汇编语言

01.汇编语言简介 02.进制 03.数据寄存器 04.指针寄存器 05.变址寄存器 06.指令指针寄存器 07.标志寄存器 08.段寄存器 09.数据传送指令 10.加减运算指令 11.逻辑运算 12.移位指令 13.test,cmp指令 14.push,pop指令 15.jmp，nop指令 16.jcc指令 17.call,retn指令 18.总结

- 学习汇编语言用处：游戏外挂与反外挂，游戏安全类，软件破解，软件暴力破解。。。

01.汇编语言简介

汇编语言的由来：方便我们阅读和记忆机器指令（硬编码）

```
操作：寄存器ebx的内容移动到eax中
机器指令：1000100111011000
汇编指令：mov eax,ebx
```

机器语言（硬编码）：由0和1组成

寄存器：cpu上的一个元件，它可以暂时保存数据（0,1）

02.进制

十进制：0 1 2 3 4 5 6 7 8 9

10 11 12 13...19 20

二进制：0 1

10 11 100 //10==2 11==3 100==4

十六进制：0 1 2 3 4 5 6 7 8 9 a b c d e f

10 11 12 //10==16 12==18

一般反汇编引擎（OD）都是以十六进制的形式表达二进制。

03.数据寄存器

寄存器：cpu上的一个元件（部件），读写速度非常快。

数据寄存器：保存操作数，计算结果。

EAX (Accumulator)：累加寄存器，也称之为累加器； EBX (Base)：基地址寄存器； ECX (Count)：计数器寄存器； EDX (Data)：数据寄存器；

32位寄存器:EAX,EBX,ECX,EDX

16位寄存器：AX,BX,CX,DX

低八位：al

高八位：ah

04.指针寄存器

指针寄存器:操作栈的寄存器

栈：参数，变量

EBP：栈底的指针

ESP：栈顶的指针

05.变址寄存器

esi和edi ： 用来存放一个地址的寄存器。

06.指令指针寄存器

eip:cpu下一次将要执行的代码的地址。

07.标志寄存器

标志寄存器：flag寄存器 16位

置为1

08.段寄存器

段寄存器是因为对内存的分段管理而设置的。计算机需要对内存分段，以分配给不同的程序使用。

mov dword ptr ds:[0x405528],edx

ds.base+0x405528

09.数据传送指令

mov ax,bx

10.加减运算指令

add

sub

add eax,8 //int a=8; a= 8+0;

11.逻辑运算

逻辑与：and

同为1就为1，只要一个不为1，就不为1。

```
mov eax,1;
and eax,2;
01
10
00
```

逻辑或：or

只要一个为1，就是1。

```
mov eax,1;
or eax,2;
01
10
11
```

逻辑异或：xor

同为0，异为1。

```
mov eax,1;
xor eax,2;
01
10
11
```

逻辑非：not

```
mov eax,3;
not eax;
11
00
```

12.移位指令

算术移位指令：

```
// 算术左移
mov eax,2  //10（二进制） == 2(十进制)
sal eax,1   //算术左移与逻辑左移  功能是一样的
00000010
00000100   //100(二进制) == 4
// 算术右移
mov eax,2  //10（二进制） == 2(十进制)
sar eax,1   //
00000010
00000001
```

逻辑移位指令:

```
// 逻辑右移
mov eax,2  //10（二进制） == 2(十进制)
shr eax,1   //
00000010
00000001
```

13.test,cmp指令

test指令：实际就是作逻辑与运算

与逻辑与(and)不同点是：test指令不会改变值，只会改变z标志位的值。

```
and eax,1

test eax,1
```

cmp指令:实际上作的是减法运算

与算术运算（sub）不同点是：cmp指令也是不会改变值，只会影响到z标志位。

```
sub eax,1
cmp eax,1
```

14.push,pop指令

push 压栈指令

pop 出栈指令

栈：先进后出，后进先出

```
push指令：
    //push ebp
    01.提升栈顶（esp-4）
    02.把ebp里面的内容压到栈（esp-4）

pop指令：
    //pop ebp
    01.把栈顶里面值取出来放到ebp
    02.恢复栈原来的样子（esp+4）
```

15.jmp，nop指令

jmp指令：无条件跳转指令

nop指令：空指令（cpu执行到这个命令什么也不干）

16.jcc指令

JCC指条件跳转指令，CC就是指条件码。

JZ 如果表达式计算的结果等于0，那么zf标志位会置为1，则jz指令跳转。

17.call,retn指令

call：

①jmp + 函数地址

②push call函数指令的下一行指令的地址

retn: