

第二章习题

9. (1) 1MB

(2) 14KB

(3) 不能。lui指令可以将一个16位的立即数左移16位，然后存储到目标寄存器的高16位。但是lui指令只能控制高16位，低16位默认为0，所以它只能设置目标寄存器的高16位地址。而jalr指令需要一个32位的目标地址，它将目标地址存储在寄存器中，并将程序计数器设置为当前PC+4的地址。因此，单独使用lui指令和jalr指令无法实现任意32位绝对地址的跳转操作，需要借助其它方法实现。

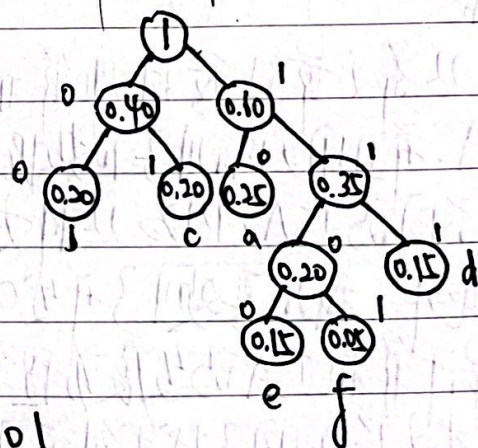
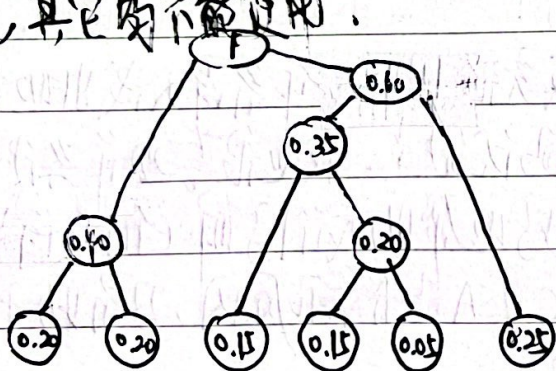
10. 条件: (1) 指令只能使用GPR寄存器，不能使用FPU寄存器

(2) 指令只能使用最多两个寄存器作为源操作数，不能使用立即数或内存操作数。

(3) 指令操作数必须满足一些特殊限制，例如必须是连续的寄存器编号，或者寄存器编号之间差值必须是某个特定值。

对于RVC指令集中各个指令类型，它们都只能使用16个通用寄存器(R0~R15)，其它均不能使用。

18.



a: 10 b: 00 c: 01 d: 111 e: 1100 f: 1101

$$\sum p_i l_i = 2.55$$

$$R = 1 - \frac{\sum_{i=1}^n p_i \log_2 p_i}{2.55} = 1 - \frac{2.46}{2.55} = 3.3\%$$

19. (1) 当函数调用层数过多时，栈的大小可能会超出系统限制，导致栈溢出。这种情况通常出现在递归函数中，如果递归深度太大或递归条件不正确，就会导致栈的空间不足。当栈空间已经耗尽时，继续压入数据会导致栈溢出。此时，程序会抛出异常并终止执行，从而防止对计算机系统造成更大的危害。

(2) 1. 优化递归函数算法, 尽量减少递归深度

2. 尽量使用迭代而不是递归

3. 增加栈空间大小。在一些编程语言中, 可以通过修改栈大小来增加栈空间从而避免栈溢出。

20. $\Gamma_1(F_1)$

参数 F_1

s_0

t_0

$\Gamma_1(F_2)$

参数 F_2

参数 F_2

s_1

t_1

$\Gamma_1(F_3)$

参数 F_3

参数 F_3

s_0

t_0