# 实验一 对指令操作码进行霍夫曼编码

1. **实验目的**

1． 了解和掌握指令编码的基本要求和基本原理

1. **实验内容**
2. 使用编程工具编写一个程序，对一组指令进行霍夫曼编码，并输出最后的编码结果以及对指令码的长度进行评价。与扩展操作码和等长编码进行比较。

有一组指令的操作码共分七类，它们出现概率如：下表所示：

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| P1 | P2 | P3 | P4 | P5 | P6 | P7 |
| 0.45 | 0.30 | 0.15 | 0.05 | 0.03 | 0.01 | 0.01 |

对此组指令进行 HUFFMAN 编码正如下图所示：

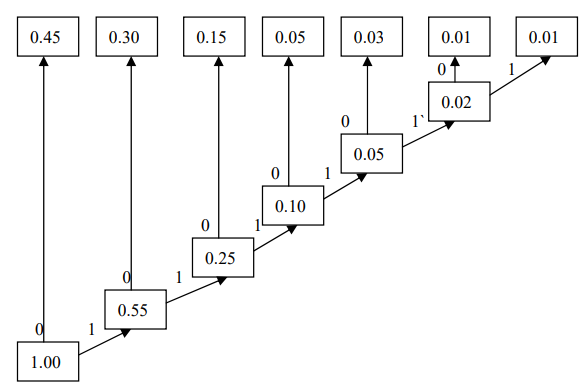


图1

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| P1 | P2 | P3 | P4 | P5 | P6 | P7 |
| 0 | 10 | 110 | 1110 | 11110 | 111110 | 111111 |
| 1 | 2 | 3 | 4 | 5 | 6 | 6 |

最短编码长度为： H=0.45\*1+0.30\*2+0.15\*3+0.05\*4+0.03\*5+0.01\*6+0.01\*6=-1.95. 要对指令的操作码进行 HUFFMAN 编码，只要根据指令的各类操作码的出现概率构造 HUFFMAN 树再进行 HUFFAM 编码。此过程的难点构造 HUFFMAN 树，进行 HUFFAM 编码只要对你所生成的 HUFFMAN 树进行中序遍历即可完成编码工作。

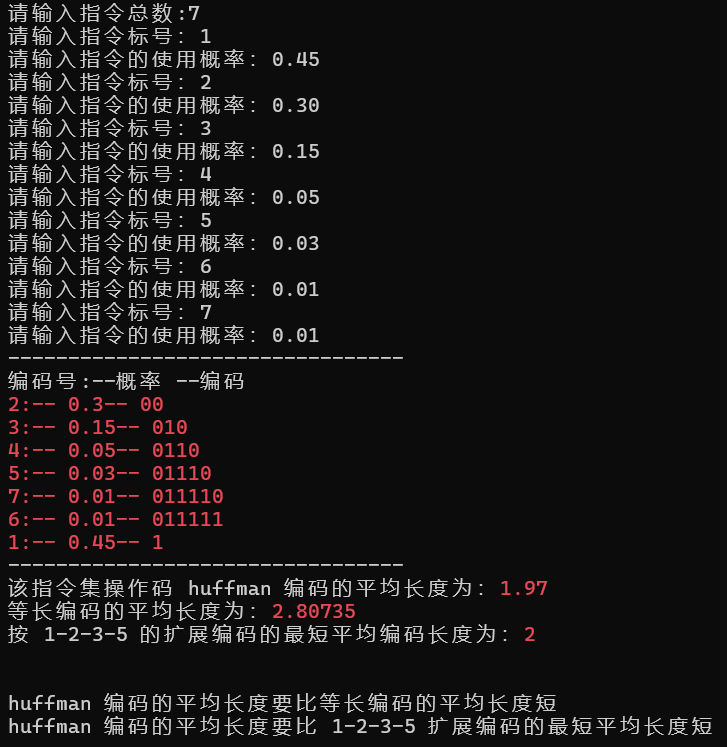
1. **实例**

观察上图 1，不难看出构造 HUFFMAN 树所要做的工作：

1、先对各指令操作码的出现概率进行排序，构造一个有序链表。

2、再取出两个最小的概率节点相加，生成一个生的节点加入到链表中，同时从两表中删除此两个节点。

3、在对链表进行排序，链表是否只有一个节点，是则 HUFFAN 树构造完毕，否则继续做 2 的操作。为此设计一个工作链表（链表的元素时类，此类的功能相当结构。）、HUFFMAN 树节点、HUFFMAN 编码表节点。

****