

9. 回答以下问题：
- 1) jal 指令包含 20 位的有符号立即数编码 (J-type)，该指令相较当前 PC 可以跳转的地址空间范围为多少？
  - 2) 条件分支指令（如 bne）包含 12 位的有符号立即数编码 (B-type)，这类指令相较当前 PC 可以跳转的地址空间范围为多少？
  - 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 32 位绝对地址的跳转操作？
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 中各类型的指令是否都可以使用完整的 32 个通用整型寄存器？

9. (1) 该指令相较当前 PC 可以跳转的地址范围为  $(-2^{19} \sim 2^{14}-1)$

(2) 该指令相较当前 PC 可以跳转的地址范围为  $(-2^{11} \sim 2^{11}-1)$

(3) 是；

$lui \quad rs1, imm16$

$jalr \quad rd, rs1, imm12$

10. RISC-V 压缩指令基于 RISC-V 指令集压缩特性，采用类似变长编码的方式实现。在 32 位指令中，有些指令的高 32 位都是 0，这些指令可以被压缩成 16 位指令。这些压缩指令可以在硬件中解压后执行，以实现与 32 位指令相同的操作。

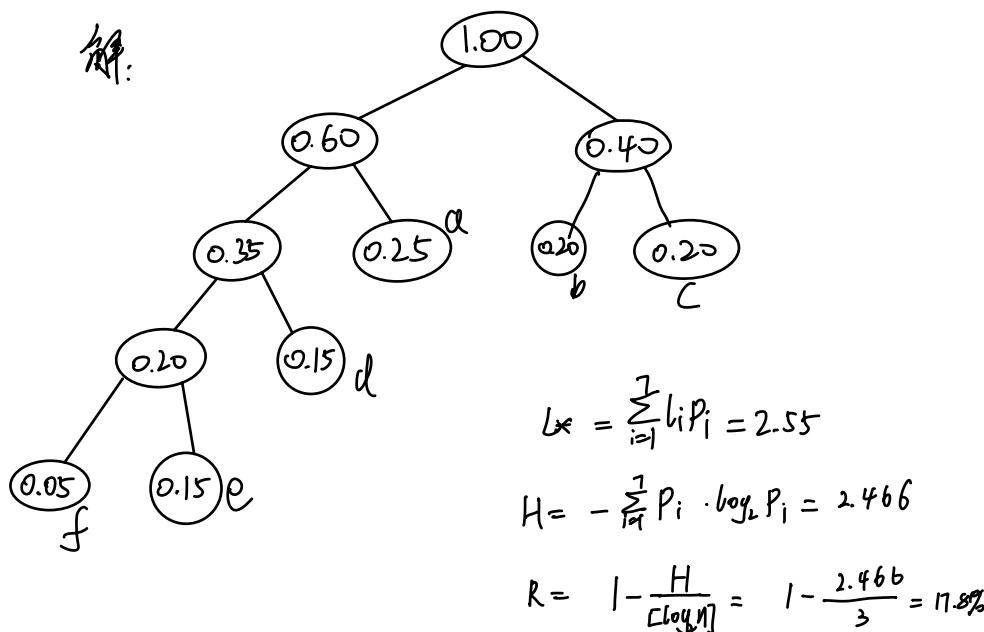
RVC 指令集中有 16, 32, 48 位；48 位指令需要使用一个 32 位指令和一个 16 位指令来实现。有些指令的编码空间小，因此不是所有的指令都支持 32 个寄存器，故 RVC 中矮型的指令并不都可使用完整的 32 位通用整型寄存器。

18. 有一组操作码，它们的出现几率如下表所示。

$a_i$	$p_i$
a	0.25
b	0.20

c	0.20
d	0.15
e	0.15
f	0.05

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。



19. 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。
- 2) 有什么办法可以缓解或避免特定情况下的栈溢出问题？

解：(1) 由于函数每嵌套一次会划分一次栈，直到嵌套到最后一个函数时，才会依次释放栈的空间。当函数嵌套过多时，创建的栈帧较多，可能会造成栈溢出。

(2) ①增加栈大小，例如在Linux系统下，可以使用ulimit命令来设置栈大小。

②使用静态分析工具，例如，使用Clang编译器的AddressSanitizer工具可以在编译时自动插入代码来检测栈溢出和内存泄漏等问题。

20. 假设有三个函数：F1、F2 和 F3。其中 F1 包含 1 个输入参数，计算过程使用寄存器 t0 和 s0；F2 包含 2 个输入参数，计算过程使用寄存器 t0-t1 及 s0-s1，返回一个 int 值。F1 执行过程中会调用 F2，F2 执行过程中会调用 F3。下表模拟了 F1 执行过程中栈的内容，其中第一行为 F1 函数被首次调用时 sp 寄存器指向的位置。请在表中填入当 F2 函数首次调用 F3 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。  
第一行的内容已经给出。（可根据需要增删行数）

ra(F1)
de(F1)
to(F1)
so(F1)
ta(F2)
ao(F1)
di(F1)
fo(F2)
ti(F2)
so(F2)
si(F2)