附加内容：B 树

初航，我带你；远航，靠自己

胡船长

B 树 – 结构定义

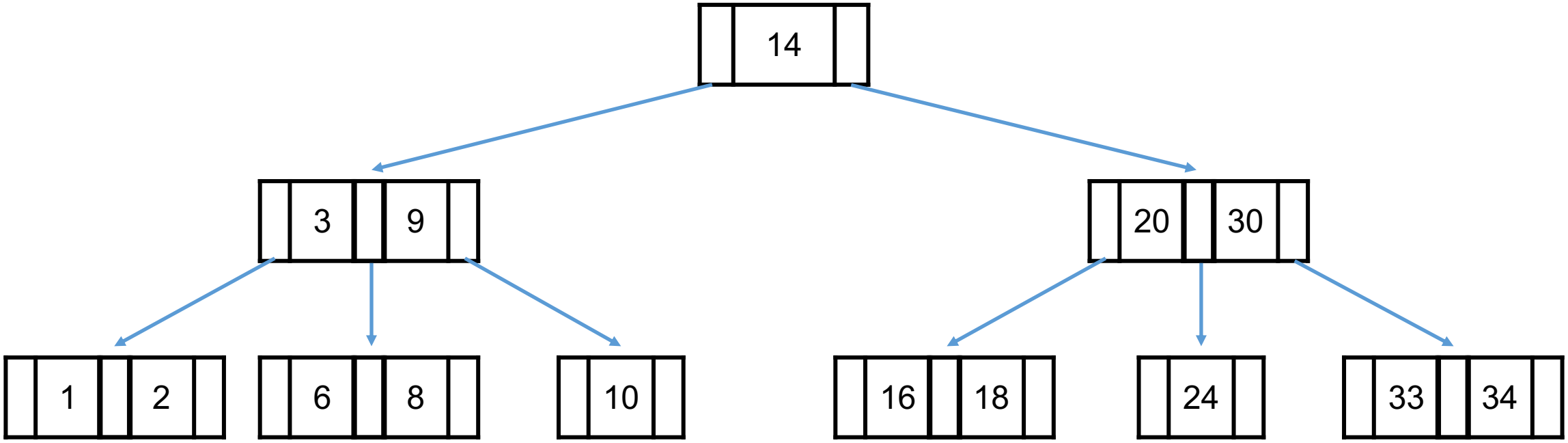
一棵 m 阶 B 树，需要满足下列特性：

1. 树中每个节点，最多含有 m 棵子树
2. 若根节点不是叶子节点，则至少有 2 棵子树
3. 除根结点之外的所有非终端结点至少有[m/2]棵子树
4. 如果一个结点有n-1个关键字，则该结点有n个分支，且这n-1个关键字按照递增顺序排列
5. 每个结点的结构为：（ n，A_0，K_1，A_1，K_2，A_2，… ，K_n，A_n ）
6. 非根结点中关键字的个数 n，满足：[m/2]-1 ≤ n ≤ m-1
7. 所有叶子节点处在同一层

B 树 – 结构定义

m 阶 B 树的性质解读：

1. B 树中只有根节点没办法满足拥有至少 [m/2] 棵子树的条件，其他节点均能满足
2. B 树是一种高度平衡的树形结构，比 AVL 树结构上更优美



1. 元素插入

2. 插入调整

3. 元素删除

4. 删除调整

1. 元素插入

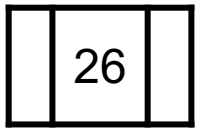
2. 插入调整

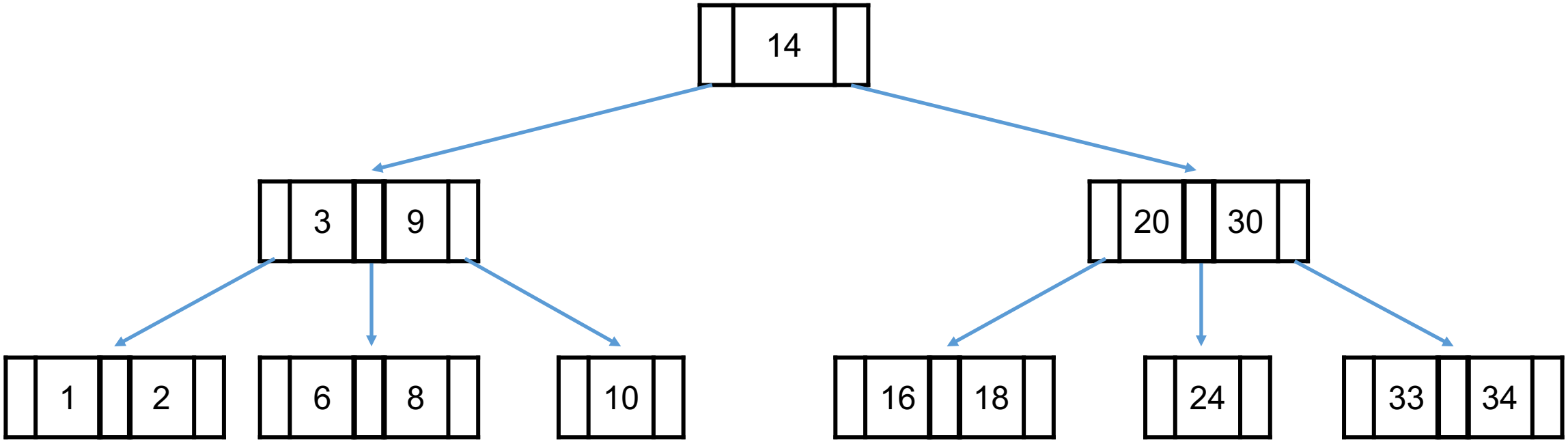
3. 元素删除

4. 删除调整

B 树 – 元素插入

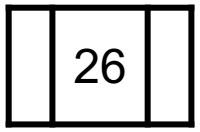
插入：

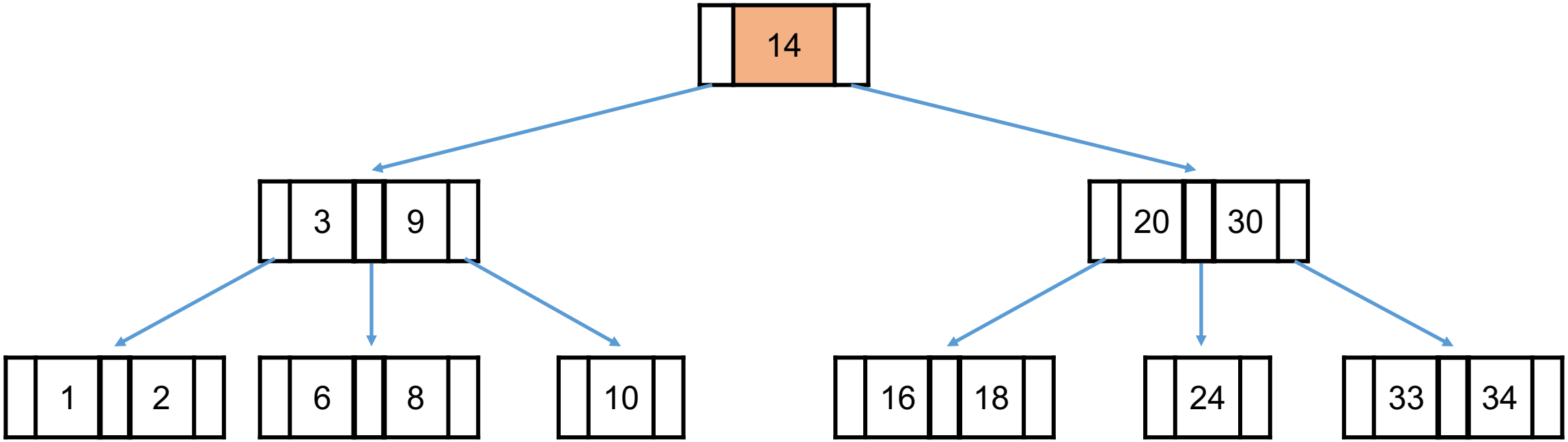




B 树 – 元素插入

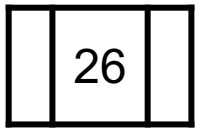
插入：

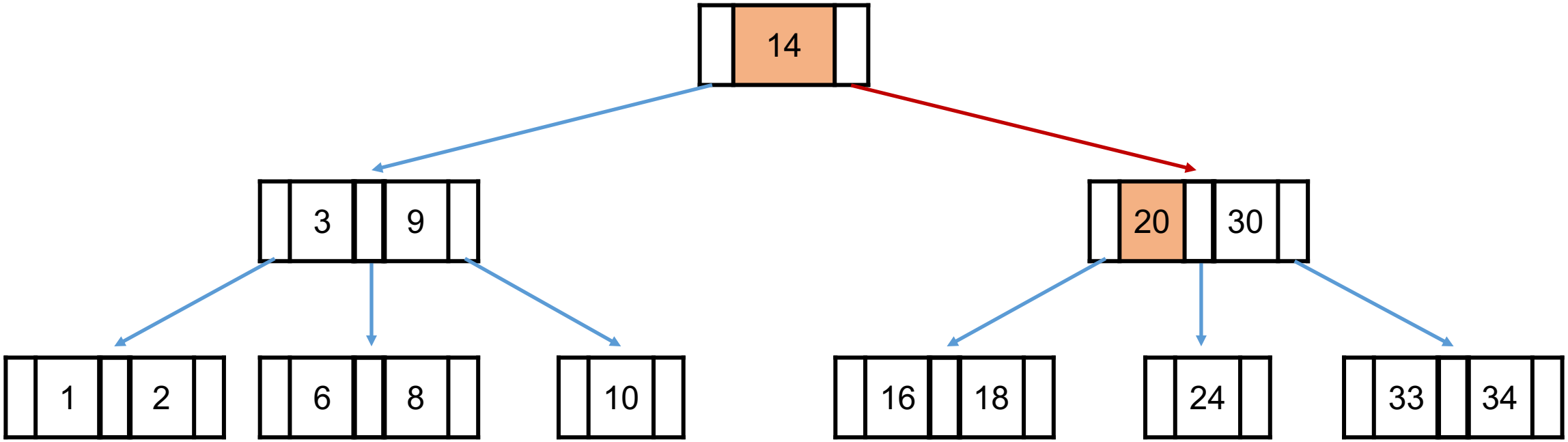




B 树 – 元素插入

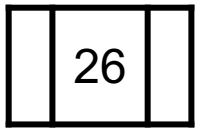
插入：

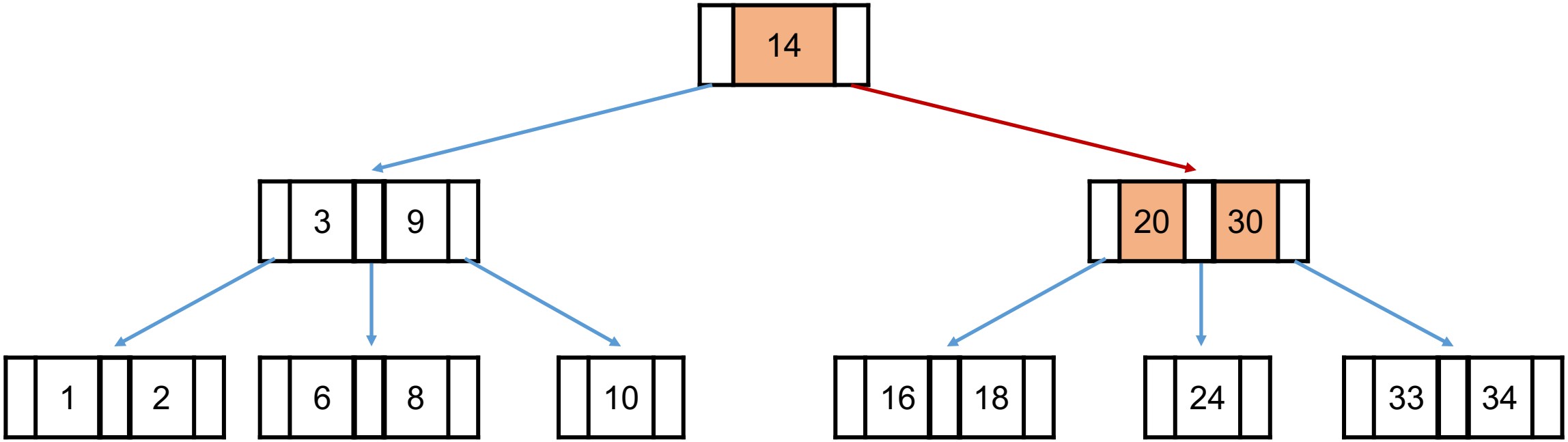




B 树 – 元素插入

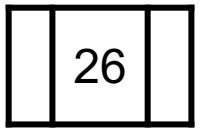
插入：

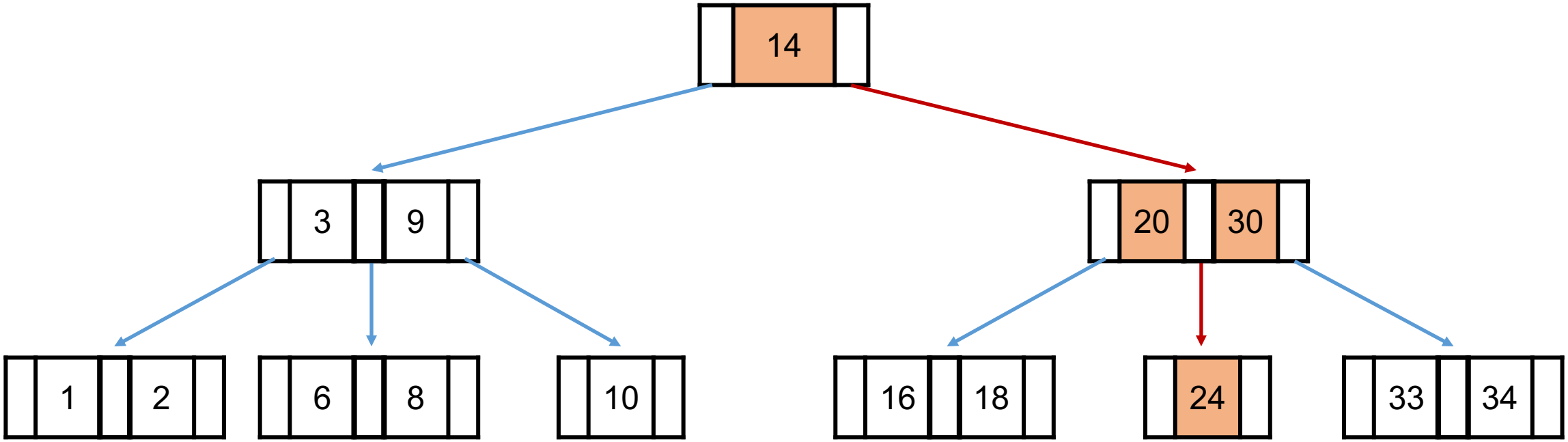




B 树 – 元素插入

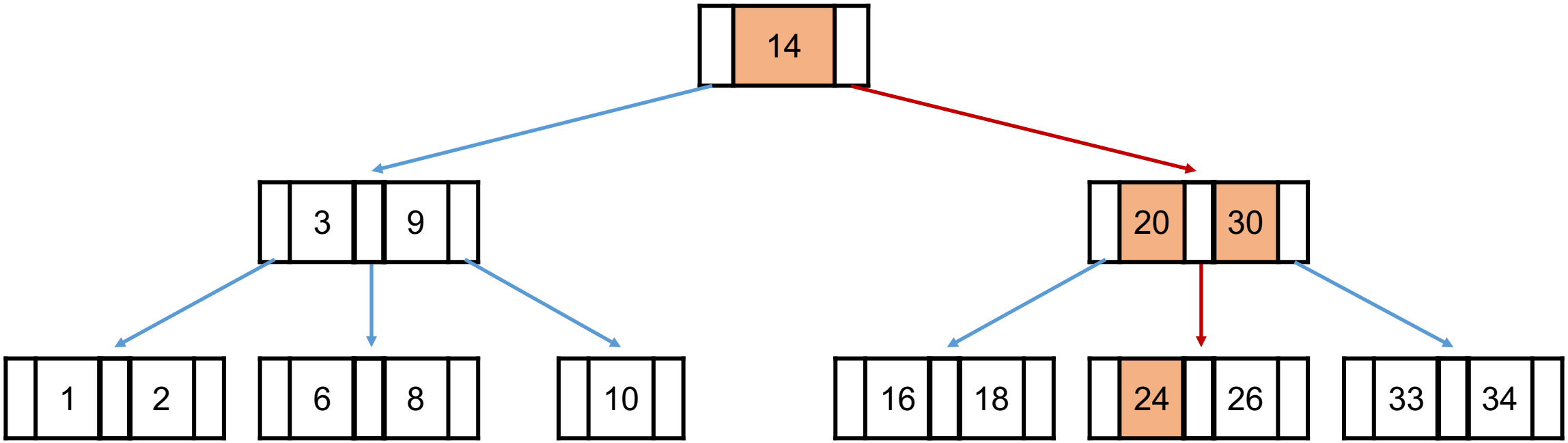
插入：



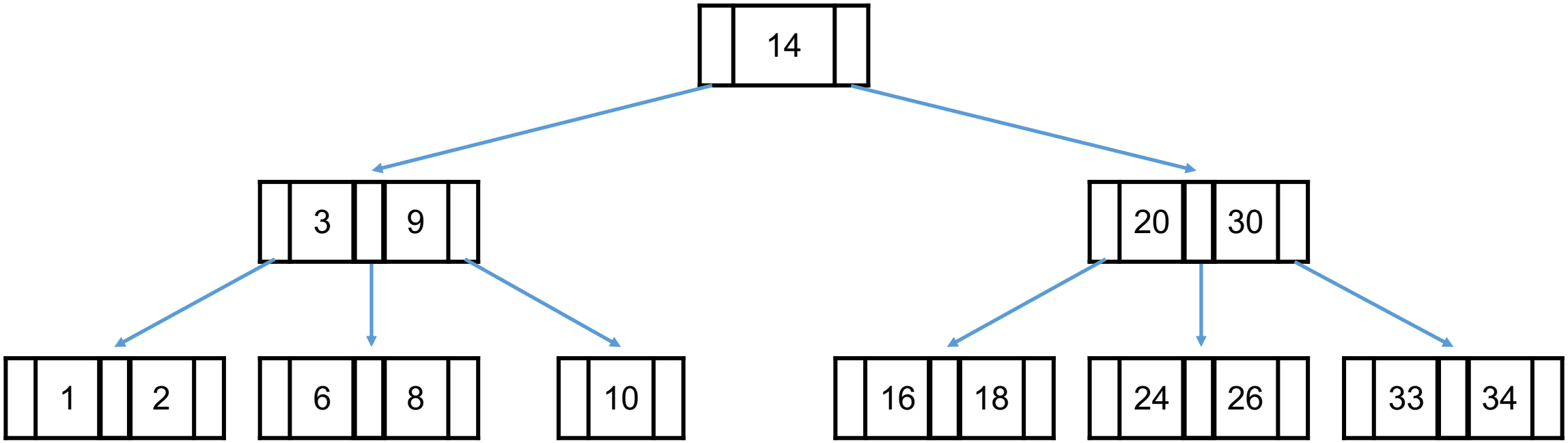


B 树 – 元素插入

插入：



B 树 – 元素插入



~~1. 元素插入~~

2. 插入调整

3. 元素删除

4. 删除调整

~~1. 元素插入~~

2. 插入调整

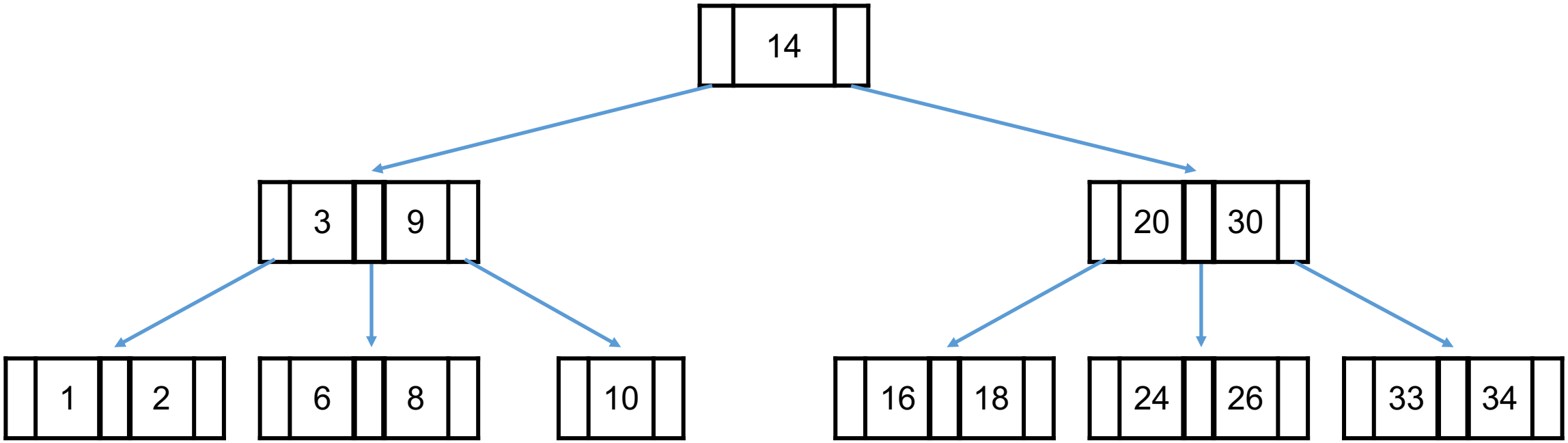
3. 元素删除

4. 删除调整

B 树 – 插入调整（上溢）

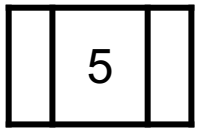
m 阶 B 树的插入调整:

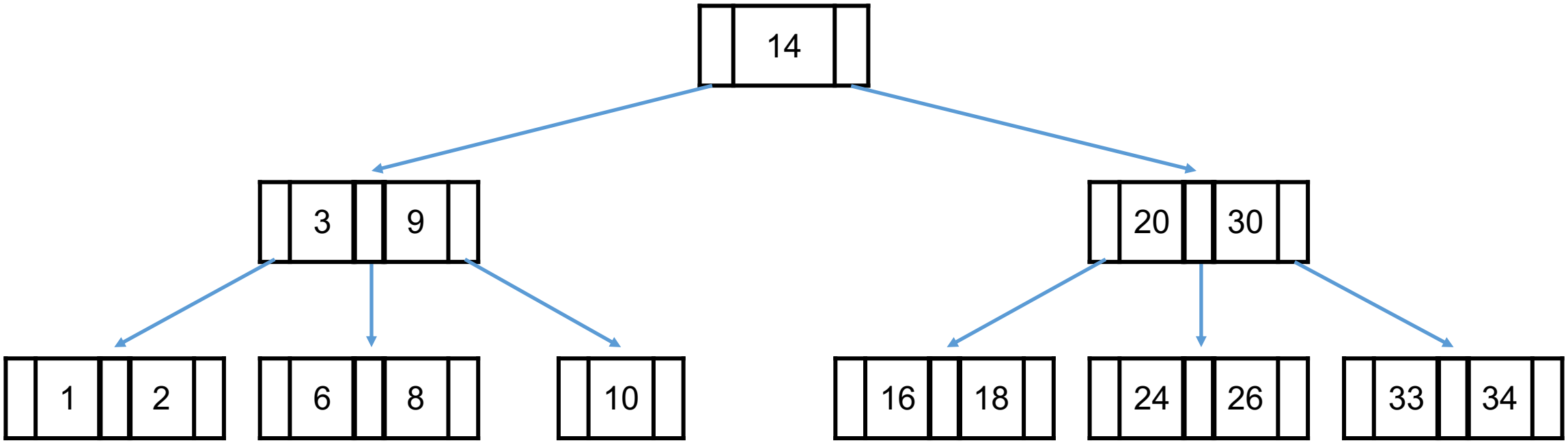
1. 插入调整站在父节点处理，发生在节点关键字数量达到 m 时
2. 插入调整的核心操作是：节点分裂



B 树 – 插入调整（上溢）

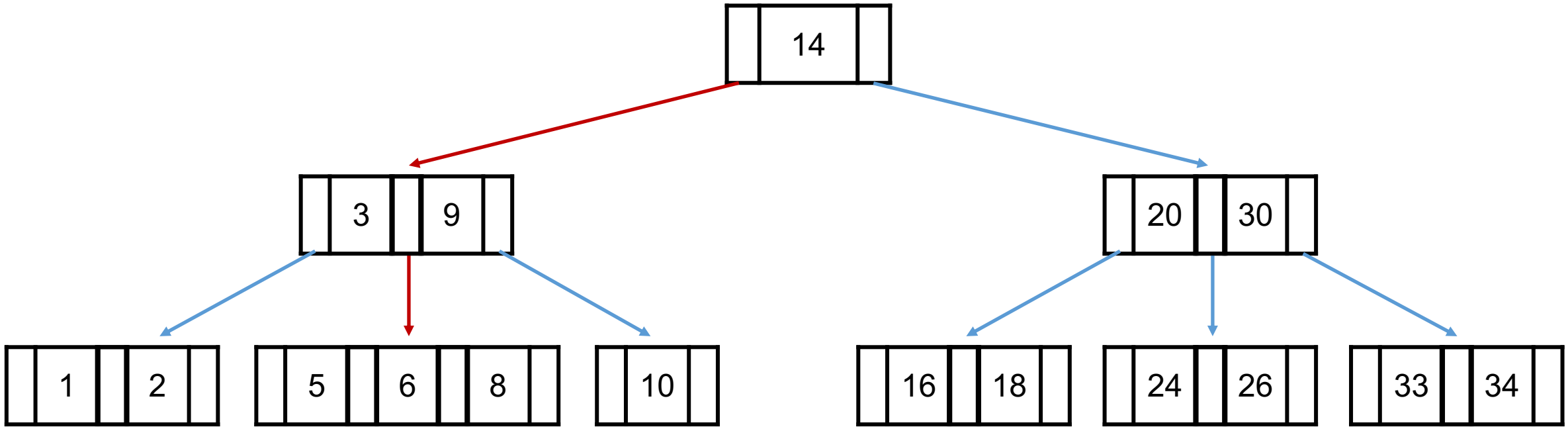
插入：





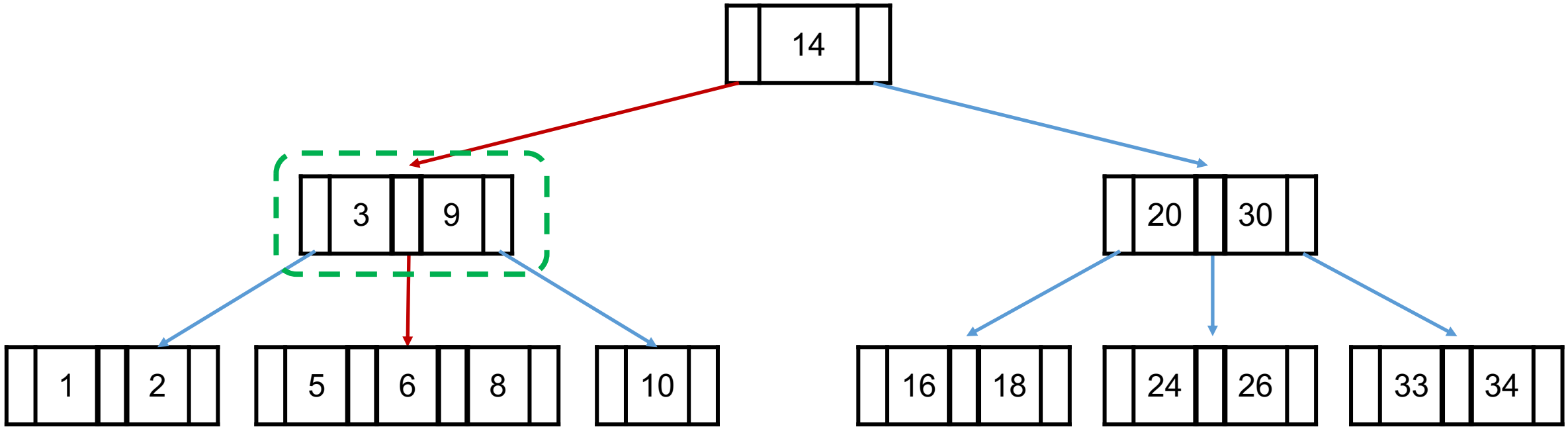
B 树 – 插入调整（上溢）

插入：



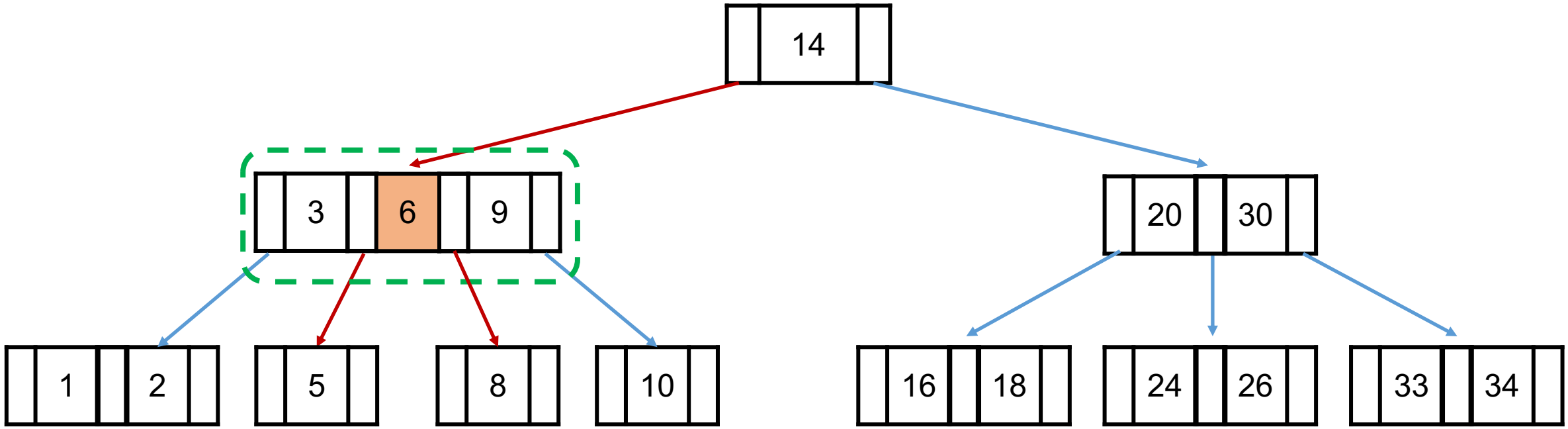
B 树 – 插入调整（上溢）

站在父节点处理



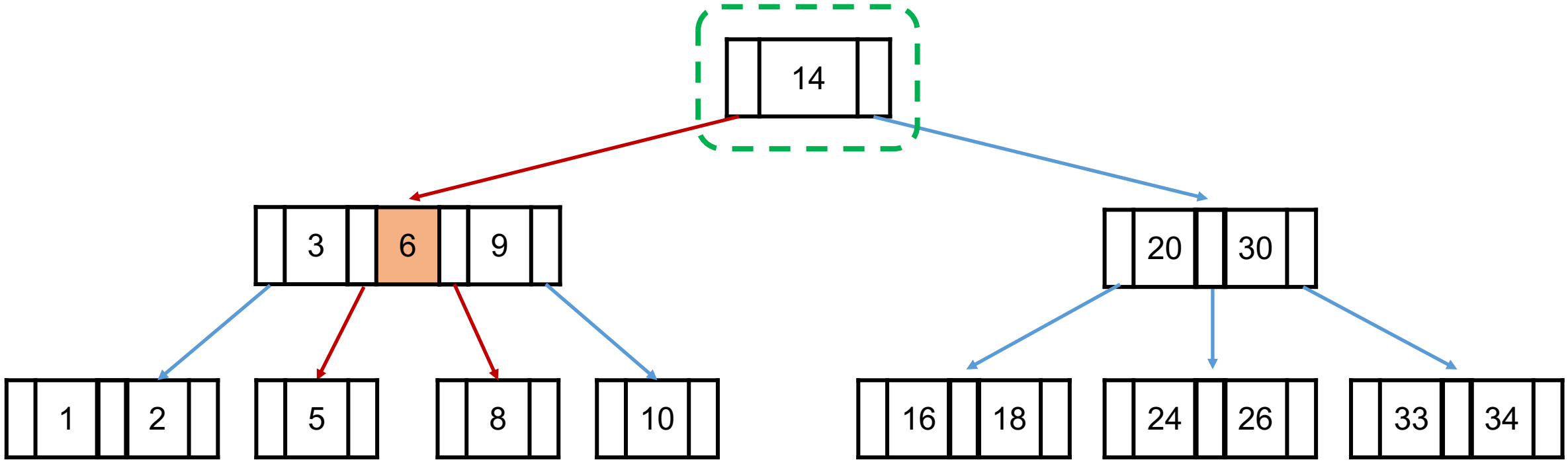
B 树 – 插入调整（上溢）

节点分裂



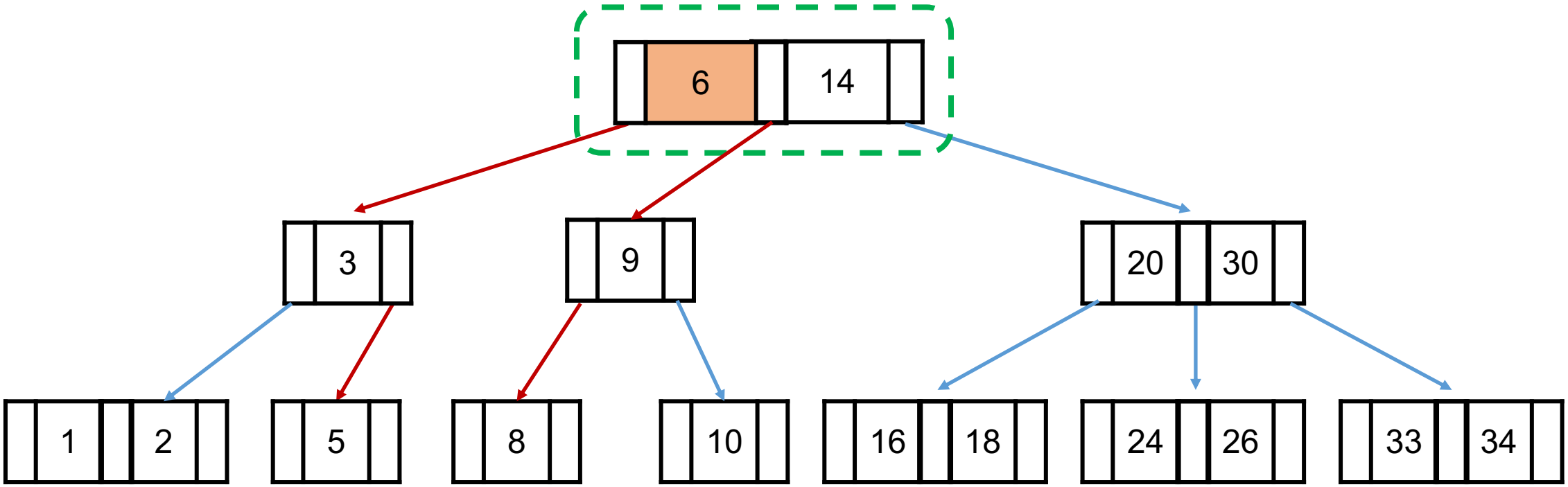
B 树 – 插入调整（上溢）

站在父节点处理

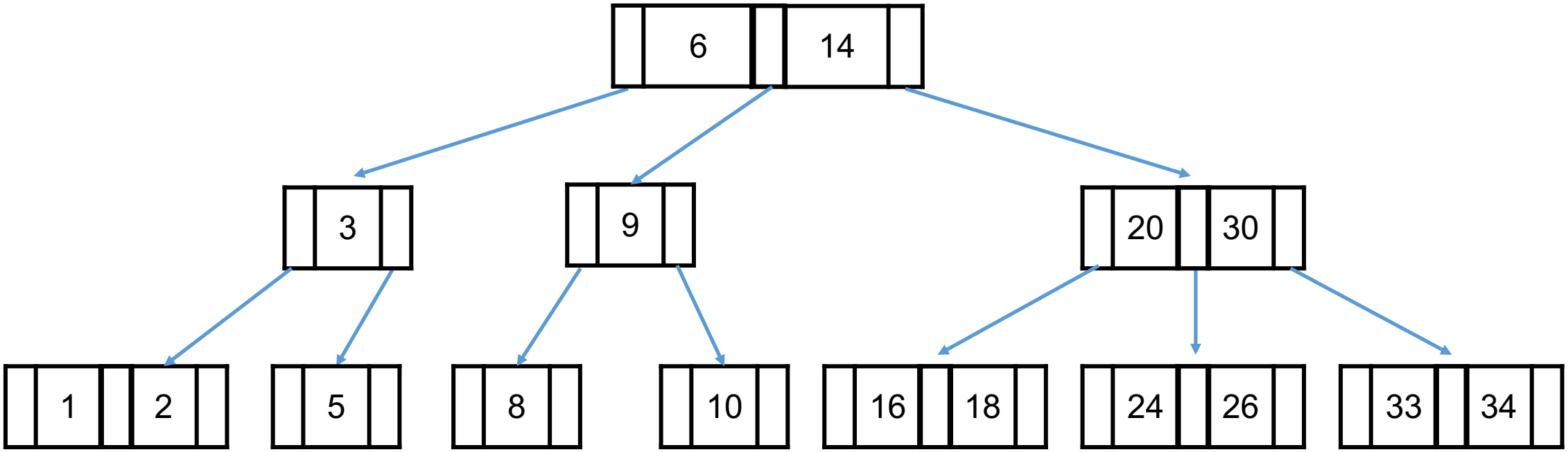


B 树 – 插入调整（上溢）

节点分裂



B 树 – 插入调整（上溢）



~~1. 元素插入~~

~~2. 插入调整~~

3. 元素删除

4. 删除调整

~~1. 元素插入~~

~~2. 插入调整~~

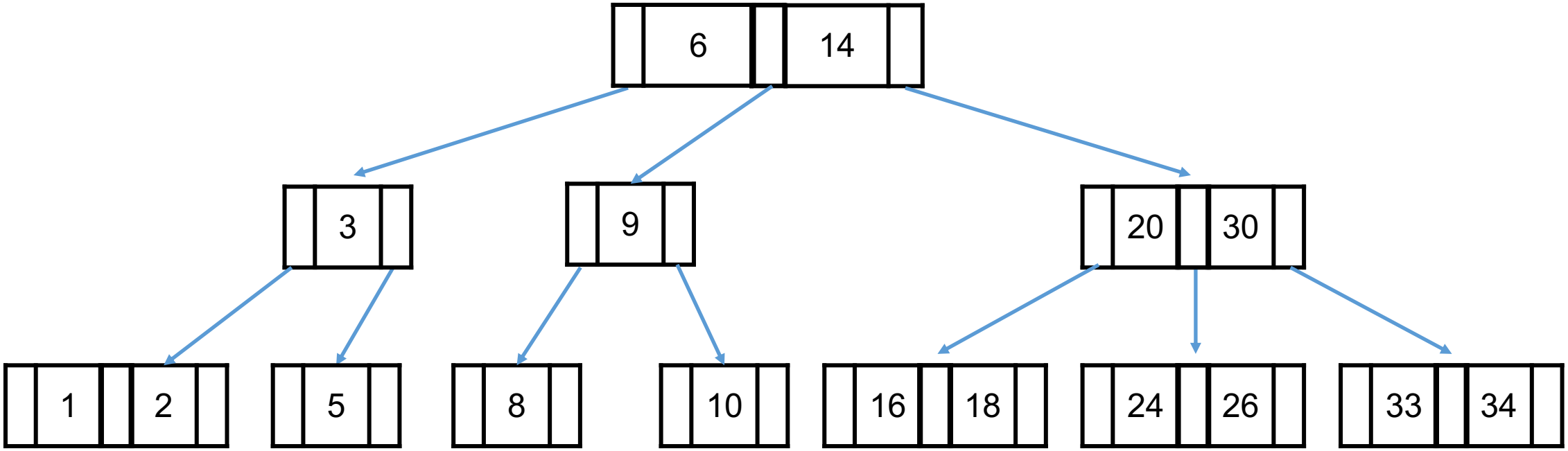
3. 元素删除

4. 删除调整

B 树 – 元素删除(1)

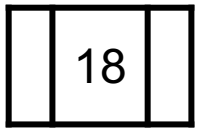
m 阶 B 树的元素删除:

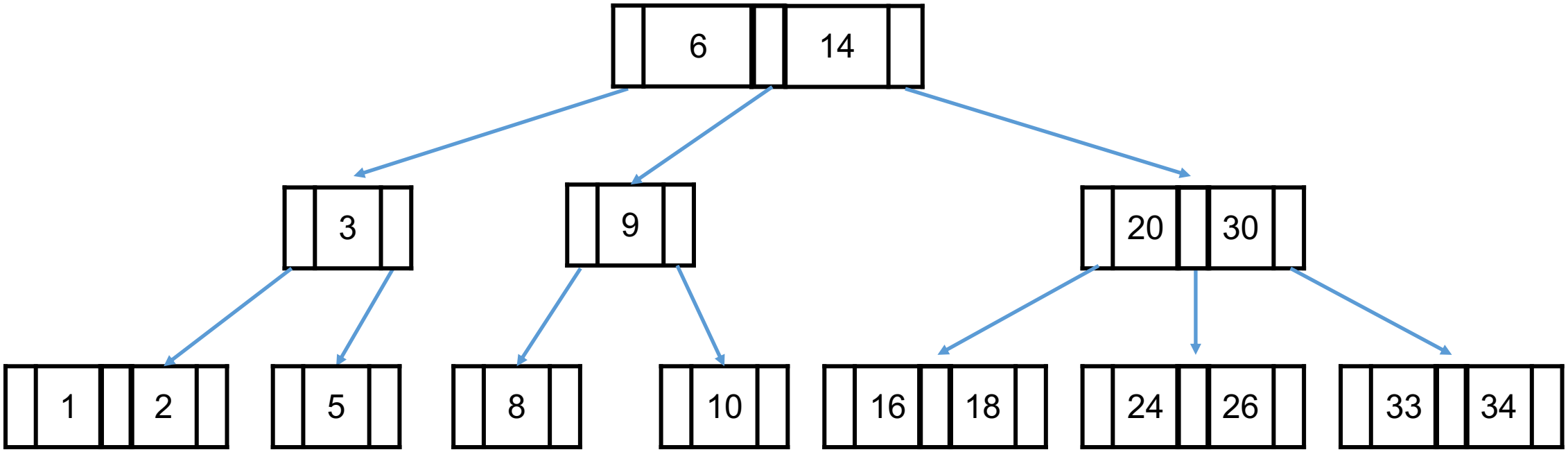
1. 终端节点直接删除
2. 非终端节点，与(前驱/后继)交换，再删除



B 树 – 元素删除(1)

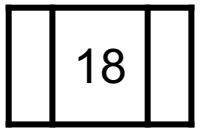
删除：

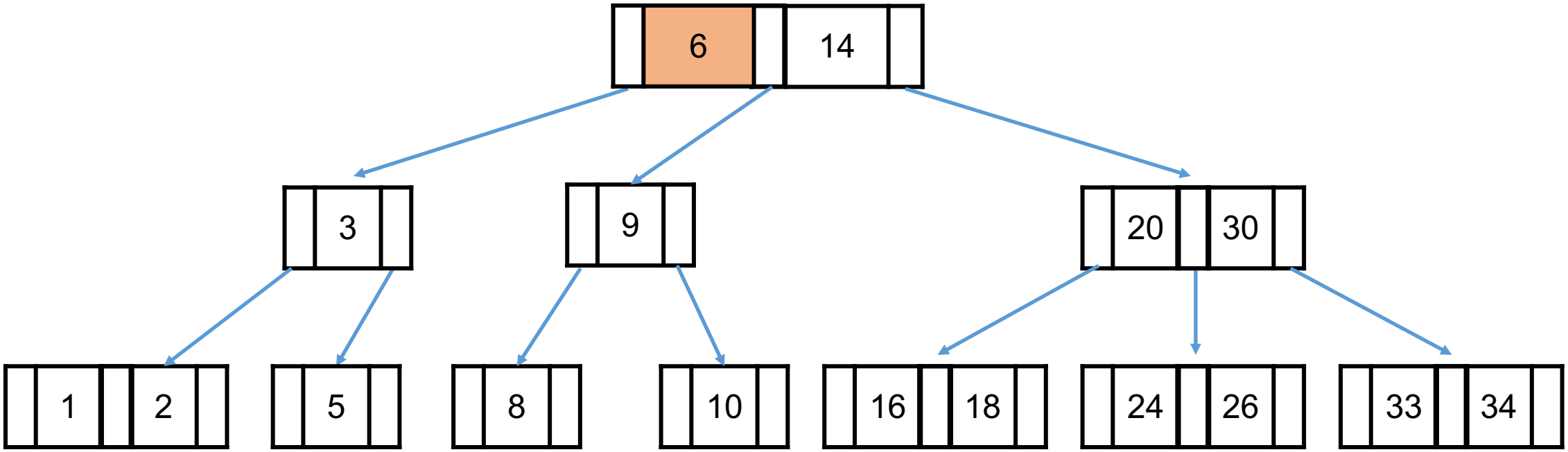




B 树 – 元素删除(1)

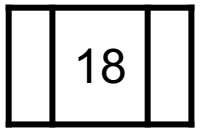
删除：

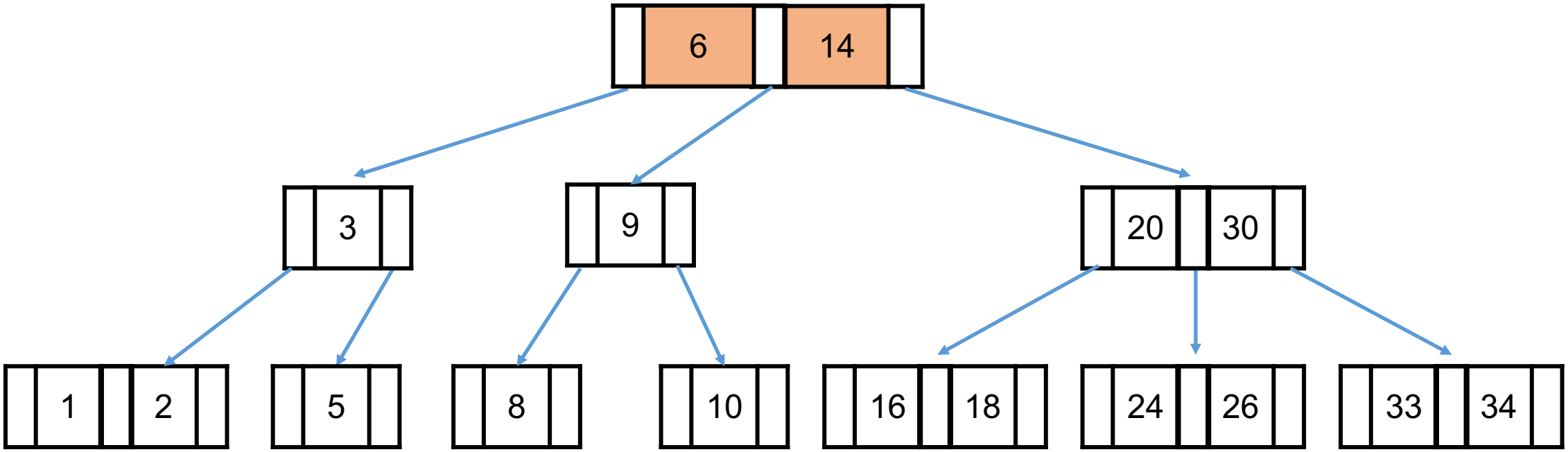




B 树 – 元素删除(1)

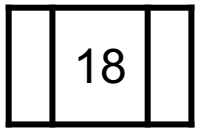
删除：

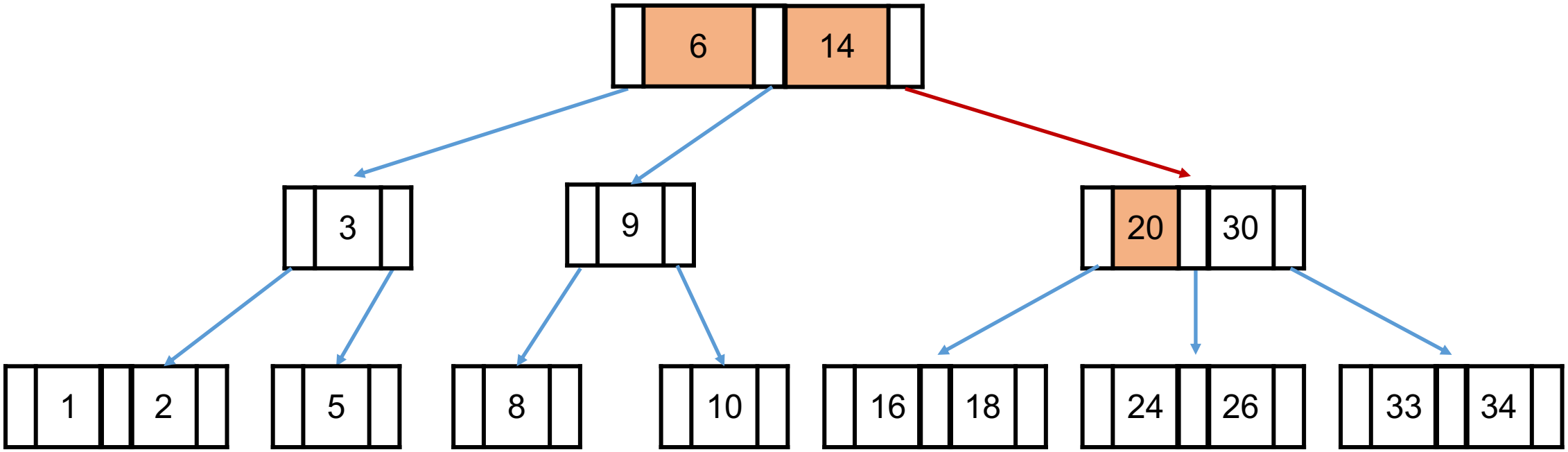




B 树 – 元素删除(1)

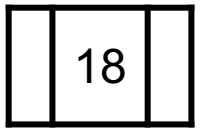
删除：

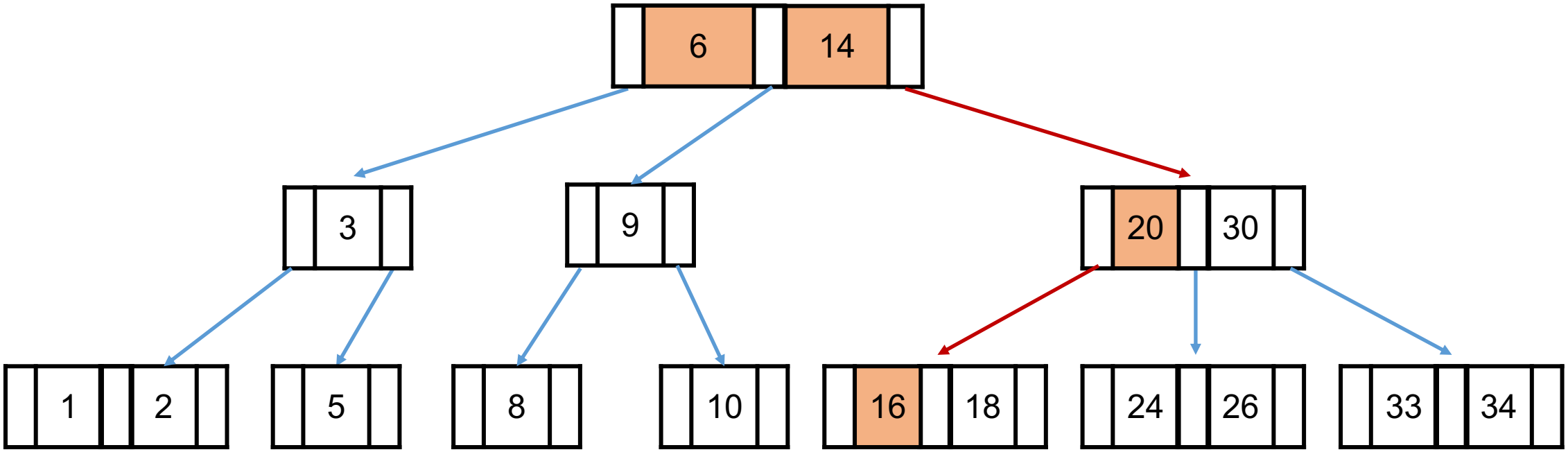




B 树 – 元素删除(1)

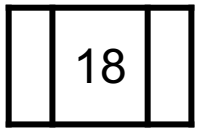
删除：

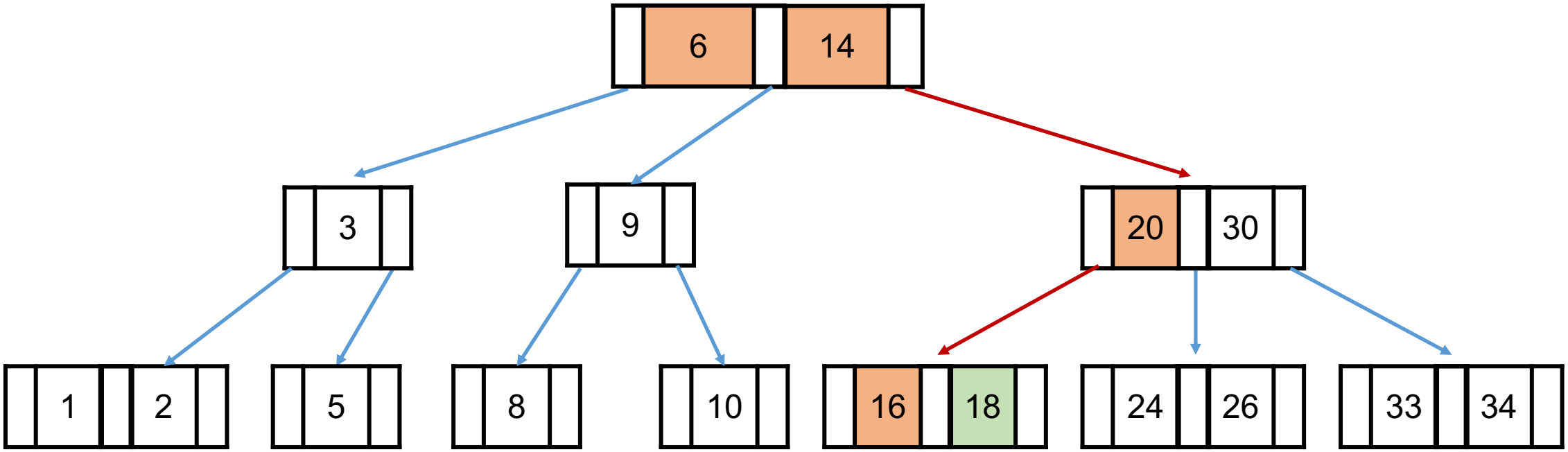




B 树 – 元素删除(1)

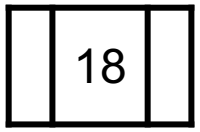
删除：

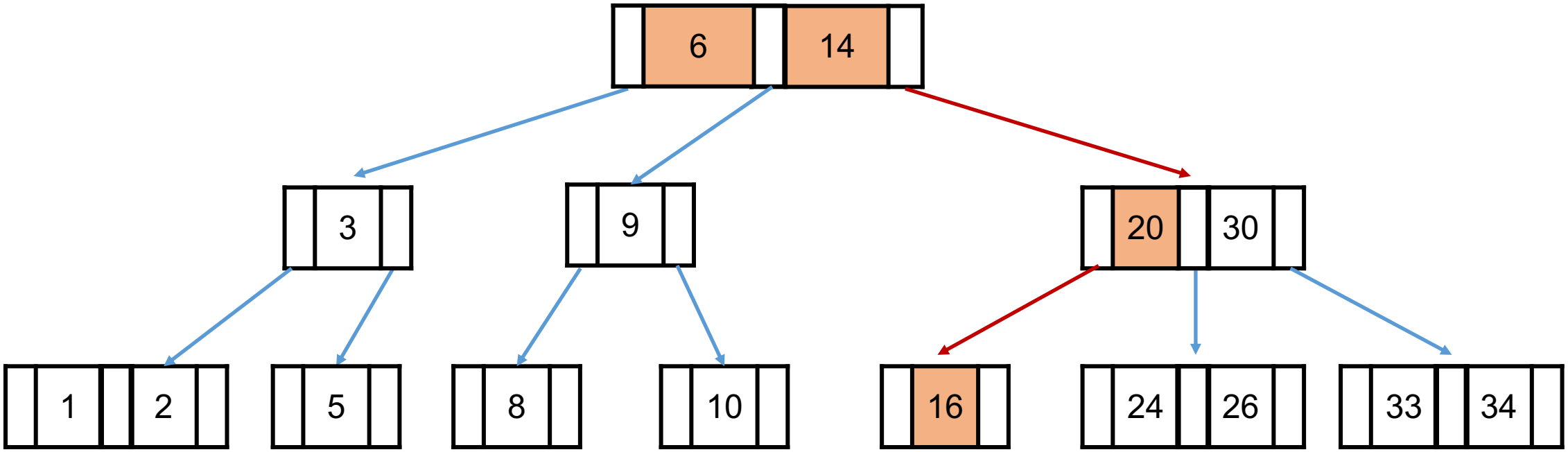




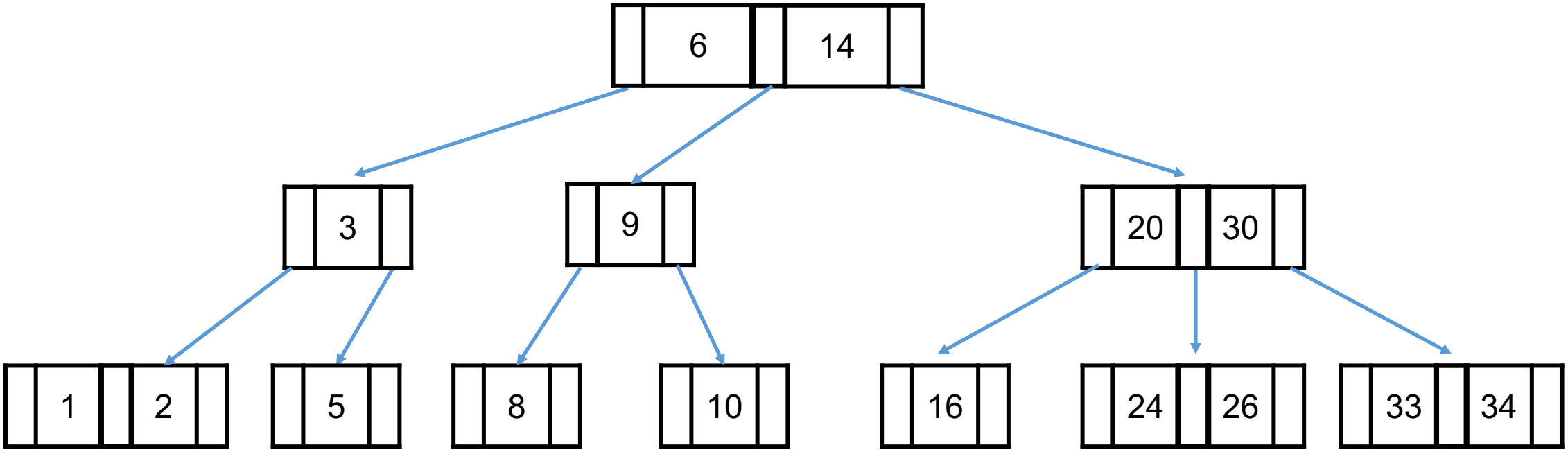
B 树 – 元素删除(1)

删除：



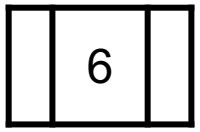


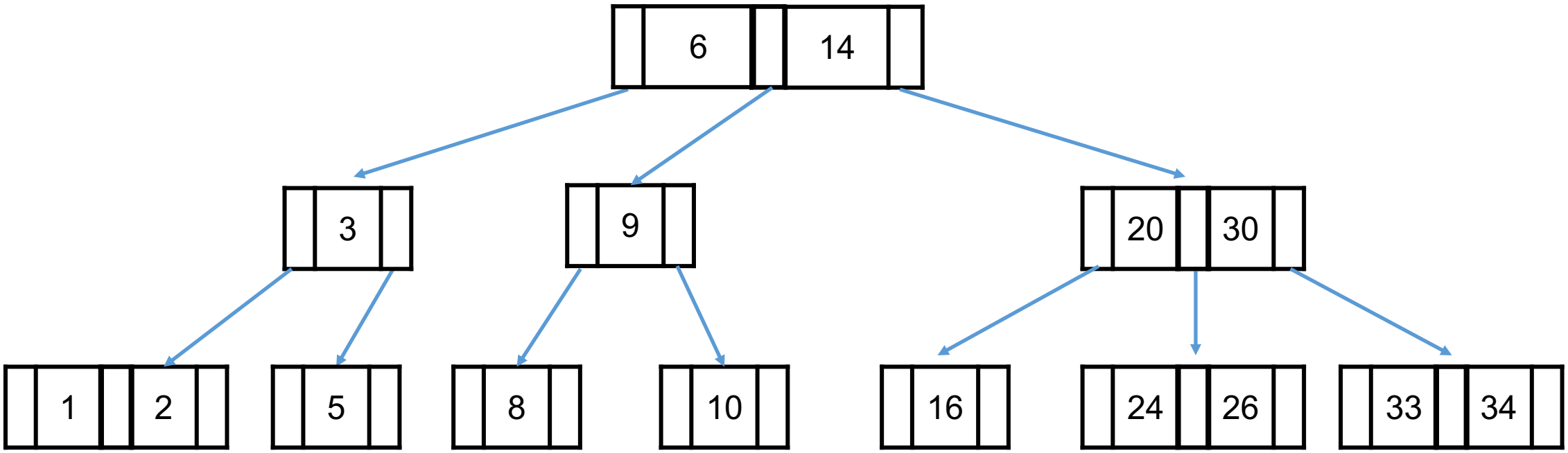
B 树 – 元素删除(1)



B 树 – 元素删除(2)

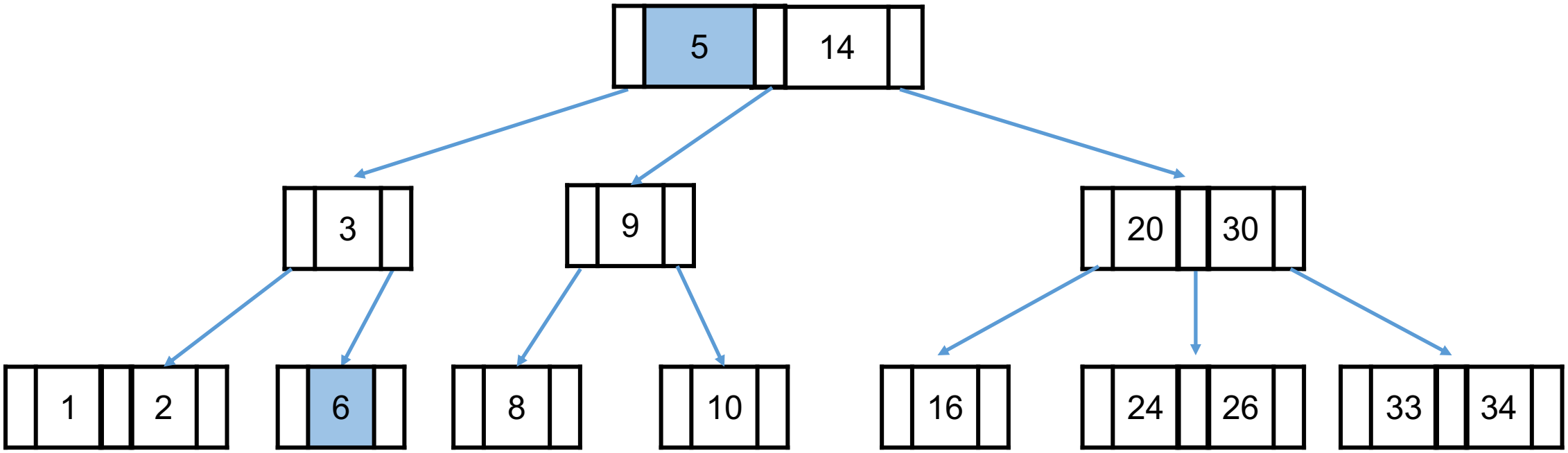
删除：





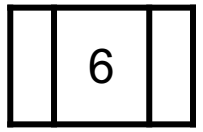
B 树 – 元素删除(2)

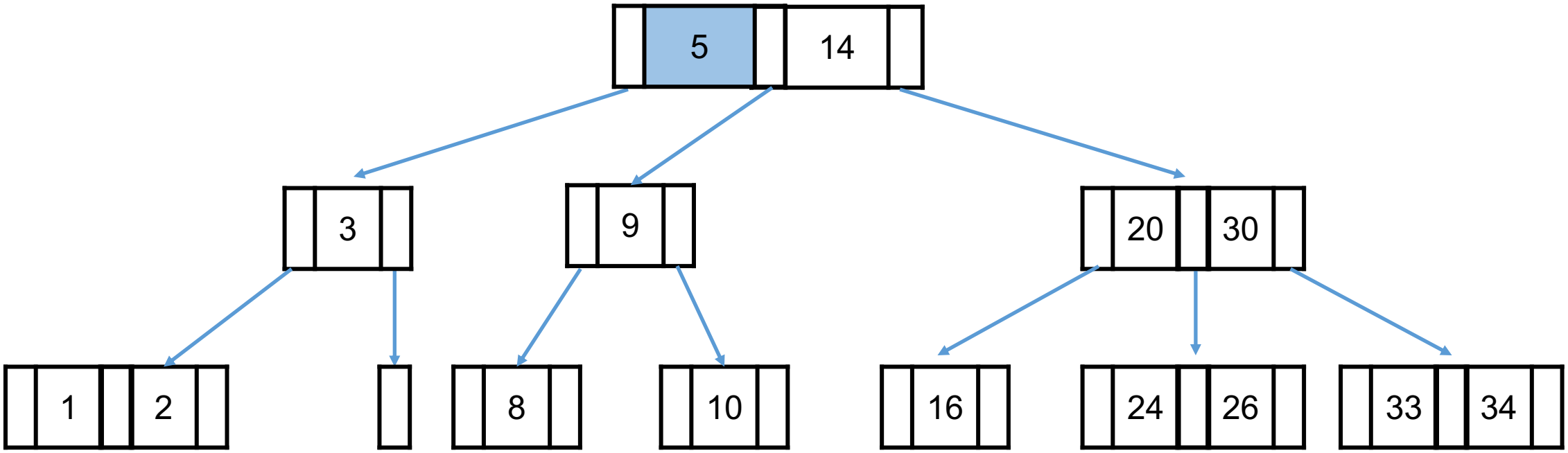
交换 6 和 5



B 树 – 元素删除(2)

删除：





~~1. 元素插入~~

~~2. 插入调整~~

~~3. 元素删除~~

4. 删除调整

~~1. 元素插入~~

~~2. 插入调整~~

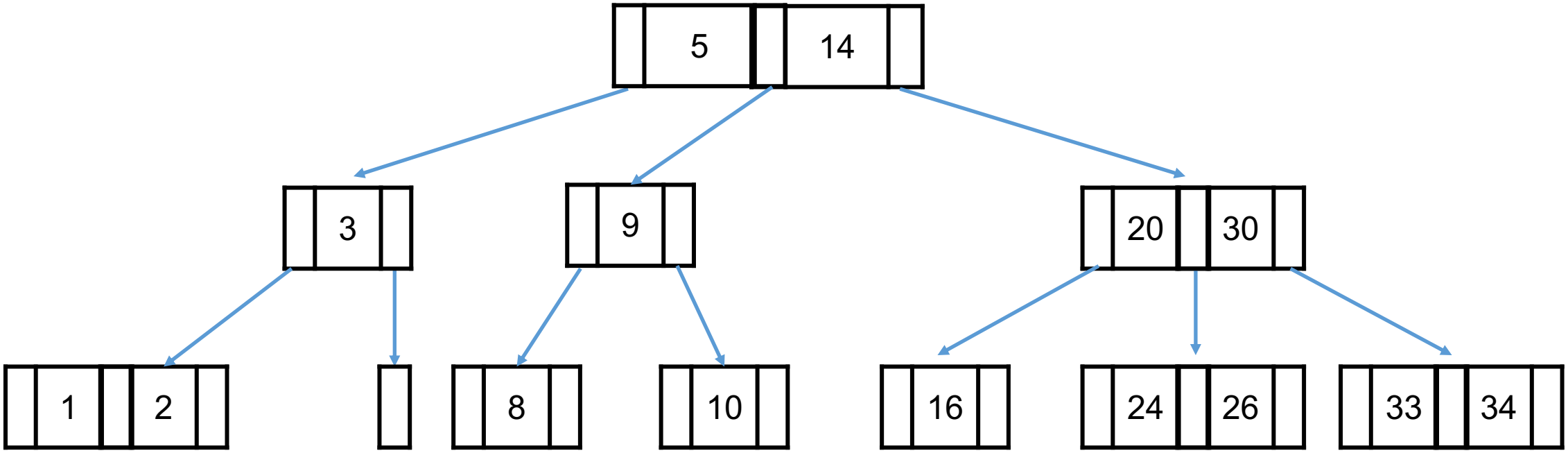
~~3. 元素删除~~

4. 删除调整

B 树 – 删除调整（下溢）

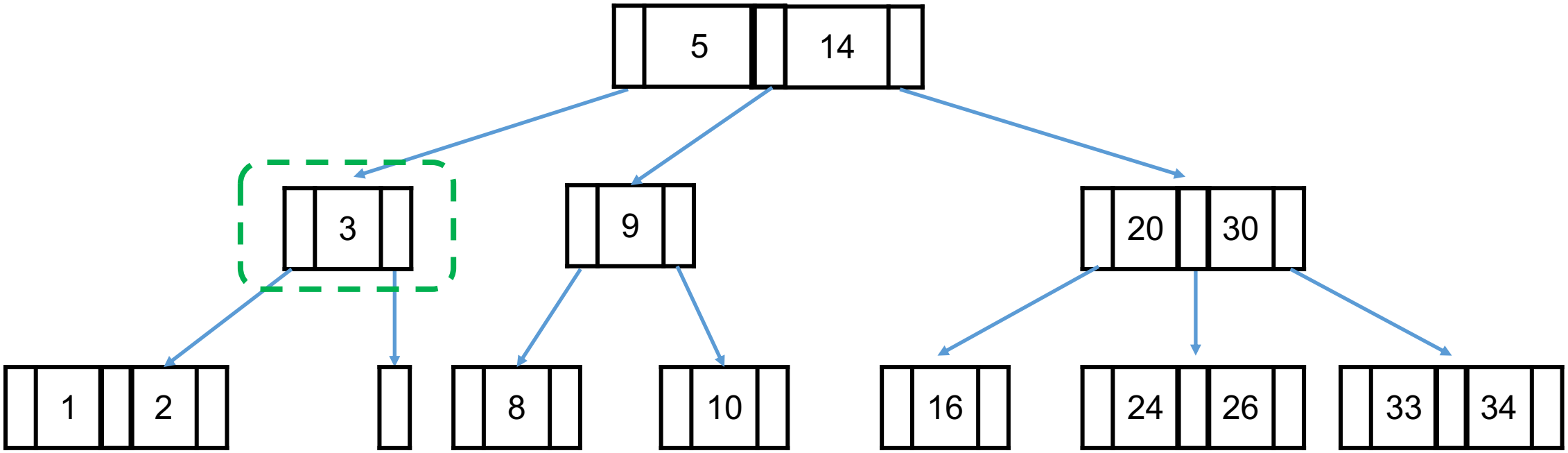
m 阶 B 树的删除调整:

1. 删除调整站在父节点处理，发生在节点关键字数量为 [m/2]-2 时
2. 删除调整的核心操作是：左旋，右旋与合并



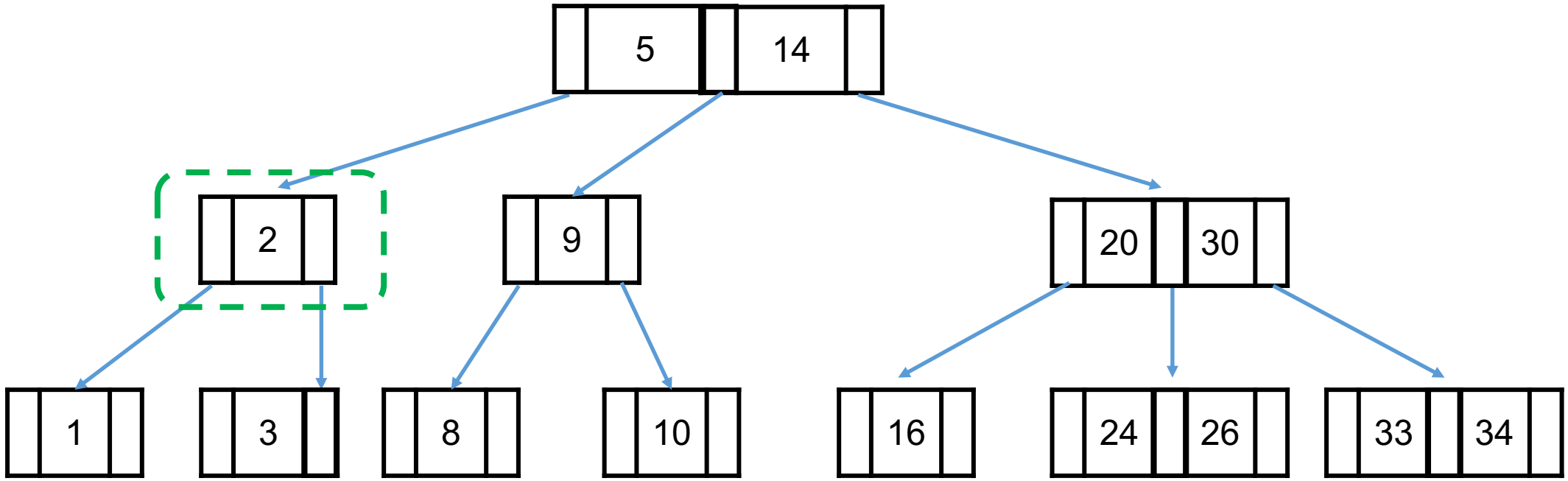
B 树 – 删除调整（下溢）

站在父节点处理

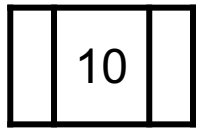


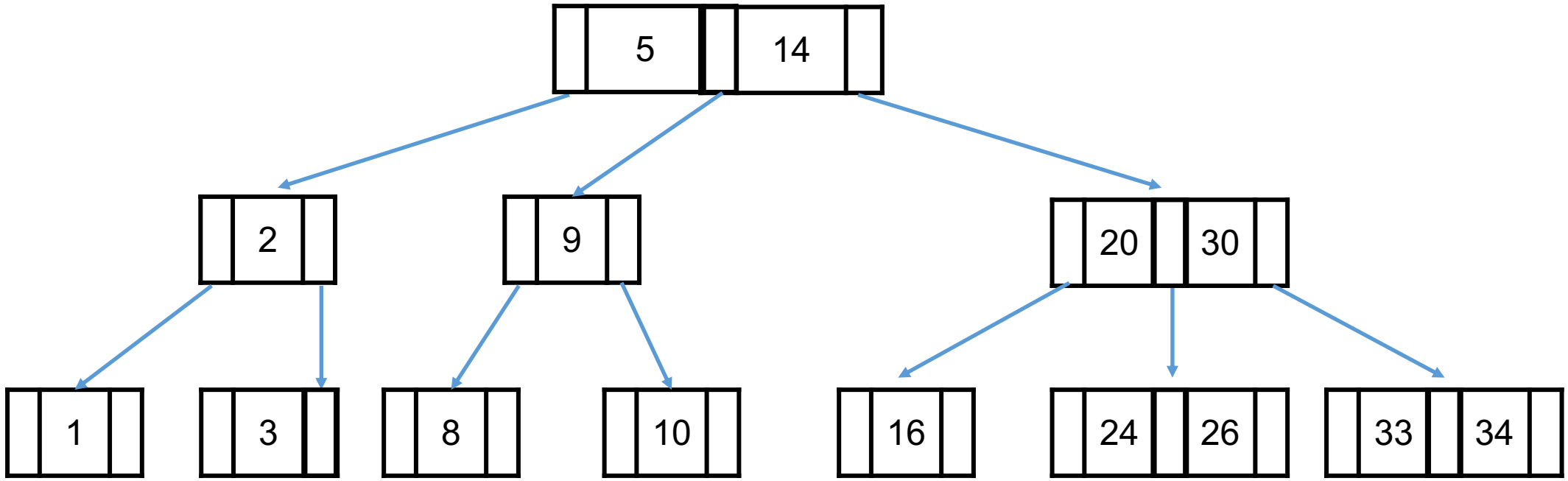
B 树 – 删除调整（下溢）

右旋处理

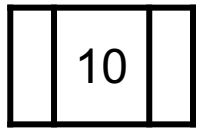


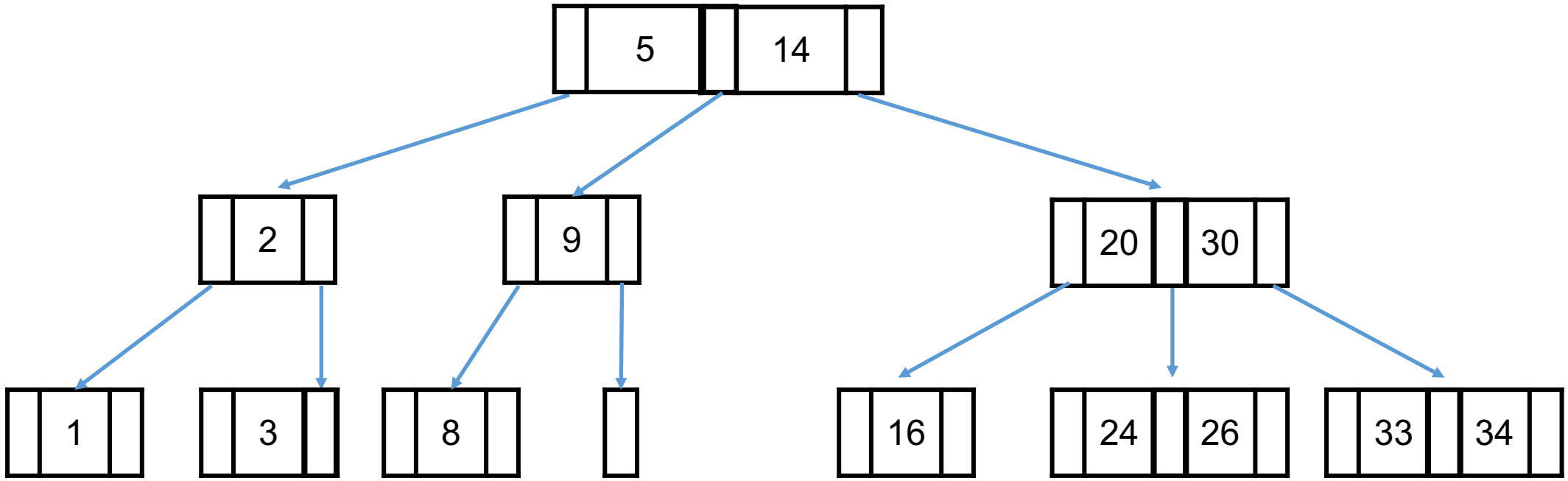
B 树 – 删除调整（下溢）

删除： 



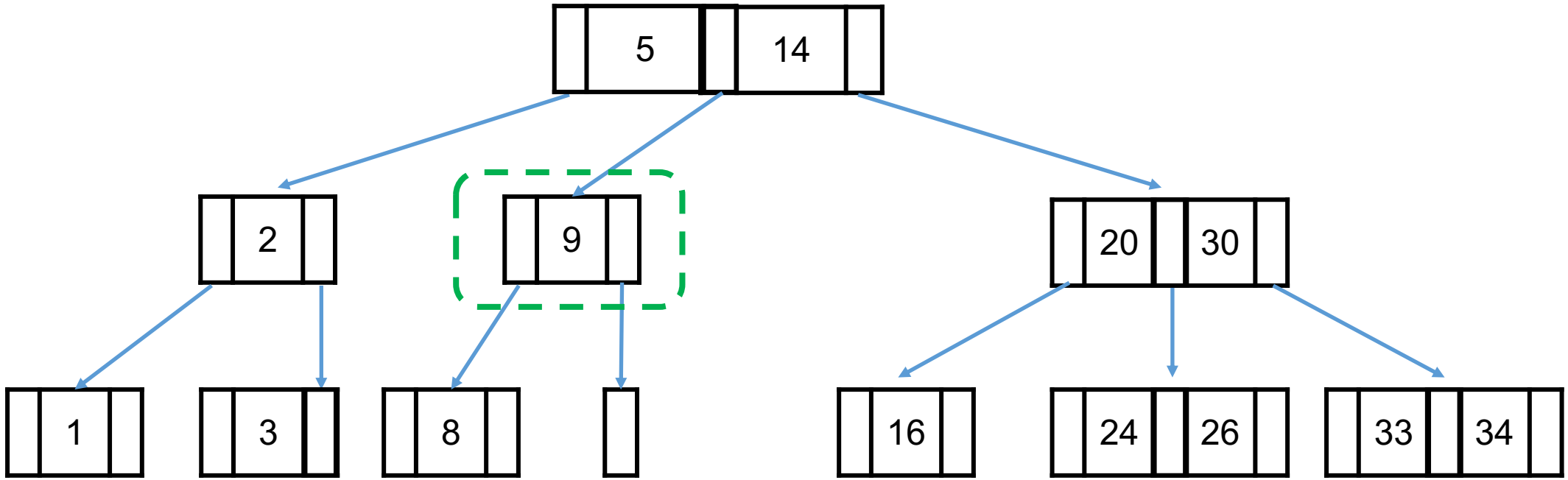
B 树 – 删除调整（下溢）

删除： 



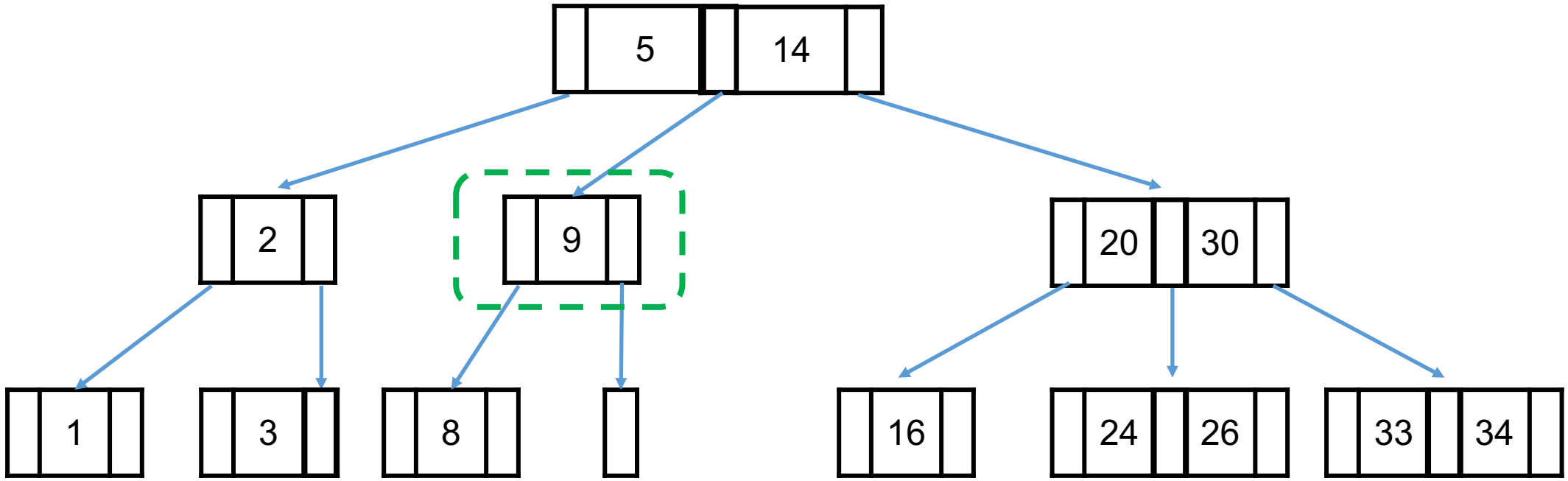
B 树 – 删除调整（下溢）

站在父节点处理



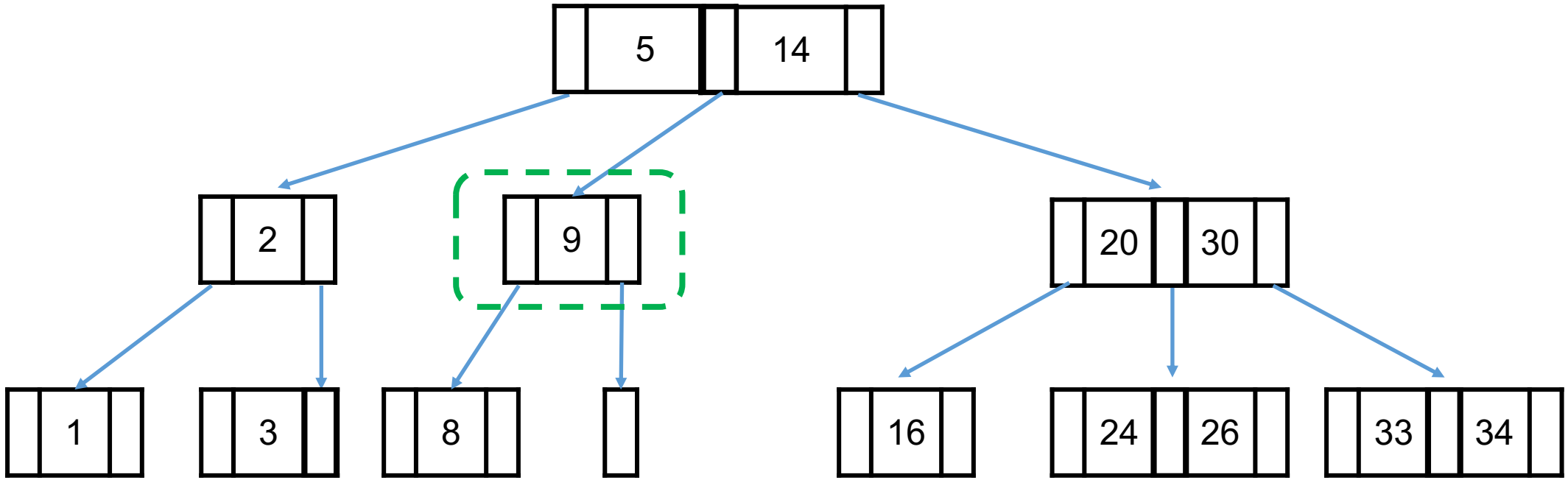
B 树 – 删除调整（下溢）

无法左旋和右旋



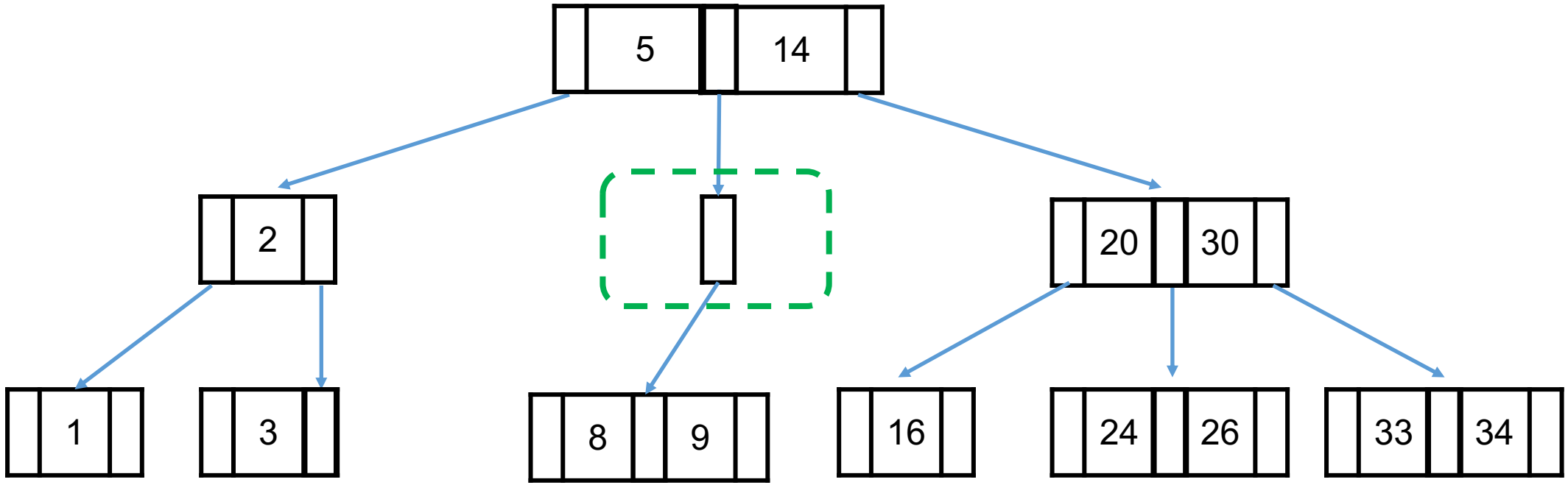
B 树 – 删除调整（下溢）

合并



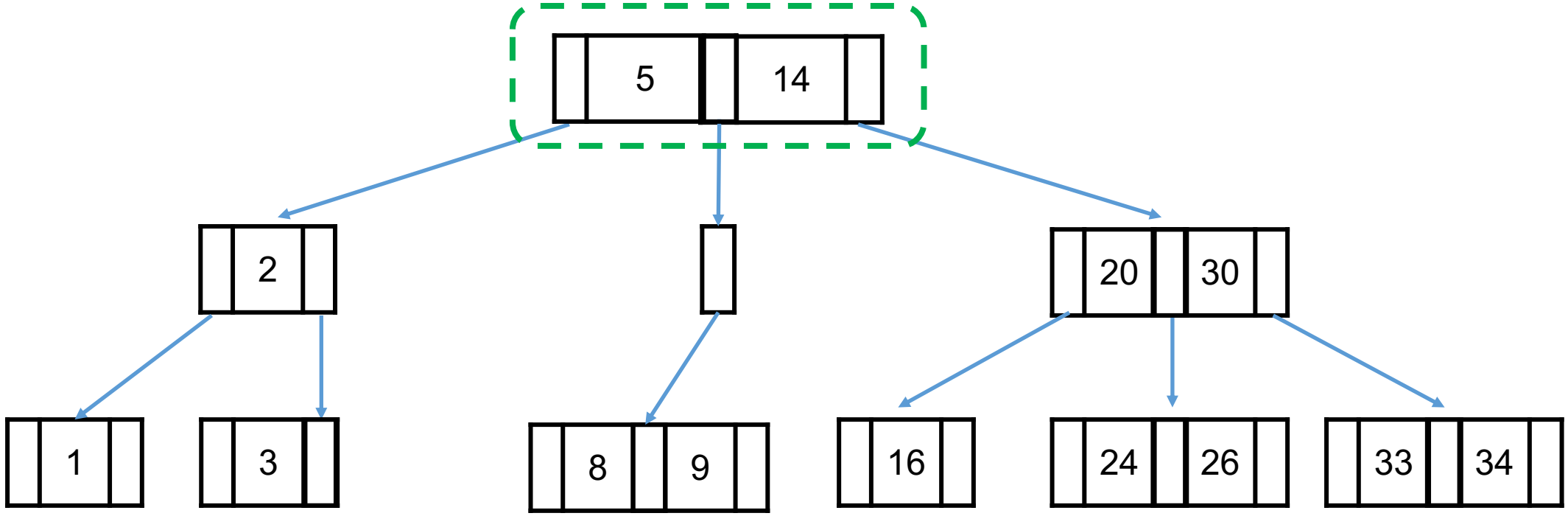
B 树 – 删除调整（下溢）

合并



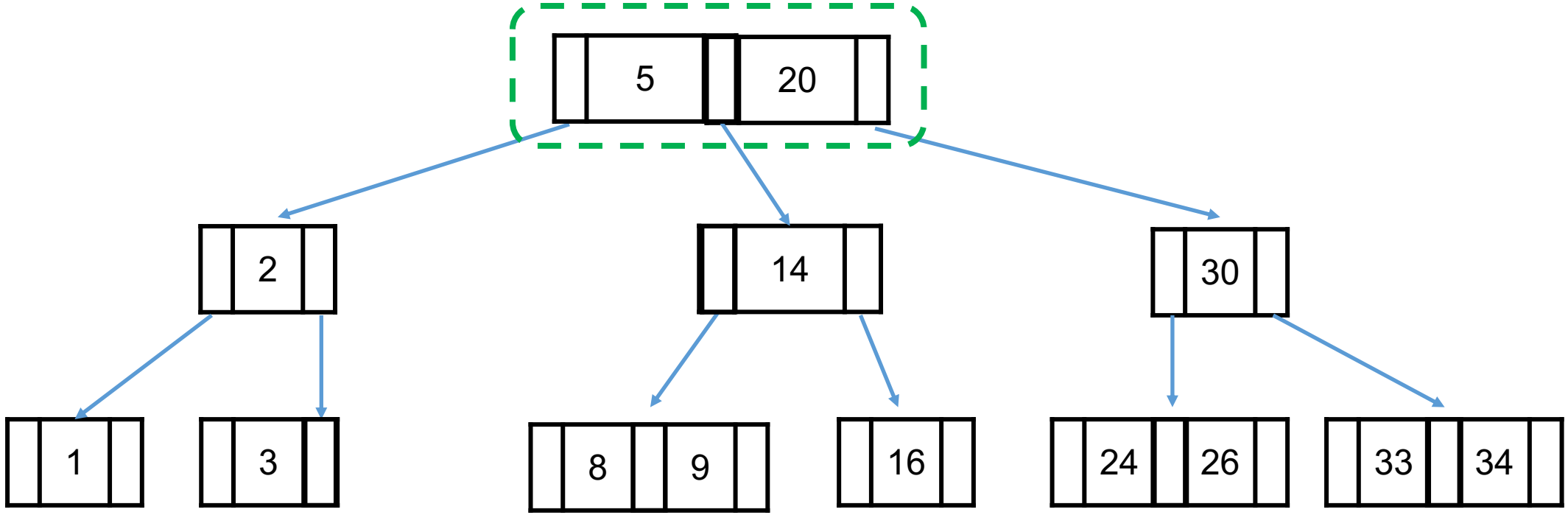
B 树 – 删除调整（下溢）

站在父节点处理



B 树 – 删除调整（下溢）

左旋



1. 元素插入

2. 插入调整

3. 元素删除

4. 删除调整

B 树 – 总结

m 阶 B 树，定义与操作总结：

1. 树中每个节点，最多含有 m 棵子树
2. 根节点中关键字数量范围 $1 \le n \le m-1$
3. 非根结点中关键字数量范围 $\lceil m/2 \rceil - 1 \le n \le m-1$
4. 插入调整为了解决『上溢』节点，关键字数量为 m 的节点
5. 删除调整为了解决『下溢』节点，关键字数量为 $\lceil m/2 \rceil - 2$ 的节点
6. 插入调整的核心操作是：节点分裂
7. 删除调整的核心操作是：左旋，右旋与合并