

-0.5  
-11 46 43 85 6

9. (1) 去符号位 20位的立即数可表示 $-2^{19}$ 到 $2^{19}-1$ 之间的数值，在risc-v中须将偏移量左移一位再与PC高12位，PC+4和符号位相拼接，因此“jal”指令可以跳转到相对于当前PC±1MB范围的任何地址。

(2) 指令跳转范围为±4KB地址空间

(3) risc-v中可以使用一条lui指令和jalr指令的组合完成任意32位绝对地址的跳转操作，LUI指令可以将一个20位的立即数左移12位，得到一个32位的数，存放目标寄存器的高20位中，故LUI指令可以用于加载一个32位地址的高20位，JALR指令可以将指令地址加上一个12位的立即数，并将结果存放在目标寄存器中，同时将PC设置为当前指令地址加上一个偏移量，结合这两条指令，具体代码实现如下：

Lui x1, %hi( )

addi x1, x1, %lo( )

Jalr x0, x1, 0

10. 在RVC指令集中，可以将常用的32位指令压缩为一条16位的RVC指令条件为：

1. 指令没有使用特权指令

2. 没有使用64位寄存器

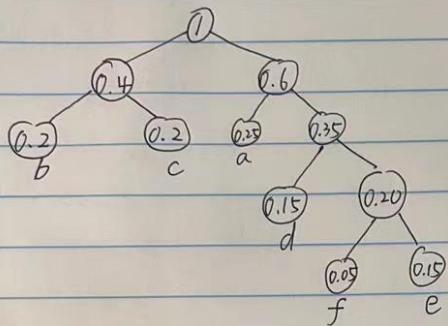
3. 指令操作的立即数是16位或更小的

4. 指令没有浮点操作码

在RVC指令集中，所有指令都可以使用完整的32个通用寄存器，但不支持浮点计数器的操作。

$$\begin{array}{r} -0.5 \\ -11.4643856 \end{array}$$

18. ai a b c d e f  
 $P_i$  0.25 0.20 0.20 0.15 0.15 0.05



$d_i$	$c(a_i)$	$l_i$	$P_i$	平均长度: $L = \sum P_i l_i = 2.55$
a	10	2	0.25	信息冗余度:
b	00	2	0.2	$R = 1 - \sum P_i \log_2 P_i$
c	01	2	0.2	$H = - \sum_{i=1}^n P_i \log_2 P_i = 1.59 + 2.46596$
d	110	3	0.15	信息冗余度 $= 1 - \frac{H}{L} = 3.296\%$
e	1111	4	0.15	
f	1110	4	0.05	

19. 如果函数嵌套调用的层数过多，栈空间可能会被耗尽，这种情况下，新的函数调用将无法分配所需的栈空间，从而导致栈溢出。当栈溢出时，程序将崩溃，并显示一个错误信息。

(2) ①增加栈空间

②减少函数调用嵌套层数

③检查代码中是否存在无限循环、递归调用等问题

④使用动态内存分配，但要注意及时释放已分配的内存。

-0.410

20.  $r_a(F_1)$

to

so

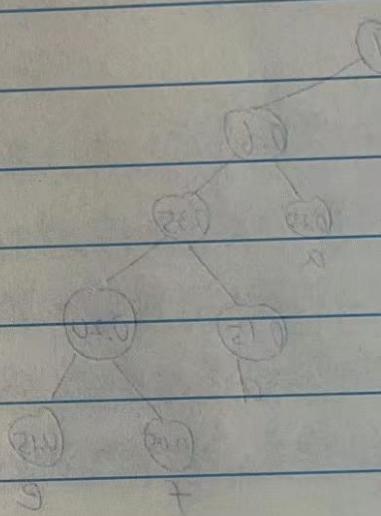
~~to~~  $r_a(F_2)$

~~to~~ so to

$t_1$

so

$s_1$



$r_a(F_1) = r_a(F_2) + s_1$

~~so~~  $r_a(F_2) = s_1$

$r_a(F_1) + r_a(F_2) = s_1$

$r_a(F_1) - r_a(F_2) = s_1$

效小(即慢、大(即快))，并从这二个方面进行评价。

效率(即快慢)、效果(即好差)、也称效果量(即好差)。