

9. (1) JAL 跳转的地址有  $+/- 1\text{MB}$  的范围  $-2^{20} - +2^{20}$

(2) 条件分支指令可跳转的地址空间范围  $-2^{12} - 2^{10} +/- 4\text{KB}$

(3). 可以, JALR 偏移的地址范围可以使 2 个指令序列在任意 32 位绝对地址中跳转

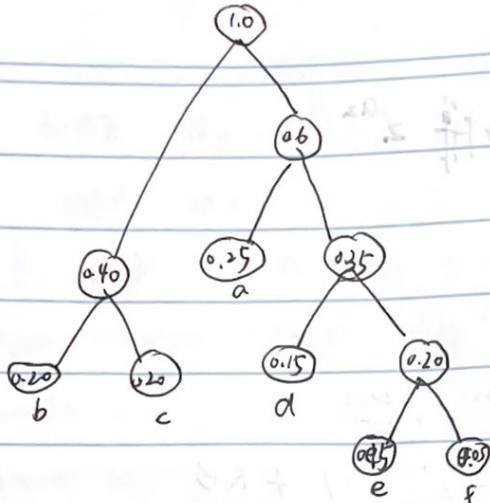
10. 条件: 满足下列情形 ① 立即数或地址偏移足够小 ② 其中个寄存器为  $x_0, x_1$  或  $x_2$

③  $rd$  和  $rs1$  为同一寄存器 ④ 使用的寄存器是 8 个最常用的

不可以完整使用 32 个通用寄存器, 指令长度受限

编码 1

118.



$$\begin{aligned} a & 0.25 \quad 10 \quad \bar{L} = \sum_{i=1}^6 p_i l_i \\ b & 0.20 \quad 100 \quad = 2 \times (0.15 + 0.2) + 0.2 \\ c & 0.05 \quad 01 \quad + 3 \times (0.15 + 0.2) + 4 \times (0.15 + 0.2) \\ d & 0.15 \quad 110 \quad = 2 \times 0.65 + 3 \times 0.15 + 4 \times 0.2 \\ e & 0.15 \quad 1110 \quad = 1.3 + 0.45 + 0.8 \\ f & 0.05 \quad 1111 \quad = 2.55 \end{aligned}$$

$$-\sum_{i=1}^6 p_i \log_2 p_i = 2.46 \quad R = 1 - \frac{-\sum_{i=1}^6 p_i \log_2 p_i}{H} = 1 - \frac{2.46}{2.55} = 0.0353$$

19. (1) 函数调用层次太深, 调用递归时要在栈中不断保存变量, 若调用太深, 会使栈溢出, 无法返回调用的返回地址。

(2) 针对函数调用过深, 可以将递归改为迭代, 或定义一个模拟栈的数据结构

20.

$ra(F_1)$
$to(F_1)$
$so(F_2)$
$tl(F_1)$
$sl(F_2)$