

第六周作业

9. (1) 由于jal指令含20位有符号立即数编码, 所以其跳转范围可以从当前PC的相对地址 -2^{19} 到 $2^{19}-1$ 的地址空间。所以jal指令可跳转的地址空间的跨度为 2^{20} 个地址。但还需考虑jal指令的指令长度为4字节, 实际上可跳转的地址范围需要考虑指令长度的影响。

(2) 条件分支指令(如bne)含12位有符号立即数编码, 所以其跳转范围相对当前PC可跳转的地址空间范围是 -2^{12} 到 $2^{12}-1$, 指令可跳转的地址空间跨度为 2^{13} 个地址。但bne指令的指令长度也为4字节, 实际可跳转的地址范围需考虑指令长度影响。

(3) 可以使用一条lui指令和一条jalr指令的组合完成任意32位绝对地址的跳转操作。首先使用lui指令将一个20位的立即数左移12位, 得到一个32位的值, 并将其存储到一个寄存器中; 然后jalr指令可以将寄存器中的值作为跳转地址, 并将当前指令的地址保存到另一个寄存器中。这样通过将lui指令和jalr指令结合, 就可以跳转到任意32位绝对地址。

10. 答: 在RVC中, 一条常用的32位指令可以被压缩为16位RVC指令的条件如下:

① 该指令必须属于RISC-V的32位指令集(即I、R、S、B、J^{imm}类型指令), 而不属于扩展指令集(如M、A、F、D等)的指令

② 该32位指令必须支持RVC指令集中提供的对应指令。

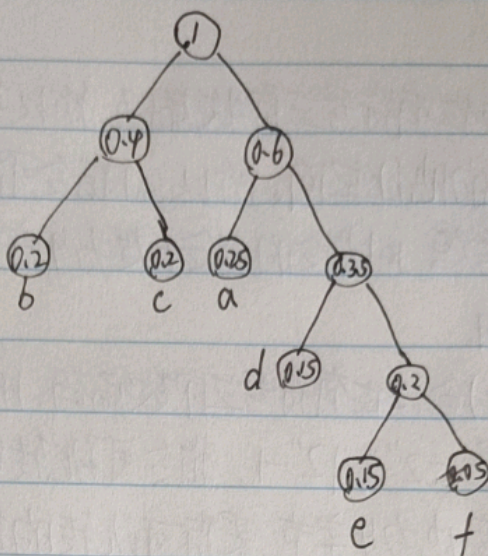
③ 指令中涉及的寄存器只能是16个通用寄存器 $x_0 \sim x_{15}$ 中的一个, 立即数的范围必须在 $-2^{12} \sim 2^{12}-1$ 之间, 指令中的操作数寻址方式必须是寄存器直接寻址或立即数寻址。

RVC中不是各类型的指令都可以使用完整的32个通用整型寄存器, 不同类型的指令可以使用的通用整型寄存器数量是不同的。其中算术/逻辑型指令、分支型指令、条件型指令、跳转型指令等可以使用32个通用整型寄存器。但一些指令只能使用16个通用整型寄存器。

18. 解

霍夫曼编码为

a_i	P_i	l_i	$c(a_i)$
a	0.25	2	10
b	0.20	2	00
c	0.20	2	01
d	0.15	3	110
e	0.15	4	1110
f	0.05	4	1111



操作码平均长度

$$\sum_{i=1}^6 P_i l_i = 2.55$$

信息冗余度

$$R = 1 - \frac{\sum_{i=1}^6 P_i \cdot \log_2 P_i}{\log_2 6} = 1 - \frac{2.4659}{2.5849} = 0.046$$

19. (1) 栈溢出是指程序在执行函数调用时, 将大量数据压入栈中, 导致栈内存耗尽, 继续向栈中压入数据时就会发生栈溢出的情况。函数嵌套调用层数过多时, 每次调用都会向栈中压入一些数据, 如局部变量等, 都会占用栈内存。若嵌套层数过多, 也容易压入过量数据导致栈溢出。递归会反复调用一个函数, 若一直不满足终止条件就会陷入死循环, 不断调用自身, 不断向栈中压入数据, 直到栈溢出。栈溢出会导致程序崩溃或不可预测的错误。

(2) 缓解或避免栈溢出问题的方法:

① 减小递归深度: 可以通过循环或者尾递归的方式来减少递归深度, 从而避免栈溢出问题。

② 增大栈空间: 在一些需要大量栈空间的场景下, 可通过增加栈空间的大小来避免栈溢出。

③ 使用堆内存: 栈内存通常比堆内存小, 因此可使用堆内存代替栈内存。

④避免过多的局部变量:过多的局部变量也可能导致栈溢出。可以将局部变量转换为全局变量或静态变量,减少栈内存的使用。

⑤使用动态内存分配,避免在栈中分配过多的内存。

20. 答:

ra(F1)

a0(F1)

t0(F1)

s0(F1)

ra(F2)

a0(F2)

a1(F2)

t0(F2)

t1(F2)

s0(F2)

s1(F2)