

2023. 3. 28

9 1) 范围为 $\pm 1\text{MB}$ 地址空间

2) 范围为 $\pm 4\text{KB}$ 地址空间

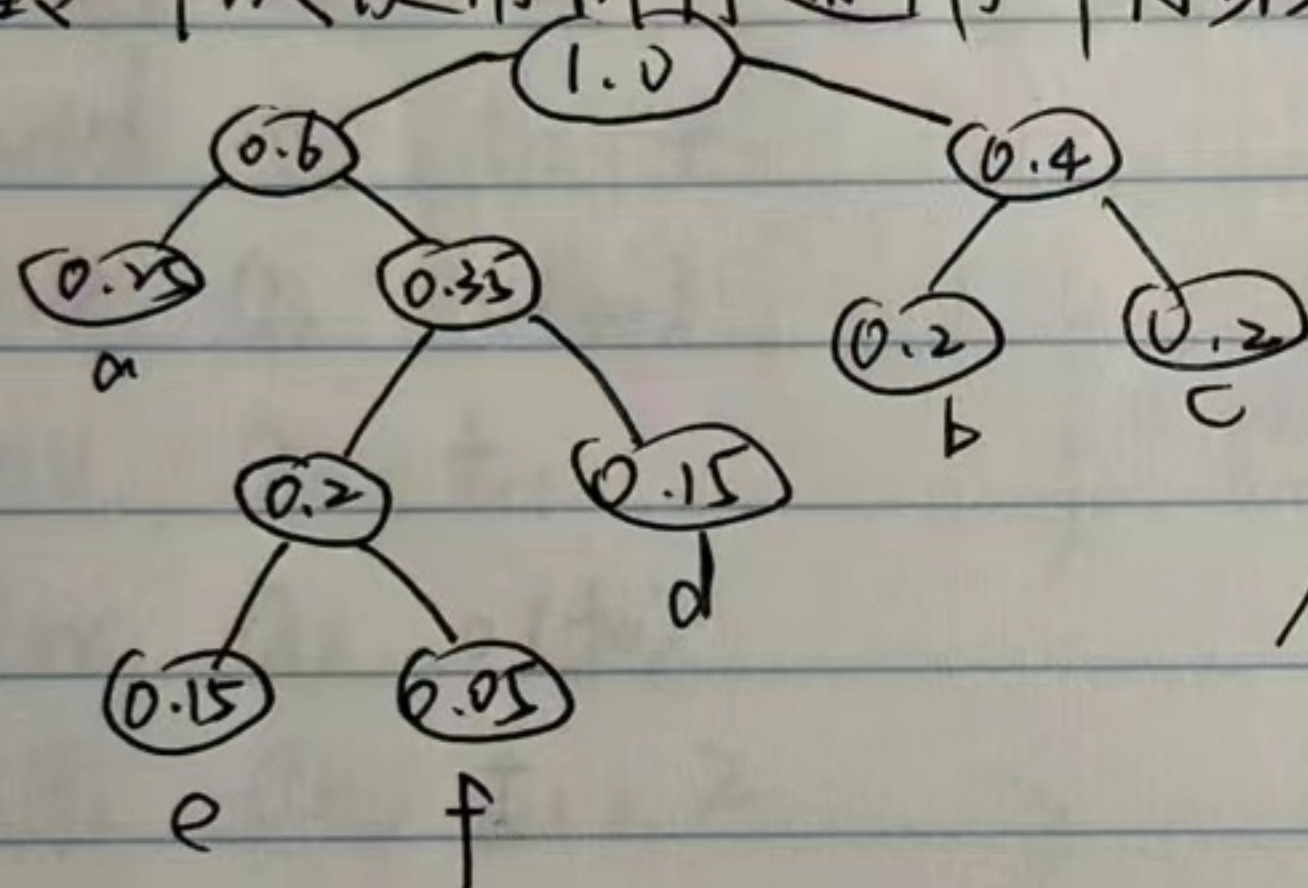
3) 是。先用 `lui` 加载高 20 位 `imm` 再用 `jal` `jalc` 加载低 12 位
跳转地址, 合成 32 位跳转地址

10 在 32 位指令中有些指令的高 16 位都是 0, 这些指令可被压缩成 16 位指令

不是。许多 RV32C 指令只能访问常用寄存器 (a_0-a_5, s_0-s_1, sp, ra)

但有些指令可以使用所有通用寄存器作为操作数

18



平均长度 $\sum_{i=1}^6 p_i l_i =$

$$2 \cdot (0.25 + 0.2 + 0.2) + 3 \times 0.15$$

$$+ 4 \times (0.15 + 0.05) = 2.55$$

$$\text{信息冗余量 } R = 1 - \frac{\sum_{i=1}^6 p_i \log_2 p_i}{\sum_{i=1}^6 p_i l_i}$$

$$= 3.296\%$$

19 1) 函数的调用的参数是通过栈空间来传递的, 在调用过程中会占用线程的栈资源。而递归调用, 只有走到最后的结束点后函数才能依次退出, 而未到达最后的结束点之前, 占用的栈空间一直没释放。如果递归调用次数过多, 则可能导致占用的栈资源超过线程的最大值, 从而导致栈溢出, 导致程序异常退出。

2) 减少栈空间需求, 不要定义占用内存较多的变量; 函数参数不要传递大型结构或对象, 多使用引用和指针作参数; 减少递归调用层次, 多使用循环调用; 通过尾递归优化。

20

$ra(F_1)$

$So(F_1)$

$ra(F_2)$

$So(F_2)$

$Si(F_2)$