**哈夫曼编码算法说明书**

**1. 算法功能**

构造字符的最优前缀码，用于数据文件压缩。

**2. 接口参数**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 名称 | 类型 | 是否必须 | 示例值 | 描述 |
| 1 | chars | char [] | 是 | [“a”, “b”, “c”] | 字符数组，需编码的字符 |
| 2 | freqs | double [] | 是 | [0.5, 0.3, 0.2] | 字符对应的频率，与chars中字符一一对应 |

1. **接口返回值**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 序号 | 名称 | 类型 | 是否必须 | 示例值 | 描述 |
| 1 | tree | Node | 是 |  | 构造好的哈夫曼树，遍历一次即可得到最优前缀码 |

1. **算法实现**
2. 初始化一个优先队列，通常用堆实现，队列元素的数据类型为哈夫曼树结 　点。
3. 为每个字符初始化一个哈夫曼树结点，即叶子节点。
4. 当队列元素数大于一进行如下的贪心选择：从队列中选择两个频率最小的 　元素，构造一棵子树，父节点的频率为此２个结点的频率之和。构造好的 　子树加入优先队列。
5. 最终队列中唯一的元素为构造好的哈夫曼树。

**5. 注意事项**

无