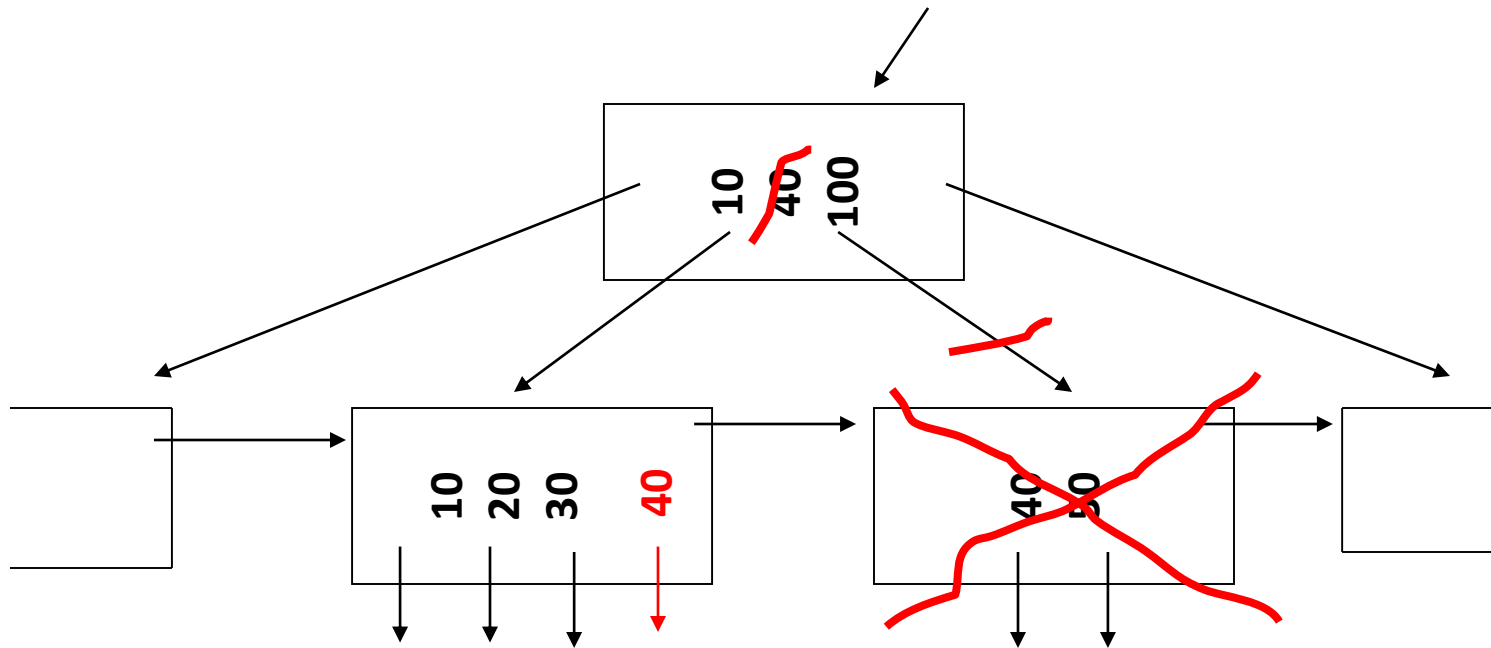


B-Tree 删除操作举例

(b) Coalesce with sibling 与兄弟节点合并

– Delete 50

n=4

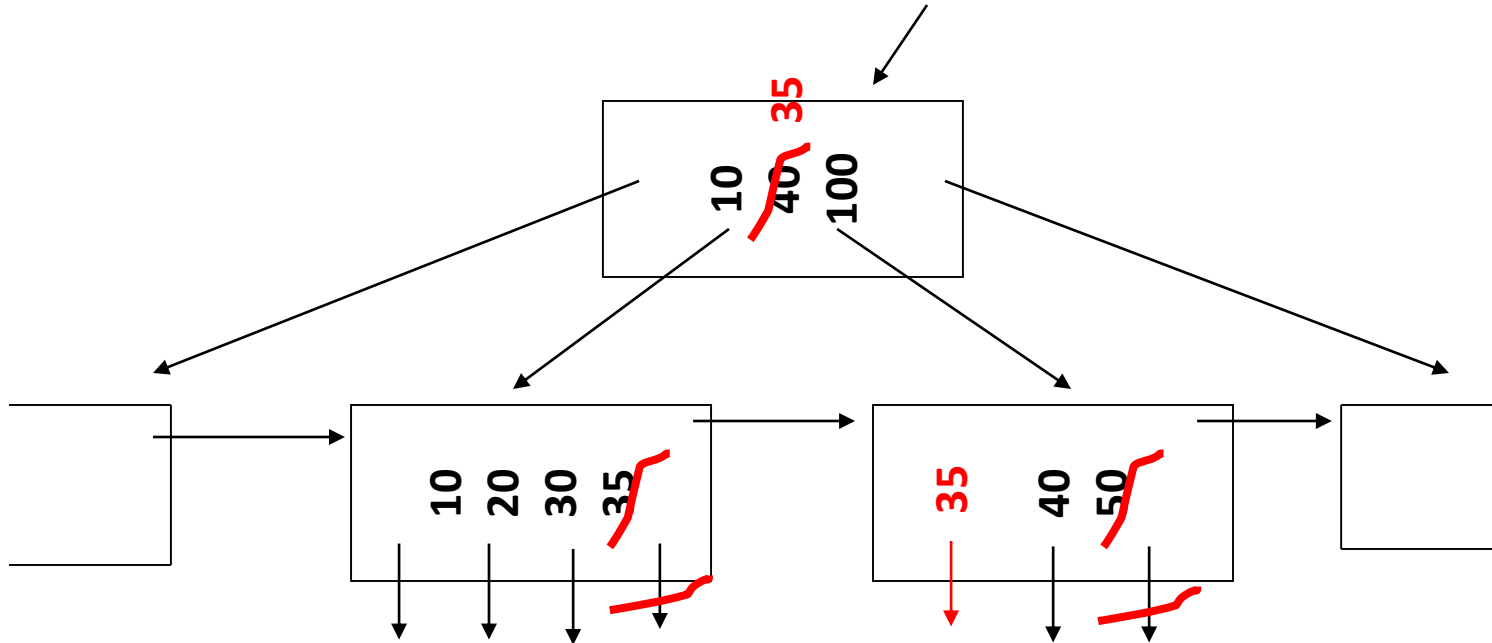


B-Tree 删除操作举例

(c) Redistribute keys

– Delete 50

n=4

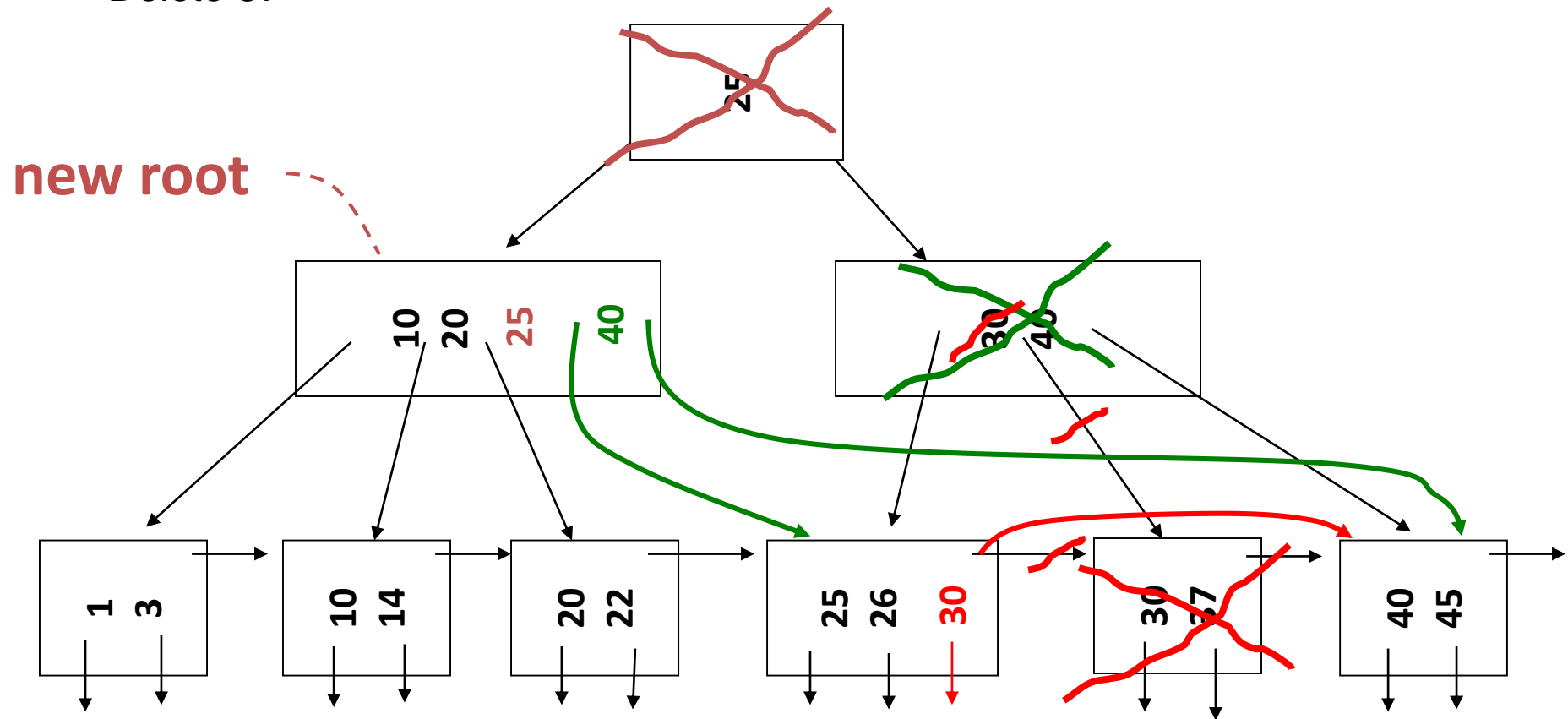


B-Tree 删除操作举例

(d) Non-leaf coalesce

n=4

– Delete 37



什么决定B-Tree的效率?

- 树的高度决定查询需要I/O次数。
- 对于同样规模的数据，n越大树的高度越低。
- 为了提高B-Tree的效率，我们需要增加n，即增加每个节点容纳键和指针的数量。
 - 用简短的数据类型定义键的属性，e.g. smallint.
 - 对B-Tree进行压缩。

B-tree的历史

- 发明人：Rudolf Bayer(with Edward M. McCreight)
 - 慕尼黑工业大学教授
 - 1969年投稿-1971年发表：历程坎坷
- 其它贡献
 - Red-black tree
 - UB-tree (with Volker Markl)
- Rudolf Bayer – Volker Markl – Chen Xu ☺

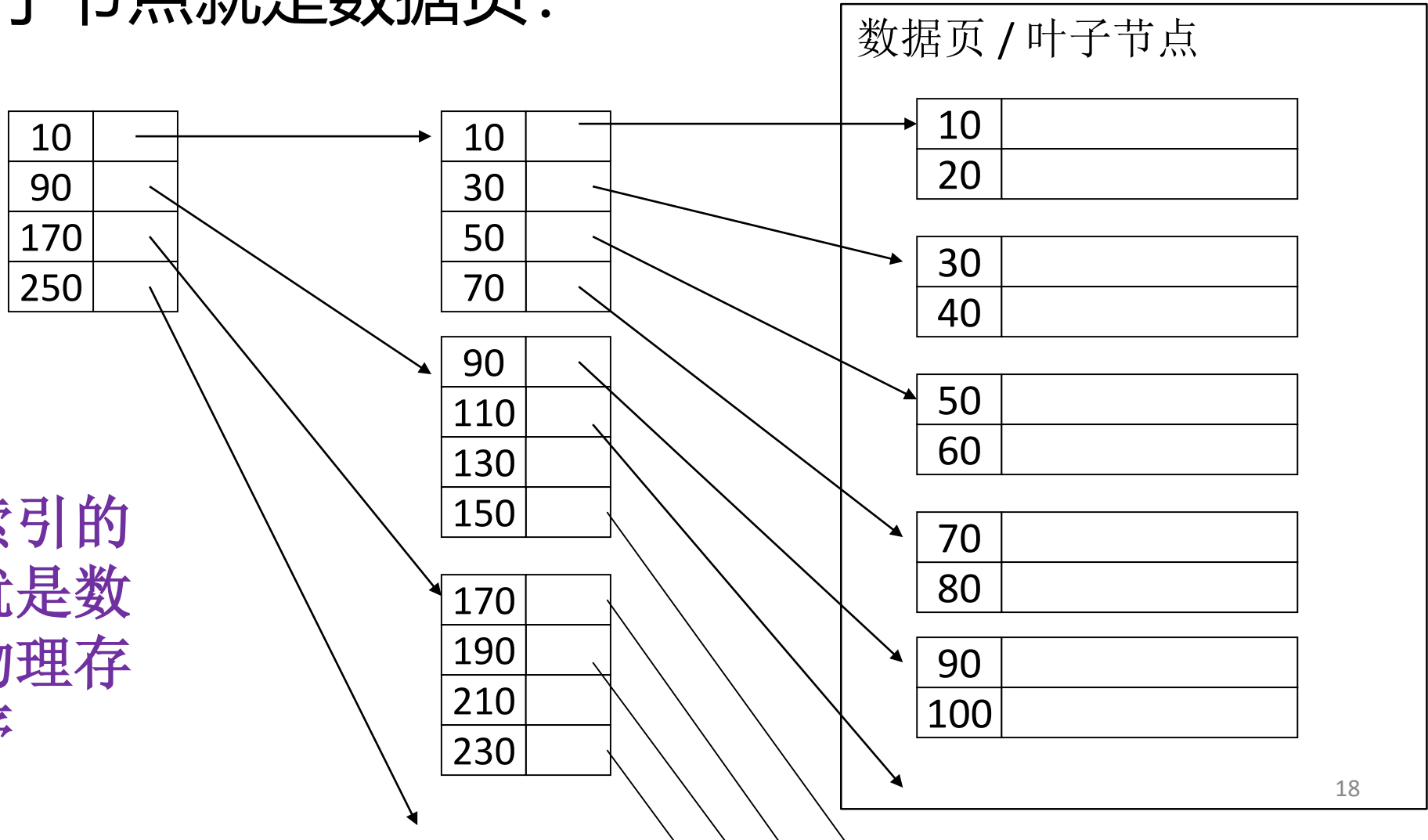


思考题

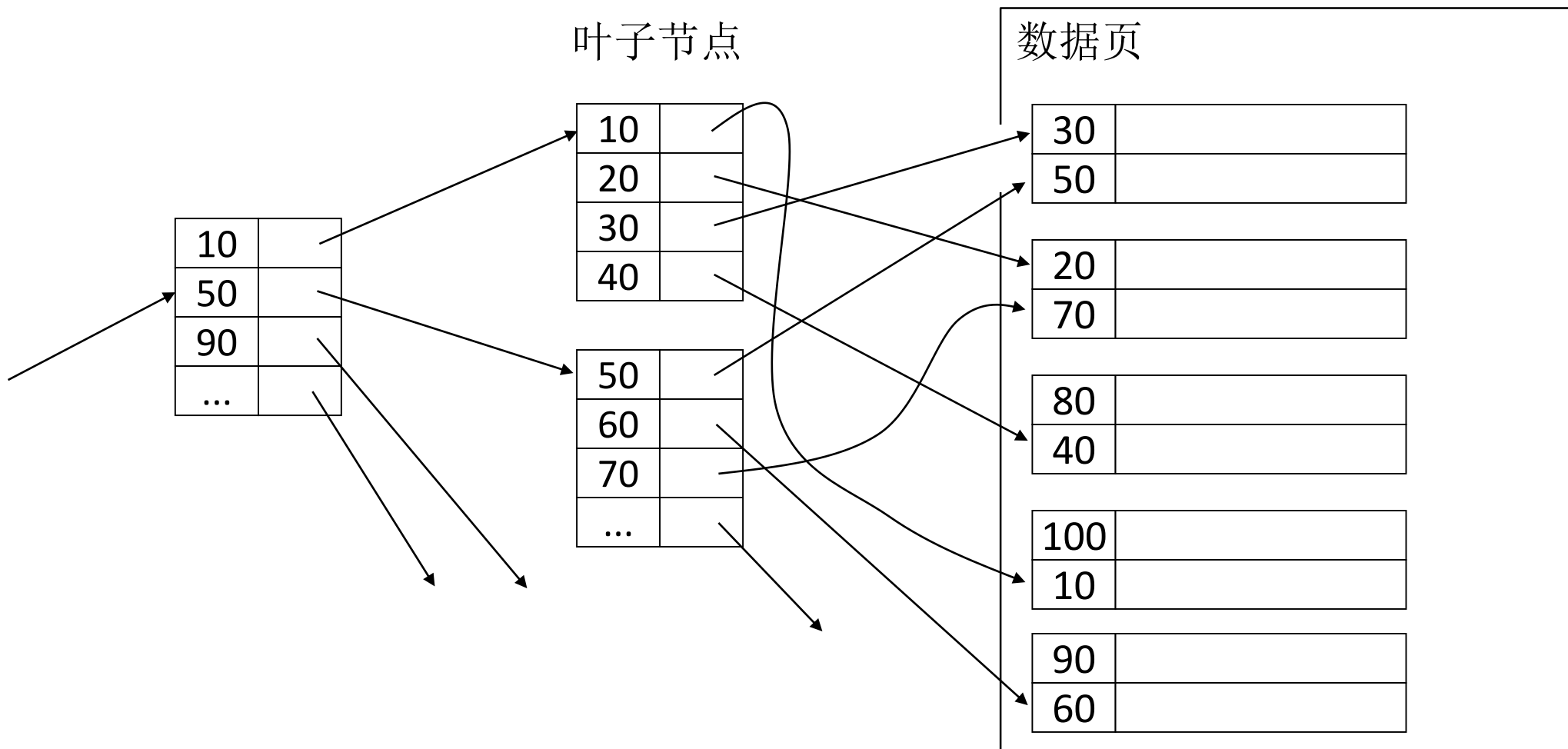
- B-tree中B的含义是什么？
- B-tree是一维的，高维数据索引怎么办？

聚簇索引 (Clustered Index)

- 叶子节点就是数据页：



辅助索引 (Secondary Index)



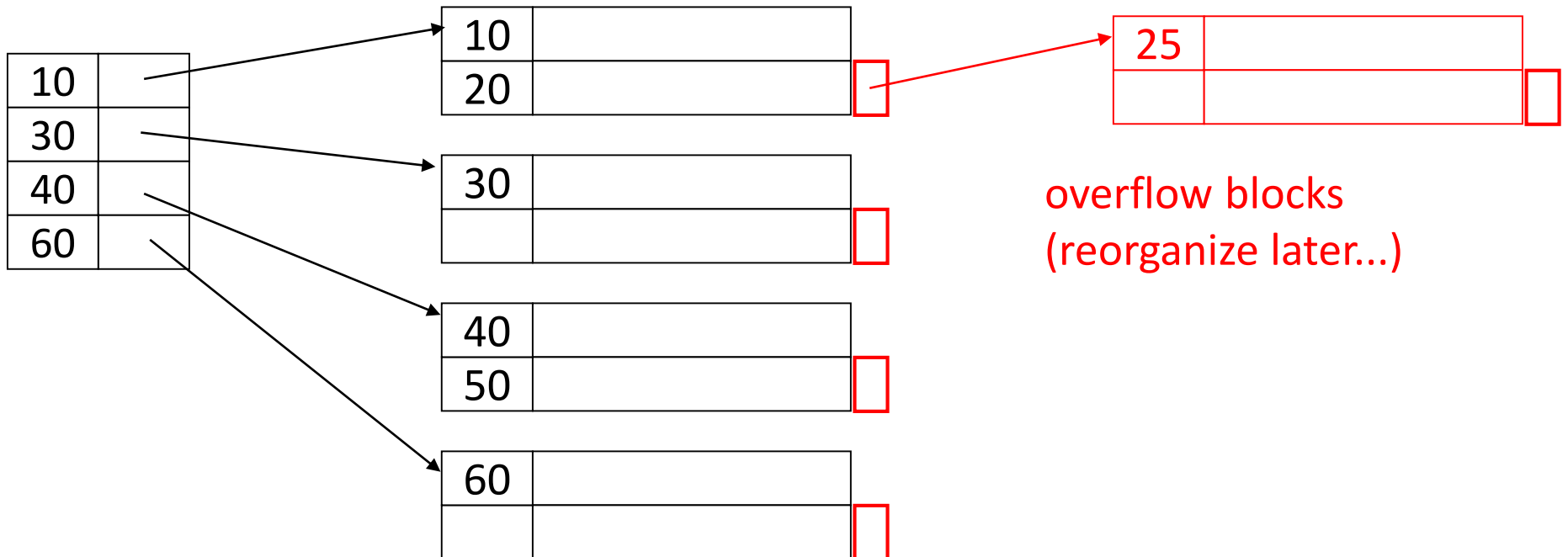
关于聚簇索引

- 聚簇索引比辅助索引更快。
- 通常一张关系表只能有一个聚簇索引。（如果定义了主码，主码使用的索引为聚簇索引。）
- 其余的索引都为为辅助索引。



聚簇索引的溢出

– insert record 25

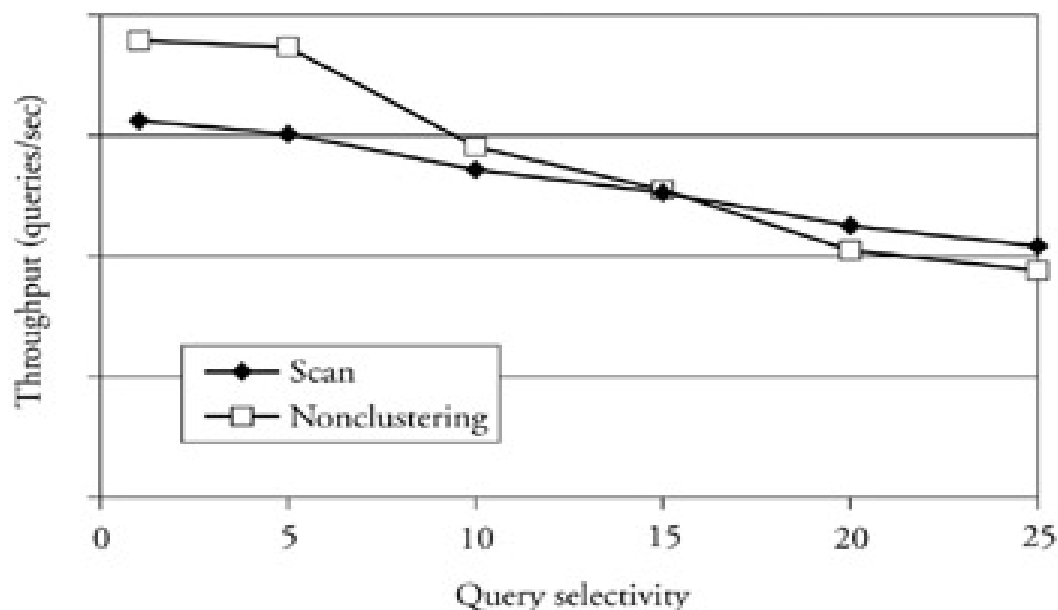


创建辅助索引的原则

- 以下属性上适合创建：
 - 经常出现在选择条件中的属性；
 - 耗时太长的查询所涉及的属性。
- 若查询满足以下条件，可考虑让辅助索引Cover查询所涉及的所有属性：
 - 常用或关键路径上的查询；
 - 查询所涉及的属性是整个关系表的一小部分。
- 辅助索引的代价：
 - 额外的存储空间；
 - 增加数据更新的代价。

辅助索引的效率与查询选择率有关

- 假设关系表中有 N 个记录，其中 K 个记录满足查询 Q 的条件，那么 K/N 为查询 Q 的选择率 (Selectivity)。
- 假设 N 个记录存放在 M 个数据页中，如果 $M < K$ ，辅助索引对 Q 则毫无意义。



总结

- 决定数据库的性能
 - 数据移动的代价和频率
- 提升数据库的性能
 - 减少数据移动
 - 提高数据访问局部性

Credits

- Slides from
 - 周烜