

第6次作业

9. (1) $\pm 1\text{MB}$ 地址空间

(2) $\pm 4\text{KB}$ 地址空间

(3) 可以, 先用 `lui` 得高20位, 再由 `jalc` 加上低12位后跳转

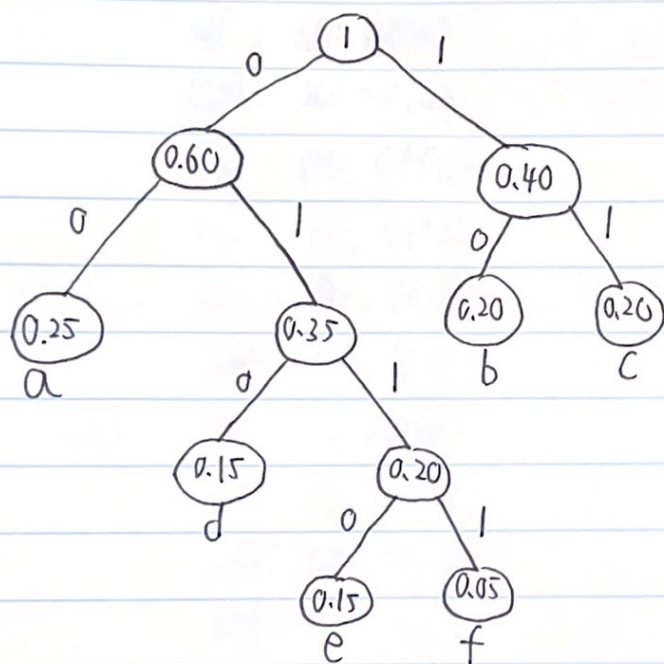
10. 条件: 立即数或者地址的长度足够小; 或其中一个寄存器是 `X0`, `X1`, `X2`;

目的寄存器和第一个源寄存器相同; 被使用的寄存器是8个最常用寄存器

以上4个条件至少满足其一

不是, 只有专用的存储器读指令和存储器写指令进行访问存储器操作

18



a: 00 b: 10 c: 11 d: 010 e: 0110 f: 0111

平均长度: $\sum_{i=1}^6 p_i l_i = 2.55$

信息冗余度: $1 - \frac{-\sum_{i=1}^6 p_i \log_2 p_i}{\sum_{i=1}^6 p_i l_i} = 0.03296$

19 (1) 原理: 对于嵌套调用, 我们一般以 SP 位移一定空间 (例如 32 位) 作为栈, 存储返回地址等信息, 因此, 若调用次数过多 (n 次), 会向 1 个方向位移 n 倍空间 (如 $n \times 32$ 位), 而内存空间有限, 可能溢出

(2) 在每次嵌套调用时, 生成栈的空间尽可能小, 以缓解栈溢出问题

(也可对每一层, 使用一公用寄存器, 将返回地址的相对位置计入其低位, 从而通过移位等操作获得地址层层返回, 但对总位数有限制)

20. F1: 1 个输入参数 使用 t0 和 s0

F2: 2 个输入参数 使用 t0-t1 及 s0-s1 返回一个 int 值

F1 执行中调用 F2, F2 执行过程中调用 F3

ra (F1)

s0 (F1)

t0 (F1)

a0 (F1)

a1 (F1)

t1 (F1)

ra (F2)

s0 (F2)

s1 (F2)

t0 (F2)

t1 (F2)

a0 (F2)

a1 (F2)