

1.10 哈夫曼编解码_物联网 / 嵌入式工程师 - 慕课网

“ 慕课网慕课教程 1.10 哈夫曼编解码涵盖海量编程基础技术教程，以图文图表的形式，把晦涩难懂的编程专业用语，以通俗易懂的方式呈现给用户。

10. 哈夫曼编解码

我们小学，初中，高中都是以分数来评价一个人的成绩的。分数代表了一切。

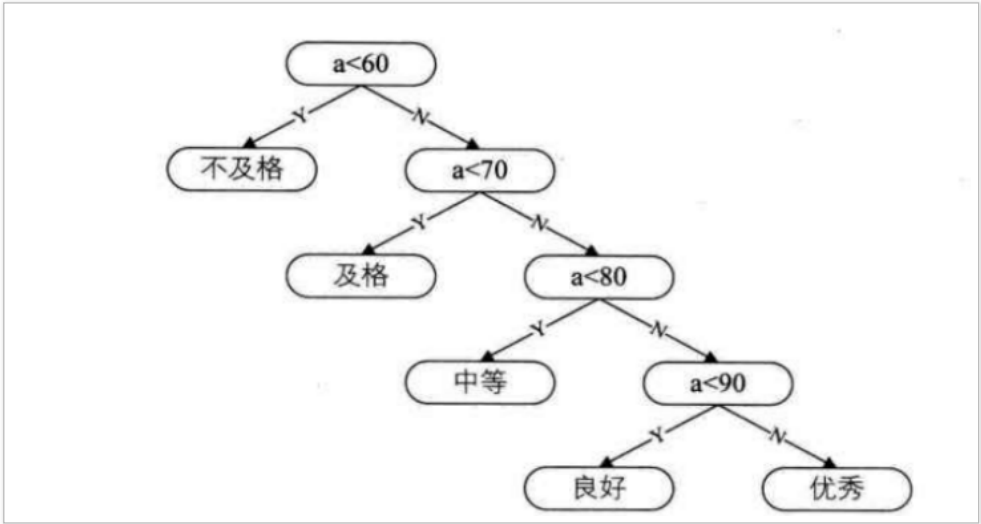
国家提倡了素质教育。我们把成绩模糊化，以优秀，良好，中等，及格，

不及格这样的词语来模糊我们的分数。我们写代码的时候，可以把分数转换位代码形式。

(默认成绩合法 [0-100])

```
if(a < 60)
    b = "不及格";
else if(a < 70)
    b = "及格"
else if(a < 80)
    b = "中等"
else if(a < 90)
    b = "良好"
else
    b = "优秀"
```

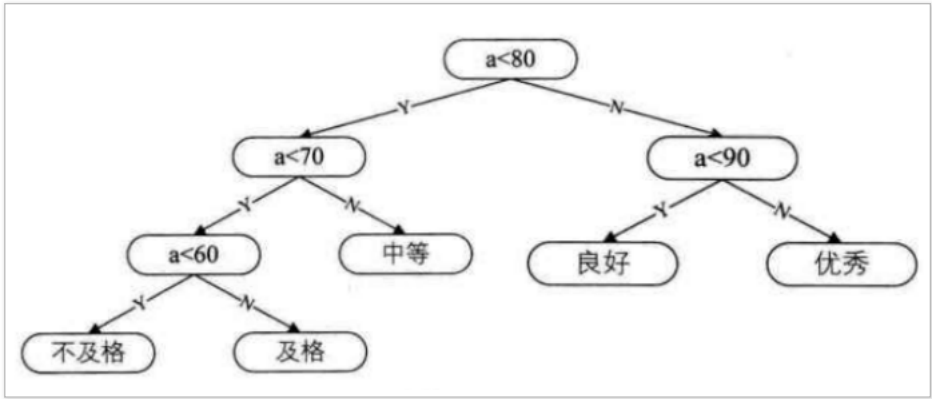
OK, 上面成绩粗率的看是没有什么太大的问题的！若是我们把它转化成我们二叉树的形式。



我们把班级的学生一般开始成绩做了一个百分比，我们来详细的看看。

分数	0 ~ 59	60 ~ 69	70 ~ 79	80 ~ 89	90 ~ 100
所占比例	5%	15%	40%	30%	10%

我们发现 70~79 这个阶段的人数最多。若是按照，我们上图画的代码的形式判断的话，我们要经过 3 个 if 语句的判断，才能得到我们的结果。这样明显效率太低。我们可以重新分配二叉树。



这个时候，我们访问 [70–80] 之间的学生，相对来说，效率就应该高不少了。那么上图是如何来的呢？

哈夫曼树：它又叫最优二叉树，指的是对于一组具有确定权值的叶子结点的具有最小带权路径长度的二叉树。

路 径 (Path)****: 从树中的一个结点到另一个结点之间的分支构成两个结点间的路径。

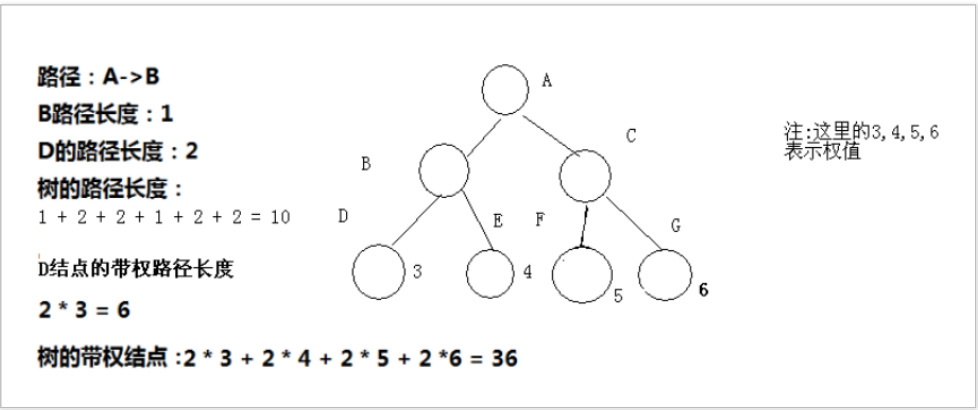
路径长度 (Path Length)****: 路径上的分支数。

树的路径长度 (Path Length of Tree)****: 从树的根结点到每个结点的路径长度之和。在结点数目相同的二叉树中，完全二叉树的路径长度最短。

结点的权 (Weight of Node)****: 在一些应用中，赋予树中结点的一个有实际意义的数。

结点的带权路径长度 (Weight Path Length of Node)****: 从该结点到树的根结点的路径长度与该结点的权的乘积。

树的带权路径长度 (WPL)****: 树中所有叶子结点的带权路径长度之和。



哈夫曼编码，就是可以用来对我们的数据进行编码，以提高压缩效率。

问题：假设 `char a[] = "ABABCAA"`，思考上述字符串应该定义多少字节存储？

答案： 8 bytes

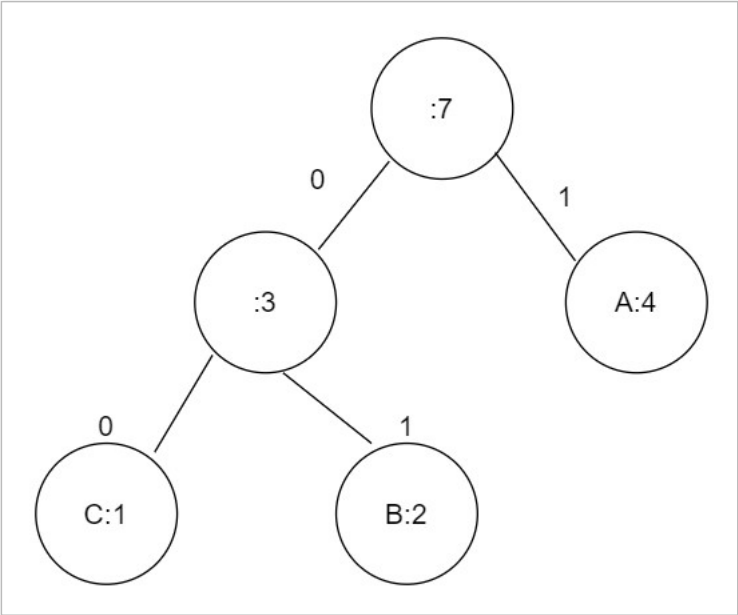
1. 统计相同字符出现的次数。字符当作数据存储，次数当作权值

A : 4

B :2

C :1

1. 把结点按照权值 ** 两两合并, 小的放左边, 大的放右边,** 形成 huffman 树。



A : 1

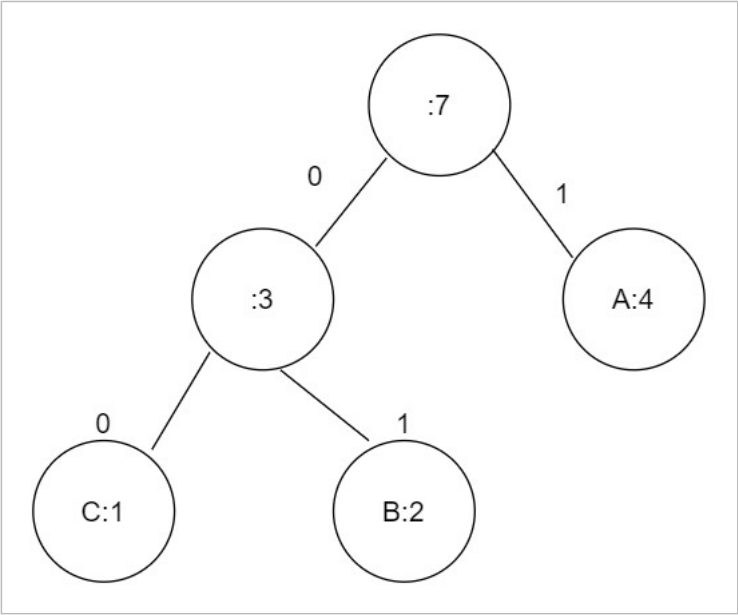
C : 00

B : 01

原始数据: ABABCAA

编码数据: 1 01 1 01 00 11

注： 2byte 即可存储编码后的数据



密码： 1011010011

解码方法： 从根结点开始访问，根据 huffman 树来依次访问，有字符数据的肯定是我们的叶子节点
**。 **

第一次， 1==> 肯定是右子树 A，然后访问到根节点，然后剩下 011010011

第二次， 0 然后有子树，在向下走， 01，肯定是 B，访问到根结点。然后剩下 1010011

第三次， 1==》肯定是右子树 A，然后访问到根节点，然后剩下 010011

第四次，0 然后有子树，在向下走，01，肯定是 B，访问到根结点。然后剩下 0011

第五次，0 然后有字数，向下走，00，肯定是 C，访问到根结点。然后剩下 11

第六次，1，然后肯定是 A。

第七次，1，然后肯定是 A。

解码后的数据位”ABABCAA”。

有个字符串为 `char b[] = “ADDCCDCBCDDBDDD”` 用上述方法，对此字符串进行编码和解码。

全文完

本文由 简悦 SimpRead 优化，用以提升阅读体验

使用了 全新的简悦词法分析引擎 beta，点击查看详细说明

