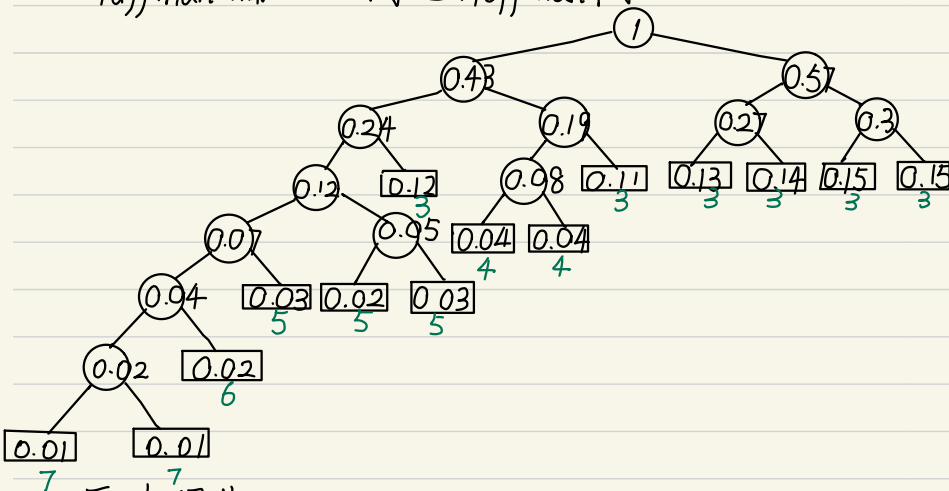


## 作业二

1. 经统计, 某种处理机14条指令的使用频度分别是: 0.01、0.15、0.12、0.03、0.02、0.04、0.02、0.04、0.01、0.13、0.15、0.14、0.11、0.03。试分别给出指令操作码的定长编码、Huffman编码、只能有两种码长且平均码长尽可能短的扩展编码, 并分别计算这3种编码的平均码长。

解: 定长编码的二进制码位数为  $\lceil \log_2 14 \rceil = 4$  位,

Huffman 编码: 构造 Huffman 树



$$\begin{aligned} \text{平均码长} &= 7 \times 0.02 + 6 \times 0.02 + 5 \times 0.08 + 4 \times 0.08 \\ &\quad + 3 \times 0.8 \\ &= 0.14 + 0.12 + 0.4 + 0.32 + 2.4 \\ &= 0.26 + 0.72 + 2.4 \\ &= 0.98 + 2.4 \\ &= 3.38 \end{aligned}$$

扩展编码:

让使用频率较高的 6 种指令用 3 位操作码编码表示, 用 5 位编码各扩展出 4 条使用频率较低的指令。

平均码长:

$$3 \times (0.15 + 0.15 + 0.14 + 0.13 + 0.12 + 0.11) + 5 \times (1 - 0.8) = 3 \times 0.8 + 5 \times 0.2 = 3.4$$

具体编码:

指令频率	定长编码	Huffman编码	扩展编码
1	001	0000000	000
2	0.01	0001	001
3	0.02	0010	010
4	0.02	0011	011
5	0.03	0100	100
6	0.03	0101	101
7	0.04	0110	11000
8	0.04	0111	11001
9	0.11	1000	11010
10	0.12	1001	11011
11	0.13	1010	11100
12	0.14	1011	11101
13	0.15	1100	11110
14	0.15	1101	11111

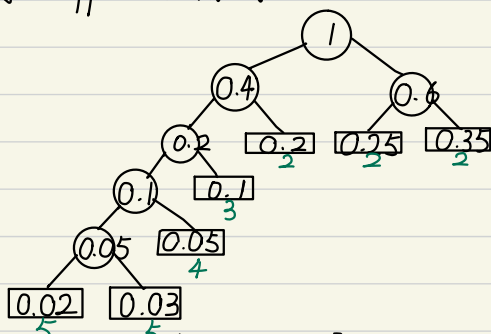
平均码长: 定长编码 4 Huffman 3.38  
扩展编码 3.4

2. 一台模型机共有7条指令，各指令的使用频度分别是35%、25%、20%、10%、5%、3%、2%，有8个通用数据寄存器，2个变址寄存器。

(1) 要求操作码的平均长度最短，请设计操作码的编码，并计算所设计操作码的平均长度。

(2) 设计8位字长的寄存器—寄存器型指令3条，16位字长的寄存器-存储器型变址寻址方式指令4条，变址范围不小于正、负127。请设计指令格式，并给出各字段的长度和操作码的编码。

解：(1) 采用 Huffman 编码



具体编码:	指令频率	Huffman 编码
1	0.02	00000
2	0.03	00001
3	0.05	0001
4	0.10	001
5	0.20	01
6	0.25	10
7	0.35	11

$$\begin{aligned}
 \text{平均长度: } & 5 \times 0.05 + 4 \times 0.05 + 3 \times 0.1 + 2 \times 0.8 \\
 & = 0.45 + 0.3 + 1.6 \\
 & = 2.35
 \end{aligned}$$

(2) 8位字长的寄存器-寄存器型指令

寄存器地址 3位 操作码 2位

源操作码 OP	源寄存器 R1	目的寄存器 R2
2	3	3

三条指令操作码分别为 00 01 10

16位字长的寄存器-存储器型变址寻址方式指令:

$2^7 = 128$  -127~128 需要 8 位存储

操作码 OP	通用寄存器	变址寄存器	偏移地址
4	3	1	8

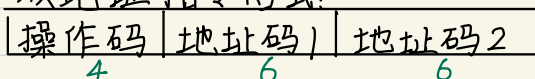
四条指令操作码分别为 1100 1101 1110 1111

3. 某处理机的指令字长为16位，有双地址指令、单地址指令和零地址指令三类，并假设每个地址字段的长度均为6位。

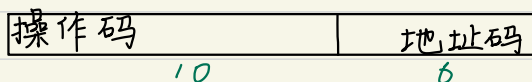
(1) 如果双地址指令有15条，单地址指令和零地址指令的条数基本相同，问单地址指令和零地址指令各有多少条？并且为这三类指令分配操作码。

(2) 如果要求三类指令的比例大致为1: 9: 9，问双地址指令、单地址指令和零地址指令各有多少条？并且为这三类指令分配操作码。

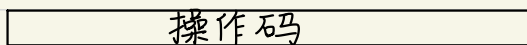
解: 双地址指令格式:



单地址指令格式:



零地址指令格式:



(1) 双地址指令 15 条: 操作码: 0000 ~ 1110

设单地址指令  $x$  条, 则零地址指令  $(64 - x) \cdot 64$  条,

令  $x \approx (64 - x) \cdot 64$  得  $x = 63$

单地址指令 63 条: 操作码: 1111 000000 ~

1111 111110

零地址指令 64 条: 操作码: 1111 111111 000000 ~

1111 111111 111111

(2) 由 (1) 可知, 1: 9: 9 中 9: 9 可以是单地址指令条数比零地址指令条数, 9: 9 为近似比, 实际比为 63: 64

设双地址指令  $x$  条, 则单地址指令  $(16 - x) \cdot 64$  条

$x : (16 - x) \cdot 63 = 1 : 9$  得  $x = 14$

故双地址指令14条,操作码: 0000~1101

单地址指令126条,操作码: 1110 000000 ~

1110 111110

1111 000000 ~

1111 111110

零地址指令128条,操作码: 1110 1111 11 000000 ~

1110 1111 11 11 1111

1111 1111 11 000000 ~

1111 1111 11 11 11 11