

第六次作业:

9. 1). $\text{current}(pc) \pm 2^{17}$

2) $\text{current}(pc) \pm 2^9$

3) lui 高位立即数加载指令 $x[ra] = \text{ sext immediate } [31:12] (<12)$

把 11-immediate 写入到 ra 的高 20 位.

jalu 间接跳转指令.

偏移的地址范围 $\pm 2KB (2^{12} = 4096 = 4KB)$

故 lui 和 jalu 配合使用可以完成任意 32 位绝对地址的跳转.

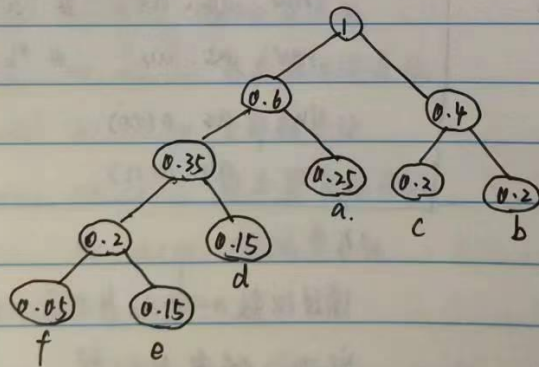
10. 条件: ① 对 4 个常用寄存器 ($a0-a5, s0-s1, sp$ 及 ra) 访问的频率远超过其他寄存器.

② 指令的写入目标是它的源操作数之一.

③ 立即数小, 且部分指令常用一些特定的立即数.

RVC 中的 lw , cl , cs 等类型的指令不能使用完整的 32 个通用寄存器

18.



$$\bar{l} = 0.2 + 0.6 + 0.45 + 0.5 + 0.4 + 0.4 = 2.55$$

$$H = - \sum_{i=1}^6 p_i \cdot \log_2 p_i \approx 2.00$$

$$R = 1 - \frac{2.00}{2.55} = \frac{1}{5}$$

而对每个程序来说,栈能使用的内存是有限的.

19. 1) 函数的每次调用都会开辟新的栈. 嵌套调用的层数过多,会造成栈溢出.
2) 栈溢出是缓冲区溢出的一种. 缓冲区溢出会导致缓冲区外的存储单元被覆
写. 缓解方法: ① 减小栈空间的需求, 将 `auto` 变量修改成指针, 从堆空间分配内存.
② 减少函数调用层数, 慎用递归函数.
③ 函数参数中不要大的数组/结构等, 使用指针.

20. `ra(F1)`

`to(F1)`

`so(F1)`

`ao(F1)`

`al(F1)`

`ra(F2)`

`to(F2)`

`tl(F2)`

`so(F2)`

`sl(F2)`