

汇编语言程序设计

自然数求和程序



求和过程

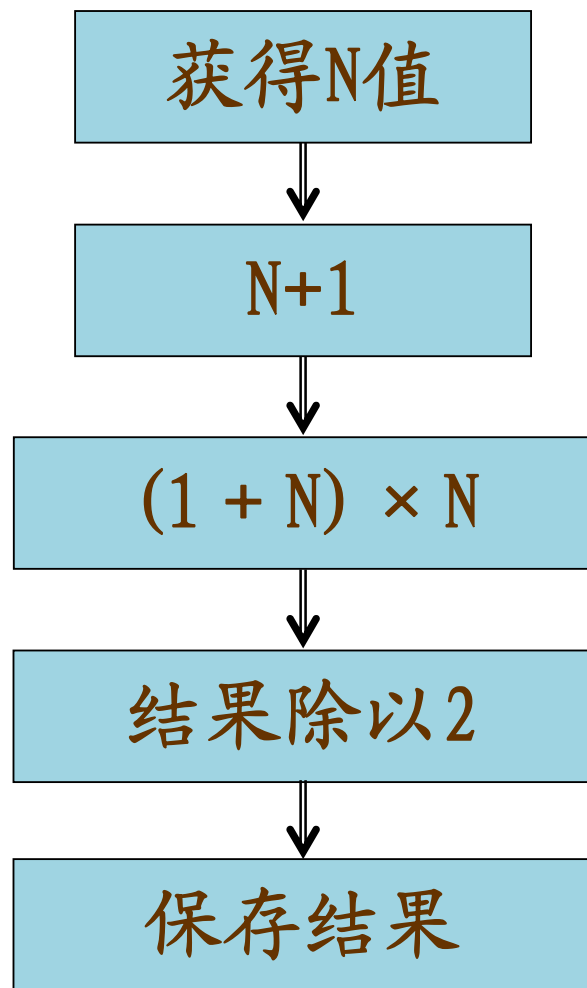
➤ 求N个自然数之和

$$1 + 2 + 3 + \dots + N$$

➤ 利用等差数列求和公式

$$1 + 2 + 3 + \dots + N$$

$$= (1 + N) \times N \div 2$$



变量定义

;数据段

num dword 3456

;假设一个**N**值（小于 **$2^{32}-1$** ）

sum qword ?

;为求和结果预留**64**位变量空间

;因为和值可能达到**64**位

dword

;定义**32**位（双字）变量

qword

;定义**64**位（4字）变量



自然数求和-1

;代码段

$$(1+N) \times N \div 2$$

mov eax,num ;EAX=N

add eax,1 ;EAX=N+1

mul num ;EDX.EAX=(1+N)×N

;32位相乘得到64位乘积

MUL r32/m32

;32位乘法指令: **EDX.EAX = EAX × r32/m32**

保存结果

;**64**位结果，保存在**64**位类型的变量**sum**

;**EDX** = 高**32**位，**EAX** = 低**32**位

;**IA-32**使用小端方式存储在主存

mov dword ptr sum, eax

mov dword ptr sum+4, edx

高对高、低对低



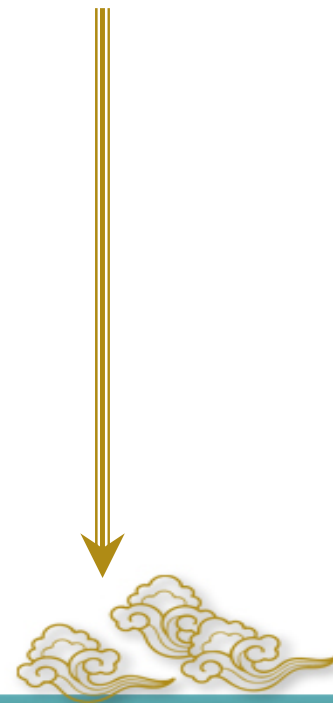
本讲总结

➤ 自然数求和使用求和公式

$$1+2+\cdots+N=(1+N)\times N\div 2$$

➤ 按照指令前后顺序执行每条指令

➤ 形成顺序程序结构



汇编语言程序设计

处理器识别程序



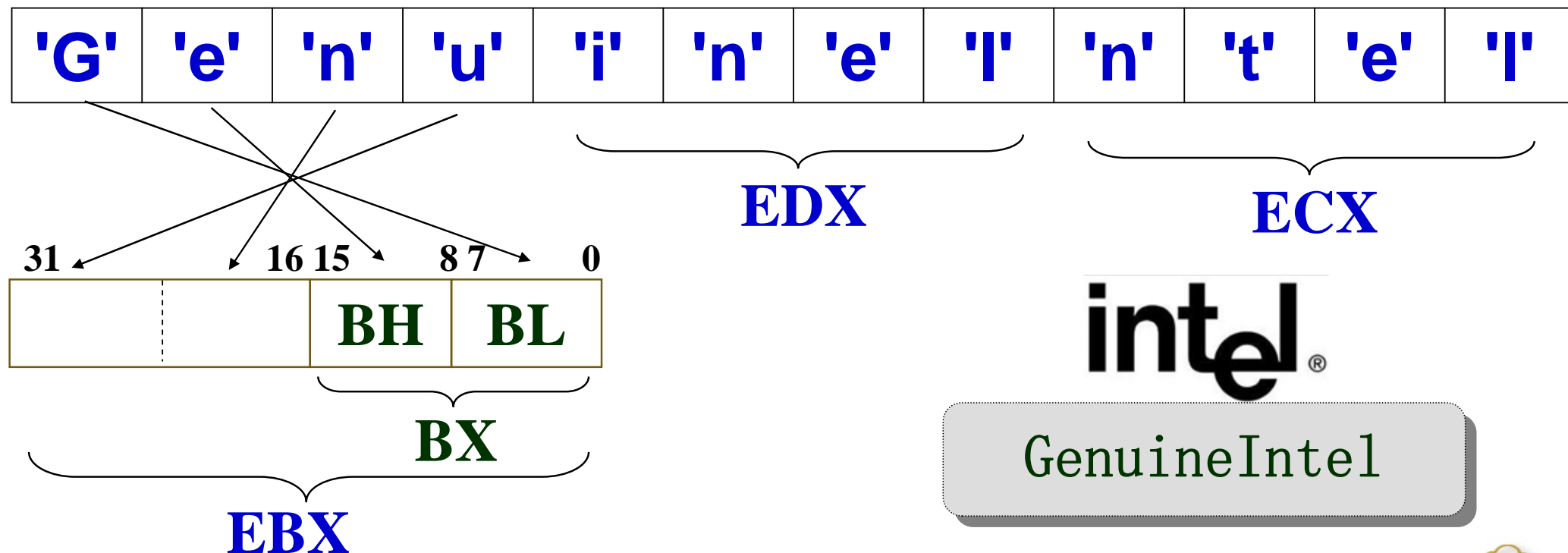
CPUID指令

EAX值	CPUID指令的输出
0	厂商字符串和支持的最大CPUID选项值
1	处理器类型、系列、型号和分步信息
2	处理器缓存配置
3	处理器序列号
4	缓存配置（线程数量、核心数量和物理属性）
5	监视信息
80000000H	扩展的厂商ID字符串和支持的级别
.....



处理器厂商字符串

➤ **12个字符**，以**ASCII**码形式保存于**3个32位寄存器**



intel®

GenuineIntel

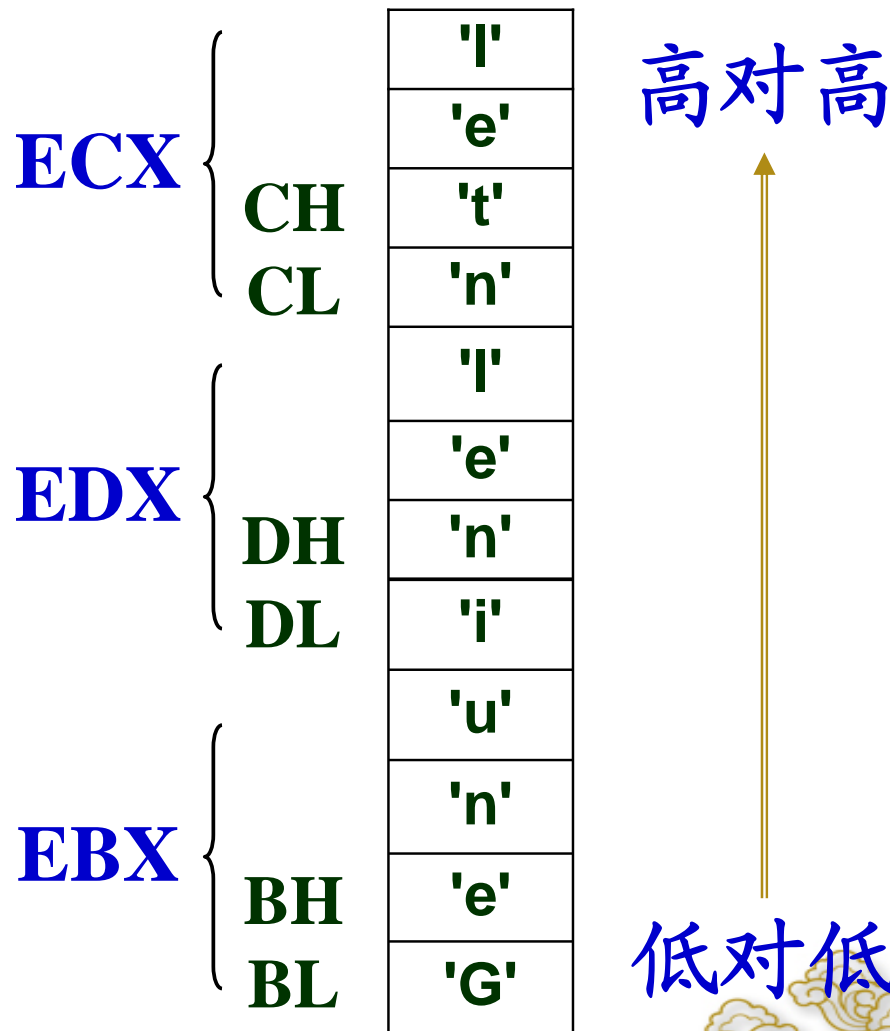


处理器厂商字符串

- 12个字符
- 以ASCII码形式
- 保存于3个32位寄存器

intel®

GenuineIntel



预留存放厂商字符串的空间

;数据段

buffer **byte** 'The processor is', **12 dup(0)**, 0

;预留**12**字节空间，存放厂商字符串

bufsize = sizeof buffer ;获得整个缓冲器长度

intel®

GenuineIntel

AMD

AuthenticAMD



字符串保存于主存

;代码段

mov eax,0

cpuid ;执行处理器识别指令

mov dword ptr buffer+bufsize-13, ebx

bufsize

13

buffer byte 'The processor is', 12 dup(0), 0



处理器厂商字符串显示

```
mov dword ptr buffer+bufsize-13, ebx  
mov dword ptr buffer+bufsize-9, edx  
mov dword ptr buffer+bufsize-5, ecx  
mov eax,offset buffer      ;指向字符串  
call dispmsg               ;显示
```

The processor is GenuineIntel

Intel处理器运行结果



本讲总结

- 按照指令前后顺序执行每条指令
- 形成顺序程序结构

顺序结构——
构成复杂程序结构的基础

