CSAPP简要复习

# 重要题型1·IEEE754浮点数表示

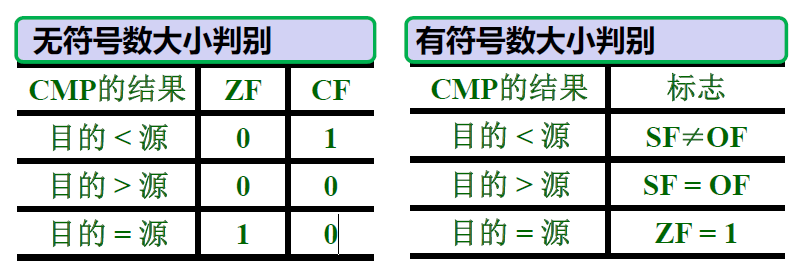


* 单精度下，1位符号，8位指数，23位尾数。
* 双精度下，1位符号，11位指数，52位尾数。
* 指数全为1，尾数为0，表示无穷
* 指数全为1，尾数非0，表示NaN
* 指数为0，尾数为0，表示0
* 指数为0，尾数非0，表示非规格化数，其值为，双精度下-1022
* 指数不是0也不是全1，是规格化数，其值为，双精度下-1023

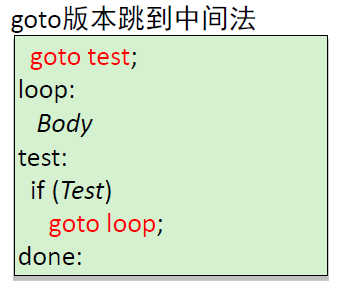
例：写出-10.1在小端模式下，在内存从低地址到高地址的存储字节内容。

# 重要题型2·汇编和C之间的转换填空，汇编填空或读汇编程序

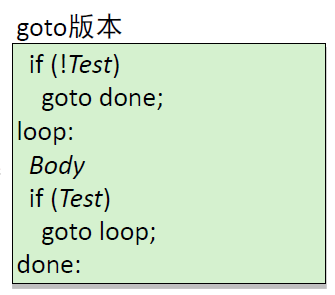
* 常见寄存器：
  + %rax：用于存储返回值
  + %rsp：用于存储栈指针，其值不能被用户覆盖
  + %rbp：栈帧基地址
  + %rdi ,%rsi, %rdx：参数1,2,3
  + %rax, %rdx, %rcx: 局部变量1, 2, 3
* cmp指令与标志寄存器



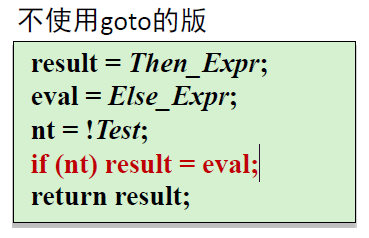
* while的两种翻译方法：
  + 跳到中间翻译法



* + guarded-do翻译法



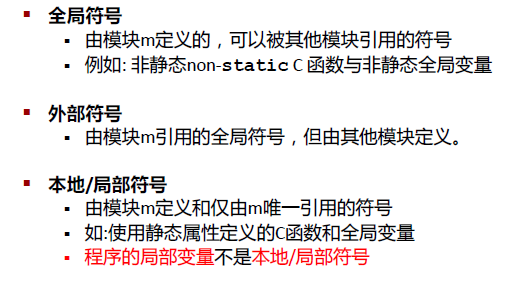
* 条件传送翻译



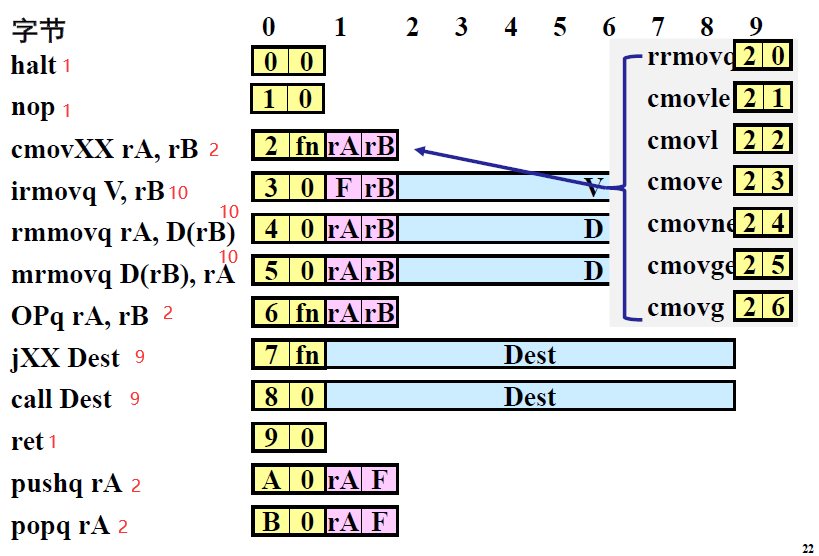
一般情况下三目表达式都使用条件传送翻译而不是分支，因为分支往往对控制流有害。但条件传送在THEN和ELSE表达式很复杂的情况下，或者表达式会影响if结果的情况下，会使得代码逻辑不符合原意。

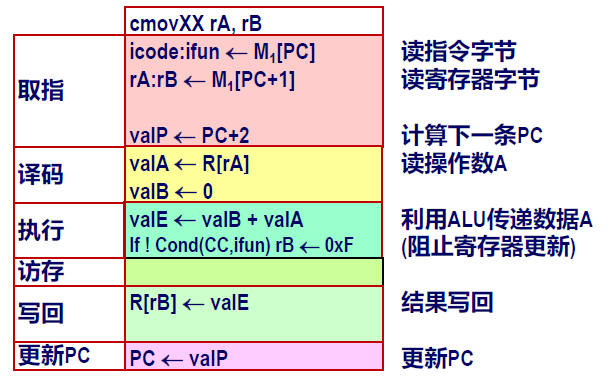
# 重要题型3·链接中的静态重定位

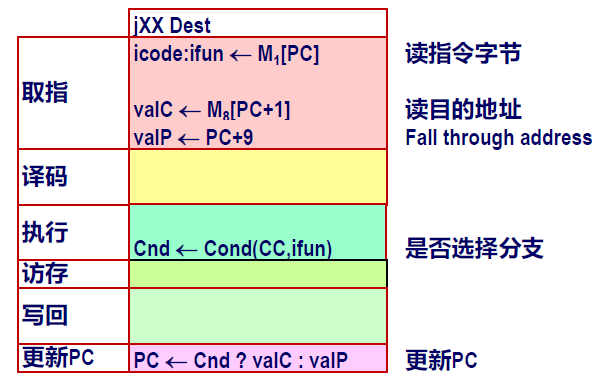
* 重定位条目
  + offset
  + symbol
  + type
  + addend
* 对于type=PC\_32
  + refaddr = ADDR(s) + offset，s是当前指令所在的节，refaddr是当前pc位置
  + \*refptr = (unsigned) (ADDR(symbol) + addend – refaddr)
* 对type=32
  + \*refptr = (unsigned) (ADDR(symbol) + addend)
* 在做重定位时，要多考察汇编代码的上下文用意



# 重要题型4·Y86-64指令格式和其执行的各个阶段









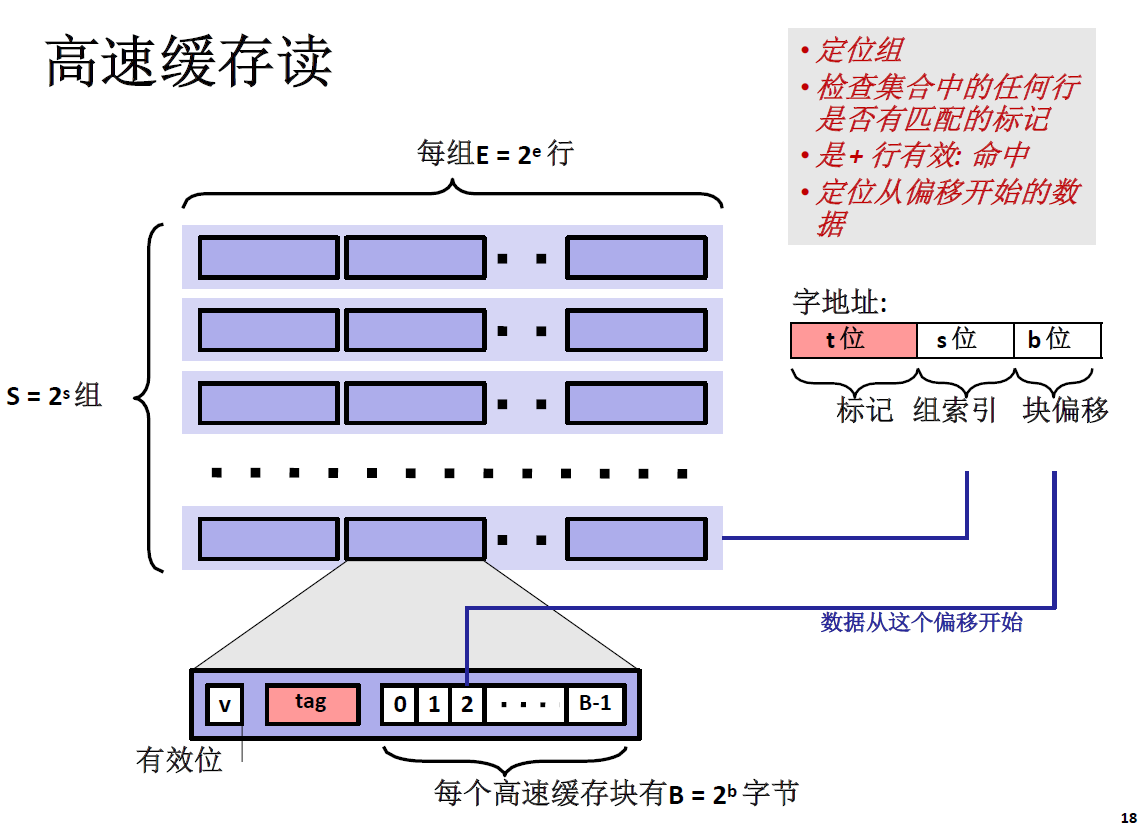
# 重要题型5·程序优化及其依据

列举至少5种常见的程序优化方法。

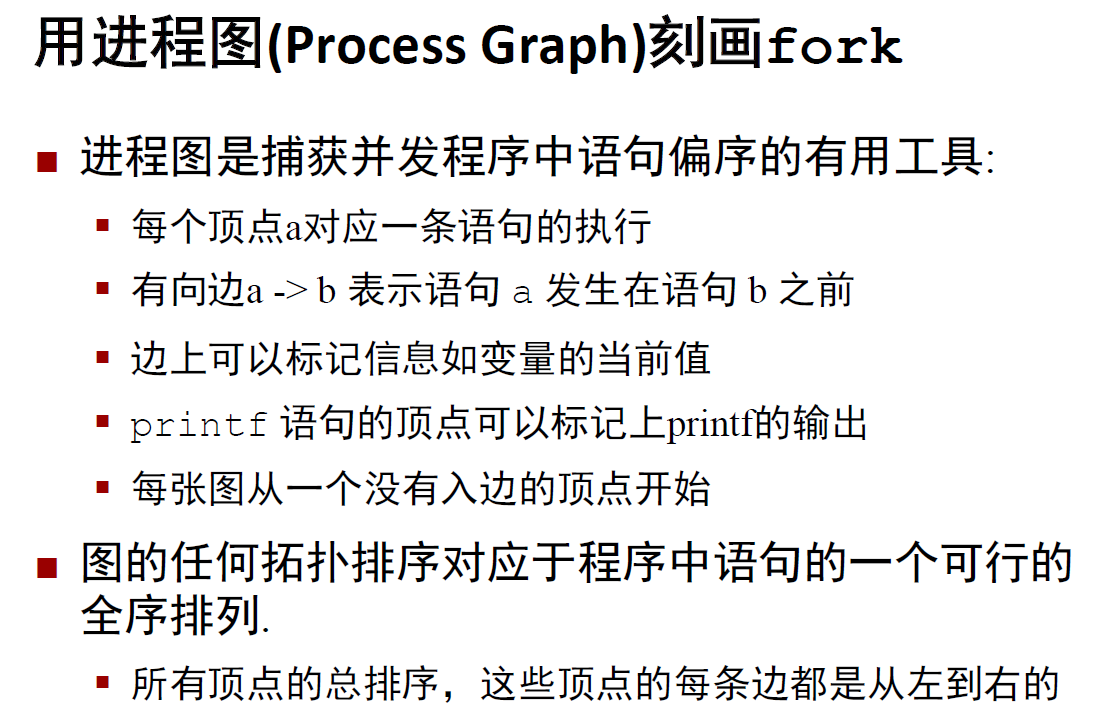
在不改变程序执行逻辑的情况下，

1. 代码移动：将代码从循环中移出以减少计算执行频率
2. 复杂运算简化：用简单计算替代复杂计算，如用移位和加法替换乘法和除法
3. 共享公共子表达式：重用表达式，如使用局部变量做累积量
4. 减少过程调用：用局部变量保存函数结果，小函数用inline的形式声明
5. 消除不必要的内存引用
6. 循环展开

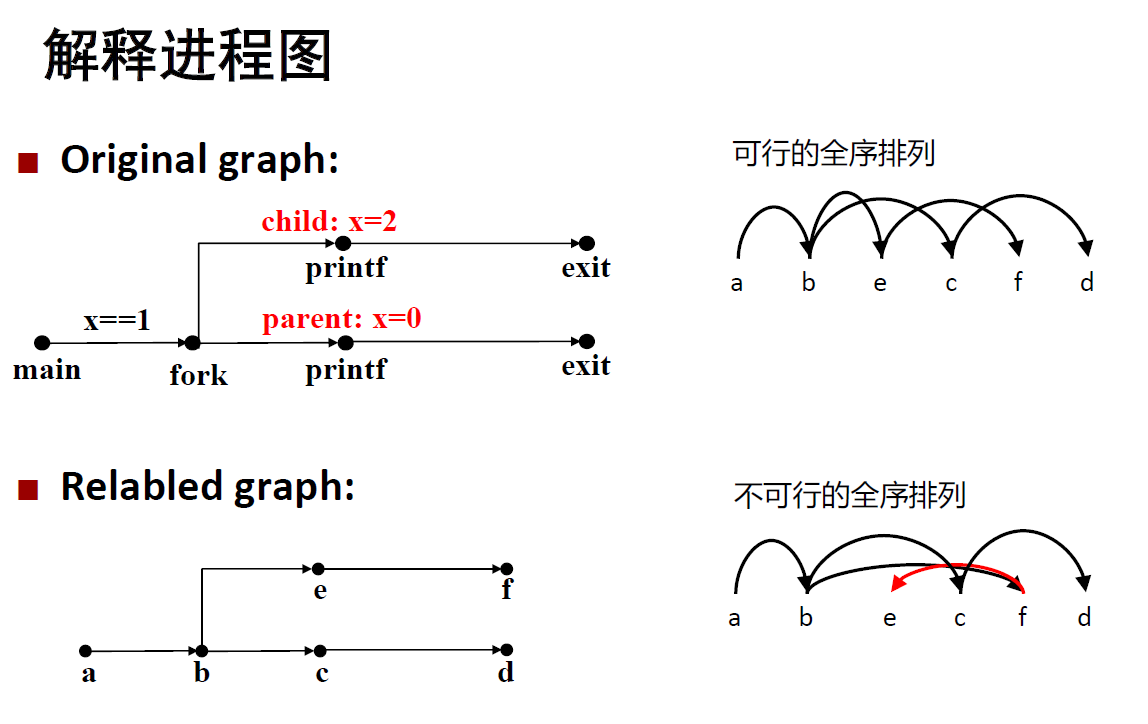
# 重要题型6·高速缓存的相关计算



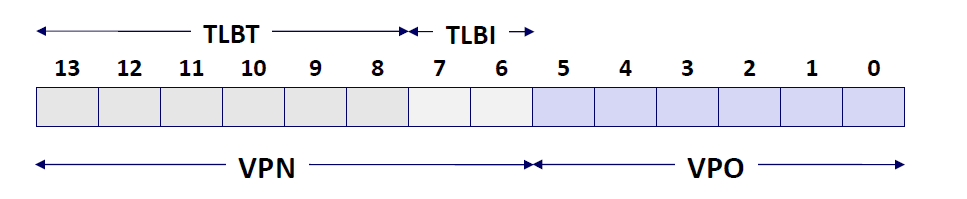
# 重要题型7·进程图

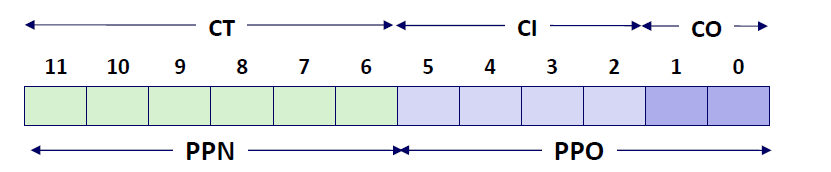


使用进程图和拓扑序列来得到所有可行的全序排列。



# 重要题型8·core i7的地址分割







# 简答题1

简述缓冲区溢出攻击的原理以及防范方法。

攻击原理：

1. 程序输入缓冲区写入特定的数据，如利用gets读入字符串
2. 使位于栈中的缓冲区数据溢出，用特定的内容覆盖栈的内容，例如函数返回地址
3. 使得程序在读入字符串，结束函数gets从栈中读取返回地址时，错误地返回到特定的位置，执行特定的代码，达到攻击的目的

防范方法：

1. 代码中避免溢出漏洞：使用限制字符串长度的库函数
2. 随机栈偏移：程序启动后，在栈中分配随机数量的空间，将移动整个程序使用的栈空间地址
3. 限制可执行代码区域
4. 进行栈破坏检查——栈金丝雀

# 简答题2

Y86-64流水线CPU中冒险的种类与处理方法。

1. 数据冒险：指令使用寄存器R为目的寄存器，然后立刻使用R为源寄存器
   1. 暂停：在执行阶段加入气泡，使得当前指令执行暂停在译码阶段
   2. 数据转发：增加valM和valE的数据旁路，直接送到译码阶段
2. 加载使用冒险：指令暂停在取指和译码阶段，在执行阶段插入1个bubble
3. 控制冒险：
   1. 分支预测错误：在条件为真的地址target处的两条指令分别插入1个bubble，类似于紧急避险
   2. ret：在ret后插入3个bubble，直到访存取到跳到地址后前递给取指阶段

# 简答题3

请解释什么是DMA。并简述CPU发起磁盘读的三条存储指令。

# 简答题4

对于著名的存储器山形状图，分析读吞吐量产生如此变化趋势的原因，说明几条山脊和斜坡出现的原因。

# 简答题5

对于一个地址，高速缓存通常使用地址的中间部分作为组索引，为什么不用高位地址作为组索引？

