哈夫曼算法的证明及应用

两个引理

引理1:设C是字符集，$\forall c \in C$, $f(c)$为频率，$x, y \in C$, $f(x)$, $f(y)$频率最小，那么存在最优二元前缀码使得 x, y 码字等长，且仅在最后一位不同.

引理2 设 T 是二元前缀码所对应的二叉树，$\forall\, x, y \in T$, x, y是树叶兄弟，z 是 x, y 的父亲，令$T' = T - \{x, y\}$, 且令 z 的频率$f(z)=f(x)+f(y)$，T'是对应于二元前缀码 $C' = (C - \{x, y\}) \cup \{z\}$的二叉树，那么 $B(T)=B(T')+f(x)+f(y)$.

算法正确性证明思路

定理　**Huffman** 算法对任意规模为n（$n\geq 2$）的字符集C 都得到关于C 的最优前缀码的二叉树.

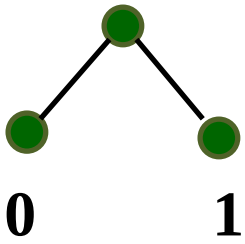
归纳基础　证明：对于$n=2$的字符集, Huffman算法得到最优前缀码.

归纳步骤　证明：假设Huffman算法对于规模为 k 的字符集都得到最优前缀码，那么对于规模为$k+1$的字符集也得到最优前缀码.

归纳基础

n=2，字符集　C={x_1, x_2}，

对任何代码的字符至少都需要1位二进制数字. Huffman算法得到的代码是 0 和 1，是最优前缀码.



归纳步骤

假设Huffman算法对于规模为 k 的字符集都得到最优前缀码. 考虑规模为 $k+1$的字符集

$$C = \{x_1, x_2, ..., x_{k+1}\},$$

其中 $x_1, x_2 \in C$是频率最小的两个字符.

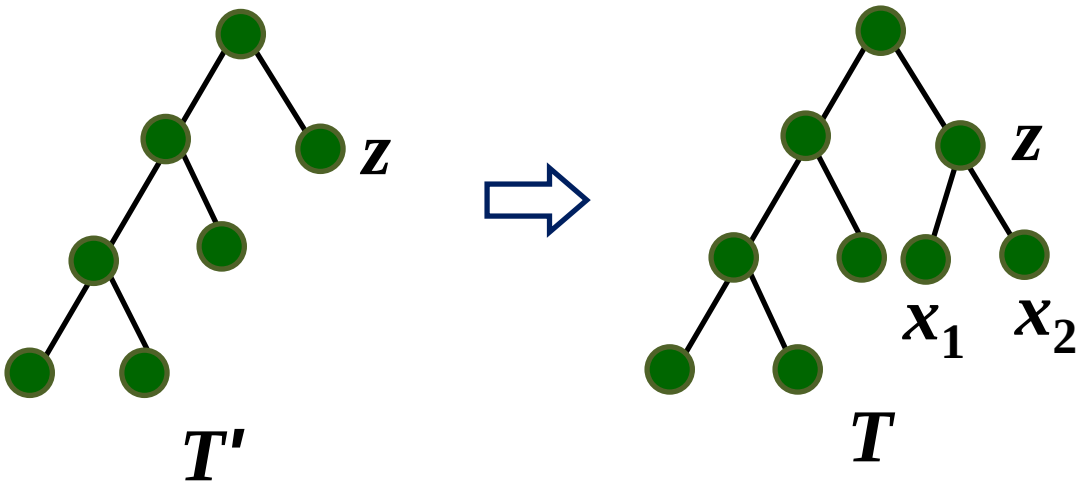
令

$$C' = (C - \{x_1, x_2\}) \cup \{z\},$$
$$f(z) = f(x_1) + f(x_2)$$

根据归纳假设，算法得到一棵关于字符集 C',频率$f(z)$和$f(x_i)$ $(i = 3,4, ...,k+1)$的最优前缀码的二叉树T'.

归纳步骤(续)

把 x_1, x_2作为 z 的儿子附到 T'上，得到树T，那么T是关于$C=(C'-\{z\})\cup\{x_1, x_2\}$的最优前缀码的二叉树.



归纳步骤(续)

如若不然, 存在更优树 T^*, $B(T^*)<B(T)$,
且由引理1，其树叶兄弟是 x_1和 x_2.

去掉 T^*中 x_1和 x_2，得到$T^{*\prime}$. 根据引理2

$$B(T^{*\prime})=B(T^*)-(f(x_1)+f(x_2))$$
$$< B(T)-(f(x_1)+f(x_2))$$
$$= B(T')$$

与 T'是一棵关于 C'的最优前缀码的二叉树矛盾.

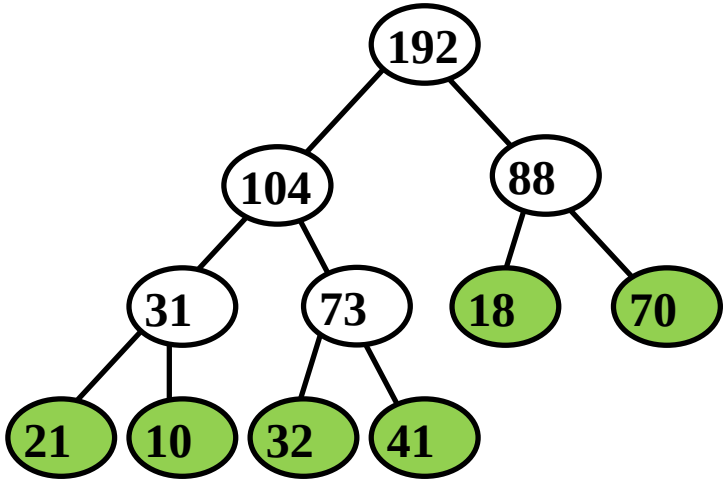
应用:文件归并

问题：给定一组不同长度的排好序文件构成的集合 $S = \{f_1, \ldots, f_n\}$，其中 f_i 表示第 i 个文件含有的项数. 使用二分归并将这些文件归并成一个有序文件.

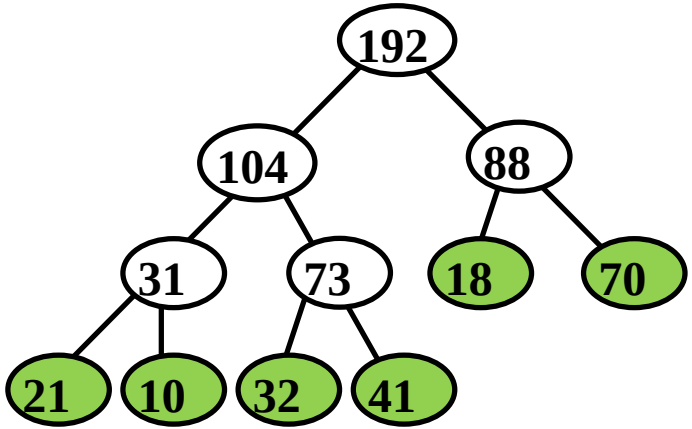
归并过程对应于二叉树：
文件为树叶. f_i 与 f_j 归并的文件是它们的父结点.

两两顺序归并

实例：S = { 21,10,32,41,18,70 }



归并代价



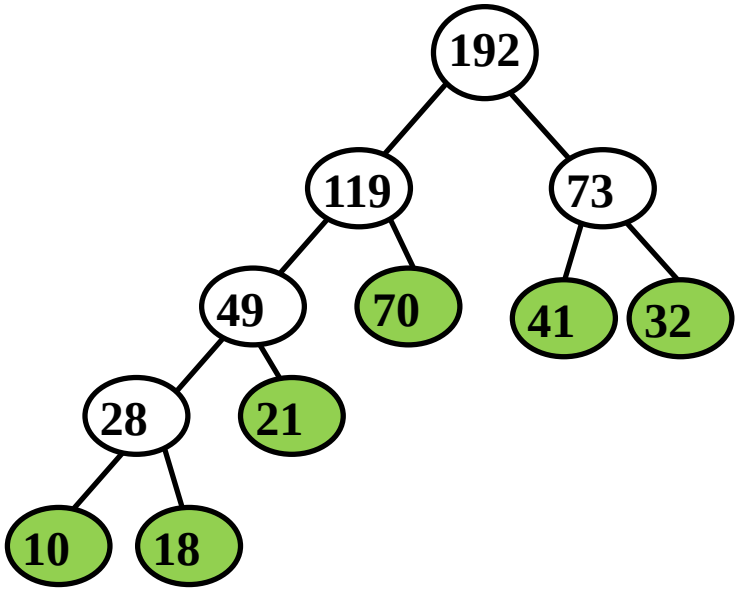
(1) (21+10–1)+(32+41–1)+(18+70–1)+
(31+73–1)+(104+88–1) = 483

(2) (21+10+32+41)×3+(18+70)×2–5=483

代价计算公式 $\sum_{i \in S} d(i) f_i - (n-1)$

实例：Huffman树归并

输入：$S=\{21,10,32,41,18,70\}$



代价: $(10+18)\times4+21\times3+(70+41+32)\times2-5=456$

小结

- 哈夫曼算法的正确性证明：
  对规模归纳
- 哈夫曼算法的应用：
  文件归并