

汇编语言与逆向技术实验报告

Lab2-dec2hex

学号：2112060 姓名：孙蓓 专业：信息安全

一、实验目的

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

二、实验环境

MASM32 编译环境

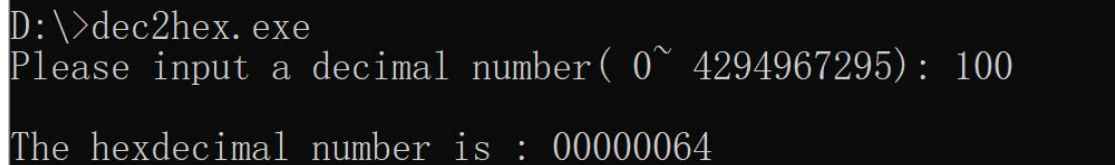
Windows 命令行窗口

三、实验内容

编写汇编程序 `dec2hex.asm`，编译成 `dec2hex.exe`。`dec2hex.exe` 的功能是将 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，如图 1 所示。

输入的十进制无符号整数的范围是 **0 到 4294967295** ($2^{32}-1$)。

输出对应的十六进制整数，对应的范围是 **00000000h 到 FFFFFFFFh**。



```
D:\>dec2hex.exe
Please input a decimal number( 0~ 4294967295): 100
The hexadecimal number is : 00000064
```

图 1. `dec2hex.exe` 将十进制 100 转换成十六进制 00000064

3.1 使用 `StdIn` 函数获得用户输入的十进制整数。`StdIn` 函数的定义在 `\masm32\include\masm32.inc`，库文件是 `\masm32\lib\masm32.lib`。`StdIn` 函数的定义 “`StdIn PROTO :DWORD, :DWORD`”，有两个参数，第一个是内存存储空间的起始地址，第二个是内存存储空间的大小。函数的例子：

```
.data
buf BYTE 20 DUP(0)
.code
invoke StdIn, addr buf, 20
invoke StdOut, addr buf
```

3.2 用户输入的十进制数对应的 ASCII 编码字符串存储在内存中，编写过程 dec2dw，将 ASCII 字符串转换成 DWORD 数据。例如，将字符串“100”转换成 DWORD 数据 00000064h。

3.3 编写过程 dw2hex，将 DWORD 数据转换成十六进制数的 ASCII 字符串。例如，将 DWORD 数据 00000064h 转换成 ASCII 字符串“00000064”

3.4 使用 StdOut 函数在 Windows 命令函中输出十六进制整数的 ASCII 字符串。StdOut 函数的定义在\masm32\include\masm32.inc，库文件是\masm32\lib\masm32.lib。StdOut 函数的定义“StdOut PROTO :DWORD”，只有一个参数，是内存存储空间的起始地址。函数使用的例子同 StdIn 函数的例子。

3.5 使用 ml 将 dec2hex.asm 文件汇编到 dec2hex.obj 目标文件，编译命令：

“\masm32\bin\ml /c /coff dec2hex.asm”

3.6 使用 link 将目标文件 dec2hex.obj 链接成 dec2hex.exe 可执行文件，链接命令：“\masm32\bin\link /SUBSYSTEM: CONSOLE dec2hex.obj”

四. 源代码

.386

.MODEL flat, stdcall

OPTION casemap :none

INCLUDE \masm32\include\windows.inc

INCLUDE \masm32\include\kernel32.inc

INCLUDE \masm32\include\masm32.inc

INCLUDELIB \masm32\lib\kernel32.lib

INCLUDELIB \masm32\lib\masm32.lib

.data

str1 BYTE "Please input a decimal number(0~4294967295):",

0 ;输入提示

str2 BYTE "The hexadecimal number is:", 0 ;

输出提示

mmc BYTE "0123456789ABCDEF", 0

dec_num BYTE 10 DUP(0), 0 ;存十进制字符串

```

    dw_num DWORD 0                                ;十进制数字
    dec_cnt DWORD 0
    hex_cnt DWORD 0
    v16 BYTE 10h                                  ;数 16
    v10 DWORD 0Ah                                  ;数 10
    power DWORD 1
    one BYTE 0, 0
    oneAH BYTE 0
    oneECX DWORD 0
    oneESI PDWORD 0
.code
main PROC
    INVOKE StdOut, ADDR str1                        ;输出输入提示字符串
    INVOKE StdIn, ADDR dec_num, 10                  ;输入十进制字符串

    call dec2dw;                                    ;调过程 dec2dw
    INVOKE StdOut, ADDR str2;                        ;输出输出提示字符串

    call Dw2hex;                                    ;调过程 dw2hex
    INVOKE ExitProcess, 0                            ;结束

main ENDP
dec2dw PROC
    mov esi, OFFSET dec_num;                        ;把变量 dec_num 的偏移
地址放到寄存器 esi 中
L1:
    inc dec_cnt                                      ;+1
    inc esi                                          ;+1
    mov eax, [esi]
    cmp al, 0                                        ;看有没有到字符串尾
    je L2                                           ;有就跳出

```

```

        jmp L1
L2:
        mov ecx, dec_cnt                ;记录几个字符数字
L3:
        sub dec_num[ecx-1], '0'        ;处理成数字

        mov eax, 0                      ;清零

        mov al, BYTE ptr dec_num[ECX-1] ;从头开始
        mul power

        ADD dw_num, eax

        mov eax, power

        mul v10                        ;移位乘十加个位

        mov power, eax
        loop L3
        RET

dec2dw ENDP
Dw2hex PROC
        mov esi, OFFSET dw_num+3
        mov ecx, 4
L4:
        mov ax, 0
        mov al, BYTE ptr [ESI]
        div v16                        ;除 16
        mov oneAH, ah
        mov oneECX, ecx                ;复制赋值

```

xchg esi, oneESI ;交换内容

;0-9 转字符

mov esi, OFFSET mmc

mov one, al

movzx ebx, one ;movzx 将源操作数取出来置于目

的操作数,目的操作数其余位用 0 填充

add esi, ebx

mov bl, BYTE ptr [ESI]

mov one, bl

INVOKE StdOut, ADDR one

;10-15 转字符

mov esi, OFFSET mmc

mov ah, oneAH

mov one, ah

movzx ebx, one

add esi, ebx

mov bl, BYTE ptr [ESI]

mov one, bl ;字符

INVOKE StdOut, ADDR one

xchg oneESI, esi ;交换内容

mov ecx, oneECX ;个数减少

dec esi

LOOP L4

RET

Dw2hex ENDP

END main

五. 实验步骤

1. 源文件：用文本编辑器编写的 asm 文本文件

```
dec2hex.asm - 记事本

文件 编辑 查看

.386
.MODEL flat,stdcall
OPTION casemap:none
INCLUDE \masm32\include\windows.inc
INCLUDE \masm32\include\kernel32.inc
INCLUDE \masm32\include\user32.inc
INCLUDE \masm32\include\shell32.lib
INCLUDE \masm32\include\ole32.lib

.data
str1 BYTE "Please input a decimal number(0-4294967295):",0 ;输入提示
str2 BYTE "The hexadecimal number is:",0 ;输出提示
mimo BYTE "123456789ABCDEF",0 ;输入十进制字符串
dec_num BYTE 10 DUP(0),0 ;十进制数字
div_num DWORD 0
dec_ent DWORD 0
hex_ent DWORD 0
v16 BYTE 10h ;数16
v10 DWORD 0Ah ;数10
power DWORD 1
one BYTE 0,0
oneAH BYTE 0
oneECX DWORD 0
oneESI DWORD 0

.code
main PROC
    INVOKE StdOut, ADDR str1 ;输出输入提示字符串
    INVOKE StdIn, ADDR dec_num, 10 ;输入十进制字符串
    CALL dec2div ;通过dec2div
    INVOKE StdOut, ADDR str2 ;输出输出提示字符串
    CALL Div2hex ;通过div2hex
    INVOKE StdProcess, 0 ;退出

main ENDP
dec2div PROC
    MOV ESI, OFFSET dec_num ;把dec_num的偏移地址放到寄存器esi中

L1:
    INC dec_ent ;位数+
    INC ESI ;
    MOV EAX, [ESI]
    CMP AL, 0 ;看有没有到字符串尾
    JE L2 ;有就跳出
    JMP L1

L2:
    mov ecx, dec_ent ;记录位数是几个数字
    L3:
        sub dec_num[ecx*1], 0 ;左端的数字
        mov ecx, 0 ;清零
        mov al, BYTE ptr dec_num[ECX*1] ;从开头
        mul power ;加个位
        ADD div_num, ecx ;
        mov ecx, power ;乘十
        mul v10 ;
        mov power, ecx ;
        loop L3
    RET

dec2div ENDP
Div2hex PROC
    mov esi, OFFSET div_num+3
    mov ecx, 4

L4:
    mov ecx, 0 ;清零
    mov al, BYTE ptr [ESI] ;取16
    div v16 ;除16
    mov oneAH, ah ;
    mov oneECX, ecx ;取制除数
    xchg esi, oneESI ;交换内容

    ;0-9的字符串
    mov esi, OFFSET mimo
    mov one, si
    movzx ebx, one
    movzx esi, ebx
    add esi, one
    mov si, BYTE ptr [ESI]
    mov one, si
    INVOKE StdOut, ADDR one

    ;10-15的字符串
    mov esi, OFFSET mimo
    mov ah, oneAH
    mov one, ah
    movzx ebx, one
    add esi, ebx
    mov al, BYTE ptr [ESI]
```

2. 汇编：用汇编程序（\masm32\bin\ml.exe）对源程序进行汇编，形成目标文件（.obj），格式如下：

“\masm32\bin\ml /c /Zd /coff dec2hex.asm”

```
选择 命令提示符
Microsoft Windows [版本 10.0.22621.674]
(c) Microsoft Corporation. 保留所有权利。

C:\Users\LENOVO>d:

D:\>cd A 汇编
'cd A' 不是内部或外部命令，也不是可运行的程序
或批处理文件。

D:\>cd A 汇编

D:\A 汇编>cd 第二次实验

D:\A 汇编\第二次实验>cd 2.13final
系统找不到指定的路径。

D:\A 汇编\第二次实验>cd 2.13 final

D:\A 汇编\第二次实验\2.13 final>\masm32\bin\ml /c /Zd /coff dec2hex.asm
Microsoft (R) Macro Assembler Version 6.14.8444
Copyright (C) Microsoft Corp 1981-1997. All rights reserved.

Assembling: dec2hex.asm

*****
ASCII build
*****

D:\A 汇编\第二次实验\2.13 final>
```

3. 连接：用连接程序（\masm32\bin\link.exe）对目标程序进行连接，形成可执行文件（.exe），格式如下

“\masm32\bin\Link /SUBSYSTEM:CONSOLE hello_console.obj”

```
命令提示符
Microsoft Windows [版本 10.0.22621.674]
(c) Microsoft Corporation. 保留所有权利。

C:\Users\LENOVO>d:

D:\>cd A 汇编
'cd A' 不是内部或外部命令，也不是可运行的程序
或批处理文件。

D:\>cd A 汇编

D:\A 汇编>cd 第二次实验

D:\A 汇编\第二次实验>cd 2.13final
系统找不到指定的路径。

D:\A 汇编\第二次实验>cd 2.13 final

D:\A 汇编\第二次实验\2.13 final>\masm32\bin\ml /c /Zd /coff dec2hex.asm
Microsoft (R) Macro Assembler Version 6.14.8444
Copyright (C) Microsoft Corp 1981-1997. All rights reserved.

Assembling: dec2hex.asm

*****
ASCII build
*****

D:\A 汇编\第二次实验\2.13 final>\masm32\bin\Link /SUBSYSTEM:CONSOLE dec2hex.obj
Microsoft (R) Incremental Linker Version 5.12.8078
Copyright (C) Microsoft Corp 1992-1998. All rights reserved.
```

 dec2hex.asm	2022/10/14 19:17	ASM 文件
 dec2hex.exe	2022/10/14 19:30	应用程序
 dec2hex.obj	2022/10/14 19:29	3D Object

六. 测试说明

```
命令提示符
D:\A 汇编\第二次实验\2.13 final>dec2hex
Please input a decimal number(0~4294967295):100
The hexadecimal number is:00000064
D:\A 汇编\第二次实验\2.13 final>
```