

2-17 解释以下代码的功能

```
addi a0, x0, 0
```

```
addi a1, x0, 1
```

```
addi a2, x0, 30
```

```
loop: beq a0, a2, done
```

```
    slli a1, a1, 1
```

```
    addi a0, a0, 1
```

```
    j loop
```

```
done: #exit code
```

简单: 该代码完成了将 $a1$ (初值为 1)

左移 $a2$ 位 (30 位) 的操作

最终得到 $a1 = 2^{30}$

2-19 (1) jal 指令包含 20 位有符号立即数编码 (J-type), 该指令相较于当前 PC 可以跳转的地址空间范围是多少?

答: 范围为 $-2^{19} \sim +2^{19}-1$

(2) 条件分支指令 (如 bne) 包含 12 位的有符号立即数编码 (B-type) 这类指令相较于当前 PC, 可跳转的地址空间是?

答: 范围为 $-2^{11} \sim +2^{11}-1$

(3) lui + jalr 可否完成任意 32 位绝对地址的跳转操作?

答: 可以, lui 可以决定跳转地址的高 20 位, jalr 又可以决定跳转地址的低 12 位, 组合起来即任意 32 位绝对地址。

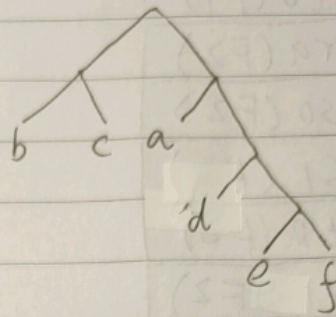
2-10, 调查 RVC 压缩指令集编码, 一条常用 32 位指令被压缩为 16 位 RVC 指令的条件是什么? RVC 中各类型指令是否都可以使用完整的 32 个通用整型寄存器?

答: 条件是有可以压缩的部分, 如压缩立即数位数、压缩寄存器地址位数、写入目标与源操作数之一使用同一地址,

通常 RVC 指令能用的寄存器很少, 有的指令只能使用常用的 8 个寄存器。

2-18. 进行霍夫曼编码并计算操作码平均长度、信息冗余度。

a_i	P_i
a	0.25
b	0.2
c	0.2
d	0.15
e	0.15
f	0.05



a_i	编码码
a	10
b	00
c	01
d	110
e	1110
f	1111

平均长度 $\sum_{i=1}^6 P_i \cdot l_i = 2.55$

信息冗余度 $R = 1 - \frac{-\sum_{i=1}^6 P_i \cdot \log_2 P_i}{2.55} = 0.03$

2-19: (1) 函数嵌套调用层数过多可能造成栈溢出, 请简述其原理。

答: 每次调用函数时, 需在内存中占用部分空间记录调用前的数据防止丢失, 调用层数过多, 内存不够用, 即造成栈溢出。

(2) 给定情况不解决或缓解栈溢出问题。

答: 减少栈空间需求, 将函数参数中传递的大型结构等改为指针传递, 或减少调用层次, 将递归改为循环等。

其中, var 是变量, expr1 和 expr2 是表达式
+ (加), - (减)

No.

Date

2-20: 解:

$ra(F1)$

$so(F1)$

$to(F1)$

$ra(F2)$

$so(F2)$

$si(F2)$

$to(F2)$

$t1(F2)$