@ sll $t_1, t_1, 2$. # a*4.
add t_0, t_0, t_1.
sw. t_1, 0(t_0) # p[a] = a.

16. # Assume t_0, t_1 hold the pointer to a, b.

优化后:	
mv t_2, t_0.	lw t_2, 0(t_0).
mv t_0, t_1.	lw. t_3, 0(t_1).
mv t_1, t_2.	sw t_2, 0(t_1).
ret.	sw t_3, 0(t_0).
	ret.

17. @ 功能：计算 z^{30}. 存在 a_1 内.

3/9.

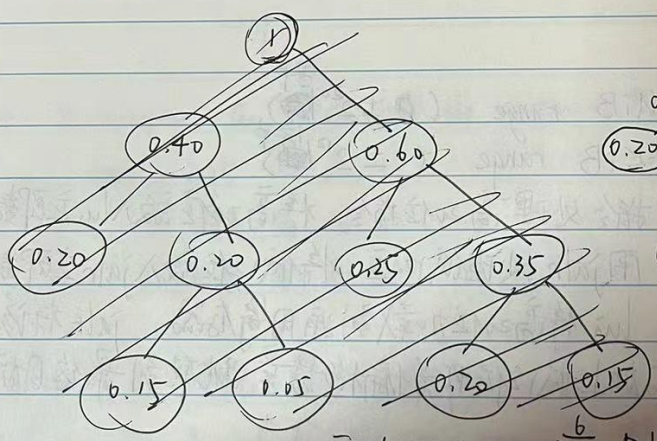
9. 1). target a ±1MiB range (@ $\pm 2^{20}$ 字节).
2). target a ±4KiB range ($\pm 2^{12}$ 字节).
3) 可以. 用 lui 指令处理高20位指令. 将高20位放入 lui 立即数字段. 用 jalr 处理低12位. 将低12位放入 jalr 立即数字段. lui 将高20位加载到通用寄存器. jalr 将该寄存器加上低12位作为偏移量. 跳转到最终目标地址.

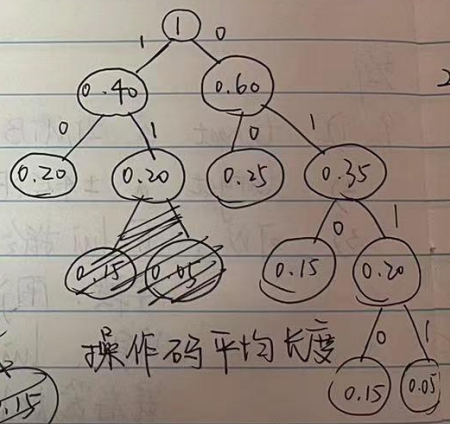
10. (1) ① 立即数或地址偏移量很小.
或 ② 其中一个寄存器是零寄存器 X_0，或ABI链接寄存器 X_1，或ABI堆栈指针 X_2 之一.
或 ③ 目标寄存器和第一个源寄存器相同.
或 ④ 使用的寄存器是最常用的8个之一.

(2) 可以. RVC指令与所有其他标准指令扩展兼容，允许16位指令和32位指令自由混合. 编码上虽然RVC只用了其中的8个寄存器，但任何指都可使用这些寄存器之外的其它寄存器，只是需要32位指令编码.

18.

a_i	a	b	c	d	e	f
p_i	0.25	0.20	0.20	0.15	0.15	0.05





操作码平均长度

平均码长 $\sum_{i=1}^{6} p_i l_i = 0.2\times2\times2 + 0.25\times2 + 0.15\times3 + 0.15\times4 + 0.05\times4 = 2.55$

熵 $H = -\sum_{i=1}^{6} P_i \times \log_2 P_i$ = =

熵 $H = -\sum_{i=1}^{6} P_i \log_2 P_i = 2.466$

信息冗余度 $R = 1 - \frac{H}{\sum_{i=1}^{6} P_i l_i} = 1 - \frac{2.466}{2.55} = 0.033 = 3.3\%$

9. (1) 每一次函数嵌套调用会开新栈，但栈的空间是有限的。不能无限开栈。层数过多后栈帧开辟的新空间占用了内存其它部分的空间。于是溢出。(越界覆盖)。

(2) ①使用循环代替递归，减少层数。
②减少局部变量的使用，解放栈空间。
③用迭代算法改写递归算法

20. S_0 (F_1 保存调用 F_2 前的 S_0)
t_0 (F_1 保存调用 F_2 前的 t_0)。
S_1 (F_2 保存调用 F_3 前的 S_1)。
t_1 (F_2 保存调用 F_3 前的 t_1)。
ra (F_2 保存 F_3 调用返回地址)。