

9. (1) jal 指令：地址空间范围： $2^{20} - 1 - (-2^{20}) + 1 = 2^{21} = \pm 1\text{MB}$

(2) beq 指令：地址空间范围： $2^{20} - 1 - (-2^{12}) + 1 = 2^{13}\text{B}$

(3) 可以，利用 lui 指令将立即数高 20 位 ~~左移~~ 存到寄存器的高 4 位，然后利用 jalr 与低 12 位相加即可实现跳转。

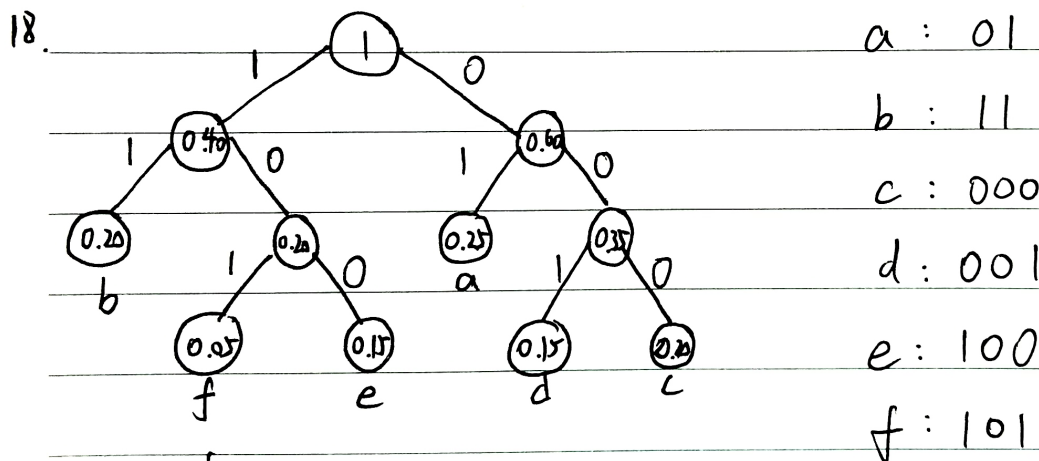
10. (1) 立即数或地址偏移量很小

(2) 其中一个寄存器调用 x_0 , x_1 或 x_2

(3) 目的寄存器和第一个源寄存器相同

(4) 寄存器使用的是八个常用寄存器

不是，CIW, CL, CS 和 CB 只能用其中的 8 个寄存器 ($x_8 - x_{15}$)



平均: $\bar{d} = \sum_{i=1}^n P_i d_i = 0.5 + 0.4 + 0.6 + 0.45 + 0.45 + 0.15 = 2.55$

最短平均: $H = -\sum_{i=1}^n P_i \log_2 P_i \approx 2.466$

信号: $k = \frac{\bar{d} - H}{\bar{d}} = 0.033$

19. (1) 调用栈的大小有限，* 函数每次被调用都要在原有基础上 ^{分配} 利用栈空间，超过最初使用调用栈大小就会溢出。

(2) 增加栈大小；优化算法；减少局部变量使用；使用动态内存分配，使用循环替代递归；使用尾递归。

20. $r\alpha (F_1)$

$\alpha_0 (F_1)$

$s_0 (F_1)$

$t_0 (F_1)$

$r\alpha (F_2)$

$\alpha_0 (F_2)$

$\alpha_1 (F_2)$

$t_0 (F_2)$

$t_1 (F_2)$

$s_0 (F_2)$

$s_1 (F_2)$