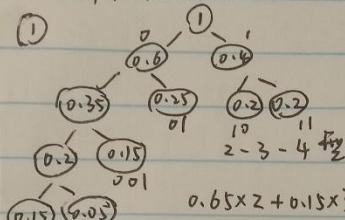
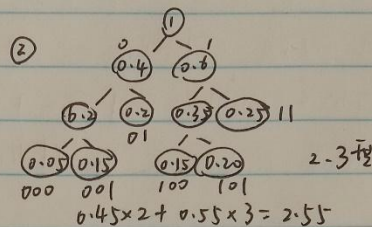


18. ~~0.25 + 0.15 = 0.4~~  $H = -0.25 \times \log_2 0.25 + 2 \times (-0.2) \times \log_2 (0.2) + 2 \times (-0.15) \log_2 (0.15) + (-0.05) \times \log_2 (0.05)$   
 $= 2.466$

有两种霍夫曼树:



$$0.65 \times 2 + 0.15 \times 3 + 0.20 \times 4 = 2.55$$



$$0.45 \times 2 + 0.55 \times 3 = 2.55$$

两种霍夫曼编码的平均长度相同 冗余度:  $1 - \frac{2.466}{2.55} = 0.033$

19. (1) 因为每次进行函数调用时, 都会存在一些压栈操作, 比如 ra, sp 等都会被存在堆栈里, 还有一些函数参数也会, 如果一直在发生调用而没有存在返回, 那么就没有单栈, 堆栈得不到释放, 造成栈溢出。

(2) 增加栈空间大小; 使用动态内存分配; 减少局部变量使用

20. a0(F1) 9. (1) 有符号:  $-2^{19} \sim 2^{19}-1$

t0(F1) (2) 有符号:  $-2^{32} \sim 2^{32}-1 = -2048 \sim 2047$  范围

s0(F2) (3) 可以, jalr 操作为  $rd = pc + 4$ ;  $pc = rs1 + imm$ ,

a0(F2) lui 操作为  $rd = imm \ll 12$  其中 imm 为高 20 位

a1(F2) 可以先用 lui 将绝对地址高 20 位存到 rs1 中, 然后 jalr 时将存有高 20

ra(F2) 位的寄存器放在 rs1, 低 12 位放在 imm, 从而实现任意 32 位绝对地

t0(F2) 址跳转。

t1(F2)

10. 32 位立即为 16 位的条件: ① 地址或偏移量比较小 ② 有一个寄存器为 x0, x1 或 x2 型

③ 目的寄存器与一个源寄存器相同 ④ 使用的寄存器是 8 个最常用的

不是。CR, C2 和 C55 可使用所有的, 但 C1W, CL, CS, CB 只能同其中的 8 个