



南开大学

NANKAI UNIVERSITY, P.R. CHINA 1919

允公允能 日新月异



汇编语言与逆向技术

第4章 汇编语言编程基础

王志

zwang@nankai.edu.cn

updated on 2022.10.6

南开大学 网络空间安全学院
2022-2023学年



允公允能 日新月异

本章知识点

- 汇编语言的基本元素
- 数据定义
- 符号常量





南开大学

NANKAI UNIVERSITY, P.R. CHINA 1919

允公允能 日新月异



汇编语言的基本元素

汇编语言的基本元素

- 整数常量、整数表达式
- 实数常量
- 字符常量、字符串常量
- 保留字、标识符
- 指令、伪指令、NOP指令



整数常量

允公允能 日新月异

- $[\{ + | - \}]$ 数字 [基数]
- 基数 后缀 (Radix)
- h十六进制、q/o八进制、d十进制、b二进制
- r编码实数



整数常量

允公允能 日新月异

- 如果整数常量后面没有基数后缀，默认是十进制整数
- 10、10d、10o、10h、10q、0A0h，10b
- 以字母开头的十六进制常量前面必须加0





整数表达式

允公允能 日新月异

- 包含整数值和算数运算符的数学表达式
- 表达式的结果不能超过32bits的表示范围
- MOD: 取余数运算



整数表达式

允公允能 日新月异

- 算术运算符的优先级
- () 优先级1
- *、/、MOD，乘、除、取余，优先级2
- +、-，加减，优先级3

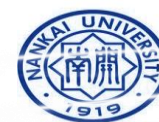




实数常量

允公允能 日新月异

- 十进制实数常量
- 编码（十六进制）实数



十进制实数常量

- -1.11E-5、2.、+3.0、2.E5
- 十进制实数常量由符号sign、整数、小数点、小数和指数组成
- [sign]integer.[integer][exponent]
- 至少要有数字和一个小数点



编码实数

允公允能 日新月异

- 编码实数是以十六进制数表示一个实数，遵循**IEEE浮点数格式**
- 《Intel汇编语言程序设计》第五版，第17章“浮点处理和指令编码”



字符常量

允公允能 日新月异

- 单引号或者双引号括起来的单个字符。
- 汇编器会将其转化为ASCII编码
- ‘A’、 “B”



字符串常量

允公允能 日新月异

- 以单引号或者双引号括起来的一串字符
- ‘ABC’、 “abc”
- 嵌套引号
- “print ‘Hello World’ on the terminal window”
- ‘print “Hello World” on the terminal window’



保留字

允公允能 日新月异

- 指令助记符：MOV、ADD
- 伪指令：INCLUDE、PROC
- 属性：BYTE、WORD
- 预定义符号：\$、?
- 参考《Intel汇编语言程序设计》第五版 附录A



标识符

允公允能 日新月异

- 标识符是程序员选择用来标识变量、常量、过程、代码的标号
 - 包含1~247个字符
 - 大小写不敏感（MASM默认）
 - 第一个字符必须是字母、下划线、@、？或\$
 - 第一个字符不能是数字（对比十六进制整数）



指令

允公允能 日新月异

- 汇编语言中的指令是一条汇编语句
- 汇编器把汇编指令翻译成对应的机器指令
 - 标号
 - 指令助记符
 - 操作数
 - 注释

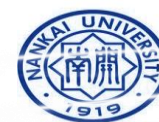




标号

允公允能 日新月异

- 标号是充当指令或数据位置标记的标识符
- 数据标号
 - 标识变量的地址
- 代码标号
 - 标识代码的地址



数据标号

允公允能 日新月异

- 标识变量的地址，方便变量的引用
- **count** DWORD 100
- **array** DWORD 100, 101, 102, 103
- 相对.data数据段在内存起始地址的偏移





OFFSET

允公允能 日新月异

- 获取数据标号的内存偏移地址

.data

str_hello BYTE "Hello World! ", 0

.code

mov eax, **OFFSET** str_hello

operator OFFSET

Article • 08/04/2021 • 2 minutes to read • [6 contributors](#)

Returns the offset into the relevant segment of *expression*.



南开大学
Nankai University

代码标号

允公允能 日新月异

- 标识代码的地址
- 必须以冒号 (:) 结尾
- 通常作为跳转、循环指令的目标地址

target:

```
mov eax, 100h
```

```
...
```

```
jmp target
```





指令助记符

允公允能 日新月异

- 指令助记符（instruction mnemonic）是一个简短的单词，用于表示一条指令。
 - mov、add、sub、mul、jmp、call



操作数

允公允能 日新月异

- 操作数是指令的操作对象
 - 寄存器
 - 内存
 - 常量
 - I/O端口



操作数

允公允能 日新月异

- `inc eax`
 - `eax`寄存器的值加1
- `mov count, ebx`
 - `mov`指令有两个操作数：`count`、`ebx`
 - 第一个操作数是目的操作数
 - 第二个操作数是源操作数



注释

允公允能 日新月异

- 单行注释
 - `mov count, ebx; save result to count`
- 块注释: COMMENT伪指令和用户定义的符号

COMMENT !

This is a comment

!



NOP指令

允公允能 日新月异

- NOP指令，空操作
 - 用于计时循环
- NOP指令占用1个字节的内存
 - 用于后继指令的对齐
 - IA-32处理器从偶数双字地址处加载代码和数据时更加快速





伪指令

允公允能 日新月异

- 伪指令内嵌在汇编语言源代码中，由汇编器识别、执行相应动作的命令
- 用于定义变量、段、过程、汇编器选项等
- 参考《Intel汇编语言程序设计》第五版，附录A，MASM的伪指令



伪指令

允公允能 日新月异

- 定义变量

my_var DWORD 100h; DWORD伪指令

mov eax, my_var ; mov指令





伪指令

允公允能 日新月异

- 定义段（Segment）
 - .data、.code、.stack
- 定义过程（Procedure）
 - PROC、ENDP
- 允许或禁止汇编器的某些特性
 - OPTION、.386、.MODEL





南开大学

NANKAI UNIVERSITY, P.R. CHINA 1919

允公允能 日新月异



定义数据

数据类型

允公允能 日新月异

- MASM以数据所占内存字节数定义了多种数据类型
 - BYTE, db, 8位
 - WORD, dw, 16位
 - DWORD, dd, 32位
 - QWORD, dq, 64





数据类型

允公允能 日新月异

- MASM汇编器默认情况下，大小写不敏感
- DWORD
 - Dword
 - dword
 - dWord



数据定义

允公允能 日新月异

- 为变量在内存中保留存储空间
- 为变量指定一个名字（数据标号）
- [变量名] 数据定义伪指令 初始值



数据定义伪指令

- BYTE, db, 8 bits
- WORD, dw, 16 bits
- DWORD, dd, 32 bits
- QWORD, dq, 64 bits



初始值

- 数据定义语句中要指定初始值
- 多个初始值用逗号隔开
 - `my_var DWORD 0, 1, 2, 3`
- 0: 可以指定初始值为0
- **?:** 表示在程序运行的时候初始化该变量

数据声明的位置

- .data段声明初始化的变量

.data

dw_var1 DWORD 0

- .data?段声明未初始化的变量

.data?

dw_var2 DWORD ?



定义字符串

允公允能 日新月异

```
str_hello BYTE "Hello World!", 0Dh, 0Ah,  
            BYTE "I love assembly language",  
            BYTE 0Dh, 0Ah, 0
```

- 0Dh和0Ah是CR/LF（回车、换行）的ASCII编码
- 字符串的结尾是0



DUP伪指令

允公允能 日新月异

- 为字符串或者数组分配内存空间
- BYTE 20 DUP (0) ; 20个字节的内存空间
- BYTE 4 DUP (“Hello”) : 20个字节，连续的4个 “Hello” ，
每个 “Hello” 5字节





南开大学

NANKAI UNIVERSITY, P.R. CHINA 1919

允公允能 日新月异



符号常量

符号常量

允公允能 日新月异

- 符号常量（或符号定义），将标识符与整数表达式或文本联系起来
- 符号常量不占用存储空间
- 变量占用存储空间



等号伪指令

允公允能 日新月异

- 等号伪指令，将符号名和整数表达式联系起来

```
COUNT = 500
```

```
mov eax, COUNT
```

- 易于阅读与维护
- 减少程序修改时的查找与替换次数

=

Article • 08/04/2021 • 2 minutes to read • [6 contributors](#)

Assigns the numeric value of *expression* to *name*.

Syntax

name = *expression*



南开大学
Nankai University

计算数组和字符串的大小

- MASM用\$运算符存储当前语句的地址偏移。
- \$可以用来计算数组或字符串的大小



计算字符串大小

str_hello BYTE “Hello World!”, 0Dh, 0Ah,
BYTE “I love assembly language”,
BYTE 0Dh, 0Ah, 0

- $\text{str_size} = (\$ - \text{str_hello})$



计算数组大小

dw_array DWORD 0, 1, 2, 3, 4

array_size = (\$ - dw_array)/4





EQU伪指令

允公允能 日新月异

- EQU伪指令将符号名与整数表达式或任意文本联系起来
 - name EQU expression
 - name EQU symbol
 - name EQU `<text>`



EQU 伪指令

允公允能 日新月异

PI EQU <3.1415926>

press_key EQU <“Press any key to continue...”, 0>

.data

prompt BYTE pressKey ; 变量





允公允能 日新月异

EQU伪指令

- EQU伪指令不能在程序中重定义
- “=” 伪指令可以在程序中重定义





南開大學

NANKAI UNIVERSITY, P.R. CHINA 1919

允公允能 日新月異



实验2: dex2hex



实验目的

允公允能 日新月异

- 熟悉汇编语言的数据传送、寻址和算术运算；
- 熟悉汇编语言过程的定义和使用；
- 熟悉十进制和十六进制的数制转换





实验环境

允公允能 日新月异

- MASM32编译环境
- Windows命令行窗口



实验内容

允公允能 日新月异

- 编写汇编程序dec2hex.asm，编译成dec2hex.exe。
- dec2hex.exe能够将Windows命令行输入的十进制无符号整数，转换成对应的十六进制整数，输出在Windows命令行中

```
D:\>dec2hex.exe
Please input a decimal number( 0~ 4294967295): 100

The hexadecimal number is : 00000064
```




StdIn函数

允公允能 日新月异

- 获得用户输入的十进制整数。
- 定义在\masm32\include\masm32.inc
- 库文件是\masm32\lib\masm32.lib。
- StdIn函数的定义 “StdIn PROTO :DWORD,:DWORD”
 - 内存存储空间的起始地址
 - 内存存储空间的大小。





StdIn函数

允公允能 日新月异

- 用户输入的十进制数对应的ASCII编码字符串存储在内存中
- 输入：100
- 内存存储：49h, 48h, 48h, 00





南开大学

NANKAI UNIVERSITY, P.R. CHINA 1919

允公允能 日新月异



汇编语言与逆向技术

第4章 汇编语言基础

王志

zwang@nankai.edu.cn

updated on 2022.10.6

南开大学 网络空间安全学院
2022-2023学年



本章学习的知识点

- 汇编语言的基本元素
- 定义数据
- 符号常量

