

第6周作业

9 (1) JAL指令可跳转到任何4KB对齐的地址,且只能使用20个位来存储,相对于当前PC可跳转地址范围 $\pm 1\text{MB}$ (220字节)

(2) 偏移量必须是4字节的倍数,并且只能使用12个位来存储,相对于当前PC跳转的地址范围是 $\pm 2\text{KB}$ (212字节)

(3) 可以

使用LUI指令将目标地址的高20位加载到寄存器中,使用JALR指令跳转到该寄存器中的地址,并以相同的寄存器作为目的地,以便将返回地址存储在该寄存器中

10 (1) 1. 该指令必须是16位RVC指令集所支持的指令

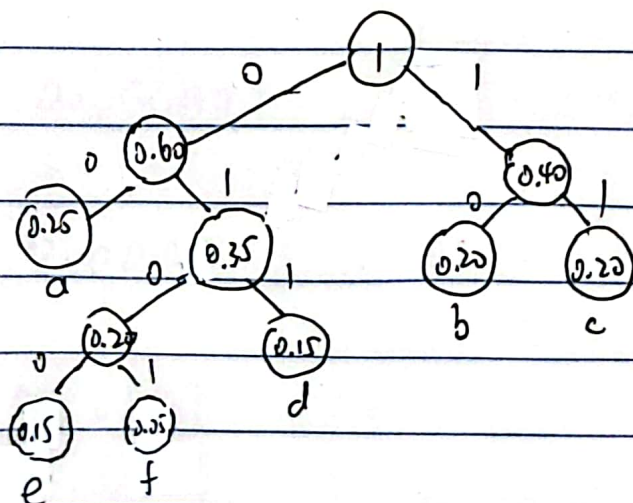
2. 指令所涉及的寄存器必须是16位RVC指令集所支持的寄存器,即X5至X15和X18至X27

3. 指令不能使用立即数,因为16位RVC只支持4位立即数

4. 指令必须符合特定的格式和约束条件

(2) 不是,L类型和S类型指令不能使用所有32个通用整型寄存器,因为它们需要使用额外的编码位来表示目标地址和立即数等信息

18



	p_i	l_i	
a	0.25	2	$\sum_{i=1}^6 p_i l_i = 0.5 + 0.4 + 0.44 + 0.45 + 0.6 + 0.2 = 2.55$
b	0.2	2	
c	0.2	2	$R = 1 - \frac{\sum_{i=1}^6 p_i \log_2 p_i}{[\log_2 n]} = 0.046$
d	0.15	3	
e	0.15	4	
f	0.05	4	

19 (1) 函数调用栈是一个后进先出的数据结构,用于存储程序的执行状态和局部变量等信息。每次函数调用时,都会将函数的返回地址和一些其他关键状态信息(如局部变量和参数)压入栈中,以便在函数返回时恢复现场。当递归调用深度过大或者函数调用层数过多时,栈空间被不断地使用,直至栈空间耗尽,就会发生栈溢出

(2) 1 优化代码:使用循环替代递归或改变算法

2 增加线程栈大小

3 使用动态内存分配

4 减少局部变量和过多参数 5 使用限制栈深度的方法

6 使用异常处理机制

20	F_2 局部变量	F_2
	F_2 局部变量	F_2
	F_1 的返回地址	CPU
	F_3 局部变量	F_2
	F_2 参数	F_1
	F_2 参数	F_1
	F_1 调用 F_2 函数的返回地址	CPU

