

4.2.

$-2^{19} \sim 2^{19}-1$

9. (1). 相较于前几, 可以跳转的地址范围是 $0 \sim 2^{20}-1$ 之间的任意地址

(2). $-2^{11} \sim 2^{11}-1$ 之间的任意地址

(3). 可以。具体来说, 首先使用 lwi 指令将当前的 PC 值转换为 32 位无符号整数, 然后使用 jalr 指令将该整数转换为跳转到相应地址的指令。

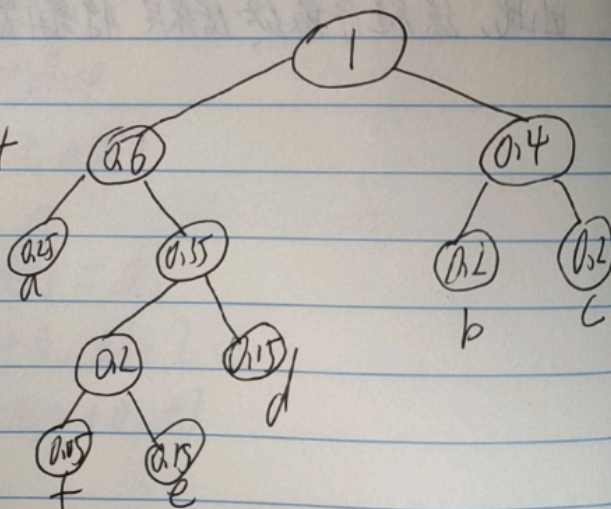
10. 答: (1) 条件: 如果该指令的操作码和操作数长度不超过 16 位, 且该指令在 RVC 指令集中有对应的 16 位编码。

(2) 并不是所有的指令都可以。在 RVC 中, 每个指令必须使用完整的 32 个通用整型寄存器, 这是因为 RVC 指令集是基于 32 位 Intel x86 指令集中, 并且 RVC 处理器需使用这些寄存器来存储和处理指令的各个部分。

18. 解: 按照霍夫曼编码如图所示

$$\begin{aligned} \text{计算平均长度 } L &= 0.25 \times 2 + (0.05 + 0.15) \times 4 \\ &\quad + 0.15 \times 3 + (0.2 + 0.2) \times 2 \\ &= 2.55 \end{aligned}$$

$$R = 1 - \frac{2.55}{\log_2 6} = 0.0135$$



19. 解: (1) 每次函数调用都会在栈中分配一定的内存空间, 用于存储函数的参数、局部变量和返回地址等信息。若函数嵌套调用层数过多, 栈中的内存空间会被耗尽, 导致栈溢出。

(2) ①. 增加栈的大小, 可以通过修改操作系统的设置或编译器的参数来实现。

②. 使用循环代替递归, 避免递归调用过深。

③. 优化算法, 减少递归调用次数。

④ 使用动态规划等非递归算法, 避免递归调用

20.

$ra(F1)$

$t0(F1)$

$s0(F2)$

$a0(F1)$

◀ #中间 应该有三行

$ra(F2)$

$t0(F2)$

$t1(F2)$

$s0(F3)$

$s1(F3)$

$a0(F2)$

$a1(F2)$