

Chapter 2 Homework (3)

▷ 9. 回答以下问题:

1) jal 指令包含 20 位的有符号立即数编码, 该指令相较于当前的 PC 可以跳转的地址空间范围是多少?

Offset 有 20 位, 即 $2 \times 2^{19} = 2^{20} = 1\text{MB}$, 故空间地址范围为 $\pm 1\text{MB}$

? Offset 有 20 位即 20 bits, 其最高为符号位, 其空间地址跳转范围为 $\pm 1\text{MB}$

2) 条件分支指令包含 12 位的有符号立即数编码 (B-type), 这类指令相较于当前的 PC 可以跳转的地址空间范围是多少?

条件分支的地址范围为 $\pm 4\text{KB}$ 。

3) 是否可以使用一条 lui 指令和一条 jalr 指令的组合完成任意值 32 位绝对地址的跳转操作?

可以实现, 通过 lui 指令左移 12 位加载后 20 位的 imm, 然后通过 JALR 加载低 12 位跳转地址, 合成 32 位跳转地址。

▷ 10. 调查 RVC 压缩指令集的编码, 说明一条常用的 32 位指令能够被压缩为 16 位 RVC 指令的条件是什么? RVC 是哪些类型的指令是否都可以使用完整的 32 个通用寄存器。

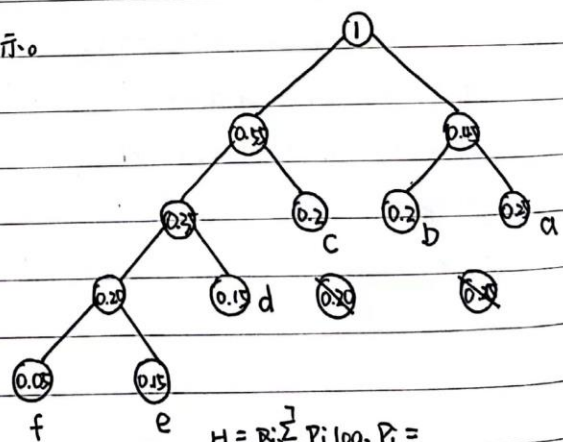
① 立即数或者地址偏移量较小时; ② 其中一个寄存器是 x_0 、ABI 软浮点寄存器 (x_1)、ABI 栈寄存器 (x_2);

③ 目标寄存器和第一个源寄存器相同; ④ 最常用情况下使用 32 位寄存器。

并非所有指令可以完整使用 32 个 RVI 寄存器, 其中格式为 CLW、CL、CS 和 CB 被限制只能使用所有 32 个寄存器中的 8 个。

▷ 18. 有一组操作码, 它们的出现几率如下表所示。

a_i	P_i
ba	0.25
b	0.20
dc	0.20
d	0.15
e	0.15
f	0.05



$$H = -\sum_{i=1}^7 P_i \log_2 P_i = -\sum_{i=1}^7 P_i \log_2 P_i = 0.2 + 0.6 + 0.45 + 0.4 + 0.4 + 0.5 = 2.55$$

请按照霍夫曼编码对这组操作码进行编码, 计算操作码的平均长度和信息冗余度。

a = 00 b = 01 c = 10 d = 110 e = 1110 f = 1111

D19. 回答以下问题:

1) 当函数嵌套调用层数过多(例如递归陷入死循环时),可能会造成栈溢出,请简述其原理。
函数递归调用时,系统要在栈中不断保存函数调用栈的变量,如果递归调用太深,就会造成栈溢出,这时递归无法返回。再有,当函数调用层次过深时也可能导致栈无法容纳这些调用的返回地址而造成栈溢出。

2) 有什么办法可以缓解或避免特定情况下的栈溢出问题?

尾递归优化: 函数调用会在内存形成一个“调用记录”,又称“调用帧”,保存调用位置和内部变量等信息。如果在函数A的内部调用函数B,在A调用记录的上方,会形成一个B的调用记录。B结束后,将结果返回A,其调用记录才会消失,从而形成“调用栈”。“尾调用优化”即只保留内层函数的调用记录,只存在一个调用记录,所以不存在“栈溢出”错误。

D20. 假设有三个函数: F_1 , F_2 和 F_3 。其中 F_1 包含 1 个输入参数, 计算过程使用寄存器 t_0 和 S_0 ; F_2 包含 2 个输入参数; F_2 包含 2 个输入参数, 计算过程使用寄存器 t_0-t_1 及 S_0-S_1 , 返回一个 int 值。 F_1 执行过程中会调用 F_2 , F_2 执行过程中会调用 F_3 。下表模拟了 F_1 执行过程中栈的内容, 其中第一行为 F_1 函数被首次调用时 sp 寄存器指向的位置。请在表中填入当 F_2 函数首次调用 F_3 前栈内保存的可能内容, 并在每行的括号内标注该值是被哪个函数所保存的。

ra(F_1)	? 没太搞懂(问)
S_0 (F_1)	
t_0 (F_1)	
ra(F_2)	
t_0 (F_2)	
t_1 (F_2)	
S_1 (F_2)	
t_0 (F_2)	
S_0 (F_1)	