

电子科技大学信息与通信工程学院

标准实验报告

(实验) 课程名称 信息论与编码

电子科技大学教务处制表

电子科技大学

实验报告

学生姓名：李聪 学 号：2019010398114 指导教师：庞宏

一、实验室名称： 自备计算机

二、实验项目名称： 霍夫曼编码实验

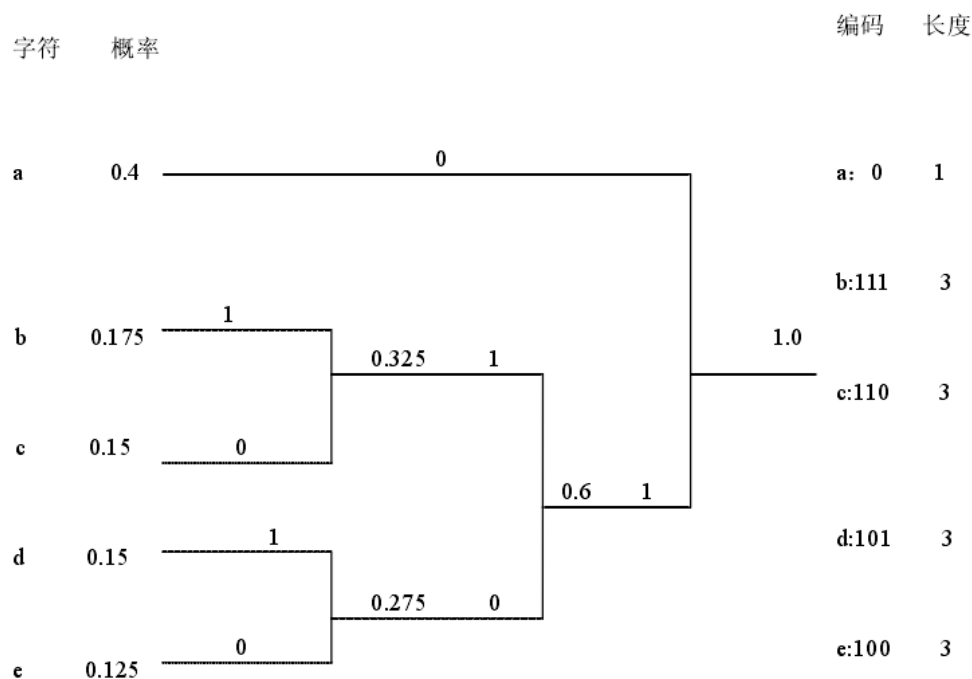
三、实验原理：

Huffman 编码是 D. Huffman 于 1952 年提出的一种基于统计模型的无损压缩编码方法。其主导思想是根据数据符号发生的概率进行编码，对小概率的输入符号使用长码字表示，而对于大概率输入符号用短字码表示，从而达到较短的平均编码。理论研究表明这是一种接近压缩比上限的好的编码方法，该方法自提出后就成了数据压缩中的一个研究主题并被广泛的应用于文本、图像及视频的压缩，如 JPEG、MPEG 标准中。

Huffman 编码原理主要是利用 Huffman 二叉树，构造方法是从树叶到树根生成二叉树，其步骤如下：

- (1) 将压缩的各字符的出现概率按减少的顺序进行排列；
- (2) 将两个最小的概率进行组合相加，得到一个新概率；将这一新概率与其它概率一起，继续执行 (1) 和 (2) 的操作，直至最后概率为 1.0；
- (3) 对每对组合中的概率较大者指定为 1,较小者指定为 0；
- (4) 画出由每个信源符号概率到 1.0 处的路径，记下路径的 1 和 0；
- (5) 对每个信源符号，由从右到左写出 1、0 序列，就得到了对应的 Huffman 码。下面举个简单的例子来看 Huffman 编码的过程： 例如一组需要编码的信息为: cabced eacacdeddaaabaababaaabbacd ebacead a。共出现 a、b、c、d、e 五个符号，按上面的步骤先将它们排好顺序：a (0.4)、b (0.175)、c (0.15) 、d (0.15) 、e (0.125)，括号中是其出现的概率，按上述步骤操作得到编码结

果为：a (0)、b (111)、c (110)、d (101)、e (100) ，整个过程如下图所示：



新的编码长度为 $16 \times 1 + 7 \times 3 + 6 \times 3 + 6 \times 3 + 5 \times 3 = 88\text{bit}$ ，而原码长 $40 \times 8 = 320\text{ bit}$ ，可见采用 Huffman 编码，获得了明显的数据压缩效果。利用公式 (2-1) 可以算出本例的信源熵 $H(x)$ 为 2.048 bit ，而编码后的平均码长 l 为 2.2 bit ，已经十分接近，

可以算出其编码效率为：

$$\eta = \frac{H(x)}{l} = \frac{2.048}{2.2} = 93.1\%$$

从上述过程可以看出，Huffman 树中没有度为 1 的结点，属于正则二叉树，一颗有 n 个叶结点的 Huffman 树共有 $2n - 1$ 个结点。

四、实验目的：

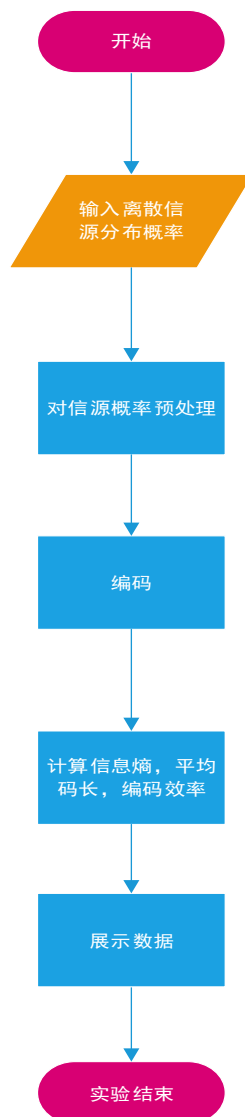
- 1、掌握霍夫曼编码原理；
- 2、练习用 MATLAB 编写霍夫曼编码

五、实验内容：

- 1、 编程实现霍夫曼编码及编码效率的计算。
- 2、 画出程序的流程图并做简要说明。
- 3、 对计算结果进行分析，得出结论。

六、实验器材（设备、元器件）：PC 计算机一台，MATLAB 软件

七、实验步骤：



根据流程图编写 MATLAB 代码。

```

clear all
clc
%户输符号概率
p = input('请输入离散信源概率分布，例如[p1, p2, ... ..pn]: \n');
N = length(p);
p0 = p;
symbols = cell(1,N);
for i = 1:N          % i 表第 i 个符号
    symbols{i} = ['x',num2str(i)];
end
[dict,L_ave] = huffmandict(symbols,p);
dict = dict.';
H = sum(-p.*log2(p));%计算信源信息熵
yita = H/L_ave;%计算编码效率
CODE = strings(1,N);%初始化对应码字
for i = 1:N          % i 表示第 i 个符号
    CODE(i) = num2str(dict{2,i});
end
%输出码字、平均码长和编码效率
fprintf('\n 运算结果:\n');
disp('信源符号: ');disp(dict(1,1:N));
disp(['对应概率: ',num2str(p)]);
disp('对应码字: ');disp(CODE);
disp(['平均码长: ',num2str(L_ave)]);
disp(['编码效率: ',num2str(yita)]);

%%%%%%%%%%%%%%
%%%%%%%%%%%%%%
%计二次拓展信源的霍夫曼信源编码
p = p'*p;
p = p(:)';
N=length(p);

symbols = cell(1,N);
for i = 1:N          % i 表第 i 个符号

```

```

        symbols{i} = ['x',num2str(i)];
    end
    [dict,L_ave] = huffmandict(symbols,p);
    dict = dict.';
    H = sum(-p.*log2(p));%计算信源信息熵
    yita = H/L_ave;%计算编码效率
    CODE = strings(1,N);%初始化对应码字
    for i=1:N                % i 表?第?个符号
        CODE(i) = num2str(dict{2,i});
    end
    %展?输出码字、平均码长和编码效率
    fprintf('\n 二次拓展信源运算结果:\n');
    disp('二次拓展信源符号: ');disp(dict(1,1:N));
    disp(['二次拓展信源对应概率: ',num2str(p)]);
    disp('二次拓展信源对应码字: ');disp(CODE);
    disp(['二次拓展信源平均码长: ',num2str(L_ave)]);
    disp(['二次拓展信源编码效率: ',num2str(yita)]);
    %%%%%%%%%%%%%%%
    %%%%%%%%%%%%%%%
    %三次拓展信源编码

    p = p(:)*p0;
    p = p(:)';
    N = length(p);

    symbols = cell(1,N);
    for i = 1:N                % i 表 i 第 i 个符号
        symbols{i} = ['x',num2str(i)];
    end
    [dict,L_ave] = huffmandict(symbols,p);
    dict = dict.';
    H = sum(-p.*log2(p));%计算信源信息熵
    yita = H/L_ave;%计算编码效率
    CODE = strings(1,N);%初始化对应码字
    for i=1:N                % i 表第 i 个符号

```

```

CODE(i)=num2str(dict{2,i});
end
%展示输出码字、平均码长和编码效率
fprintf('\n 三次拓展信源运算结果:\n');
disp('三次拓展信源符号: ');disp(dict(1,1:N));
disp(['三次拓展信源对应概率: ',num2str(p)]);
disp('三次拓展信源对应码字: ');disp(CODE);
disp(['三次拓展信源平均码长: ',num2str(L_ave)]);
disp(['三次拓展信源编码效率: ',num2str(yita)]);

```

八、实验数据及结果分析：

按照格式输入离散信源分布概率得出结果：

```

命令窗口
请输入离散信源概率分布，例如[p1, p2, ... pn]：
[0.2 0.8]

运算结果：
信源符号，
    'x1'    'x2'

对应概率： 0.2        0.8
对应码字：
    "1"    "0"

平均码长： 1
编码效率： 0.72193

二次拓展信源运算结果：
二次拓展信源符号，
    'x1'    'x2'    'x3'    'x4'

二次拓展信源对应概率： 0.04        0.16        0.16        0.64
二次拓展信源对应码字：
    "1 0 1"    "1 0 0"    "1 1"    "0"

二次拓展信源平均码长： 1.56
二次拓展信源编码效率： 0.92555

三次拓展信源运算结果：
三次拓展信源符号，
    'x1'    'x2'    'x3'    'x4'    'x5'    'x6'    'x7'    'x8'

三次拓展信源对应概率： 0.008        0.032        0.032        0.128        0.032        0.128        0.128        0.512
三次拓展信源对应码字：
    "1 1 1 1 1"    "1 1 1 1 0"    "1 1 1 0 1"    "1 1 0"    "1 1 1 0 0"    "1 0 1"    "1 0 0"    "0"

三次拓展信源平均码长： 2.184
三次拓展信源编码效率： 0.99166
fx >>

```

```
命令窗口
请输入离散信源概率分布，例如[p1, p2, ... pn]：
[0.1 0.3 0.6]

运算结果：
信源符号：
    'x1'    'x2'    'x3'

对应概率： 0.1        0.3        0.6
对应码字：
    "1 1"    "1 0"    "0"

平均码长： 1.4
编码效率： 0.92533

二次拓展信源运算结果：
二次拓展信源符号：
    'x1'    'x2'    'x3'    'x4'    'x5'    'x6'    'x7'    'x8'    'x9'

二次拓展信源对应概率： 0.01        0.03        0.06        0.03        0.09        0.18        0.06        0.18        0.36
二次拓展信源对应码字：
    "0 1 0 1 0 1"    "0 1 0 1 0 0"    "0 1 1 1"    "0 1 0 1 1"    "0 1 0 0"    "0 0 1"    "0 1 1 0"    "0 0 0"    "1"

二次拓展信源平均码长： 2.67
二次拓展信源编码效率： 0.97038

三次拓展信源运算结果：
三次拓展信源符号：
    'x1'    'x2'    'x3'    'x4'    'x5'    'x6'    'x7'    'x8'    'x9'    'x10'    'x11'    'x12'    'x13'    'x14'    'x15'    'x16'    'x17'

1 至 17 列

    'x1'    'x2'    'x3'    'x4'    'x5'    'x6'    'x7'    'x8'    'x9'    'x10'    'x11'    'x12'    'x13'    'x14'    'x15'    'x16'    'x17'

18 至 27 列

    'x18'    'x19'    'x20'    'x21'    'x22'    'x23'    'x24'    'x25'    'x26'    'x27'

三次拓展信源对应概率： 0.001        0.003        0.006        0.003        0.009        0.018        0.006        0.018        0.036        0.003        0.009
三次拓展信源对应码字：
1 至 6 列

    "0 0 0 1 0 1 ..."    "0 0 0 1 0 1 ..."    "0 0 0 1 0 1 ..."    "0 0 0 1 0 1 ..."    "0 1 0 0 1 1 0"    "0 0 1 0 0 1"

7 至 12 列

    "0 0 0 1 0 1 ..."    "0 0 1 0 0 0"    "0 0 1 1 1"    "0 0 0 1 0 1 ..."    "0 1 0 0 1 0 1"    "0 0 1 0 1 1"

13 至 19 列

    "0 1 0 0 1 0 0"    "0 0 0 1 0 0"    "0 0 0 0 1"    "0 0 1 0 1 0"    "0 0 0 0 0"    "1 1 1"    "0 1 0 0 1 1 1"

20 至 27 列

    "0 1 0 0 0 1"    "0 0 1 1 0"    "0 1 0 0 0"    "0 1 0 1"    "1 1 0"    "0 0 0 1 1"    "0 1 1"    "1 0"

三次拓展信源平均码长： 3.927
三次拓展信源编码效率： 0.98966
```

这里用 x_1, x_2, \dots, x_n 表示信源多次拓展后的码字。Huffman 编码算法非常实用、易于实现且编码效率高。通过分析拓展信源的编码平均码长和编码效率得知：信源的脱产次数 N 越大，霍夫曼的平均码长越长，编码效率也就越高。当 N 趋向无穷时编码效率为 1。

十、总结及心得体会：

通过本次实验，我掌握了霍夫曼信源编码的方法，利用 matlab 软件的 huffmandict 函数顺利的完成了霍夫曼编码的实验，进一步加深了我对霍夫曼信源编码的理解。

报告评分：

指导教师签字：