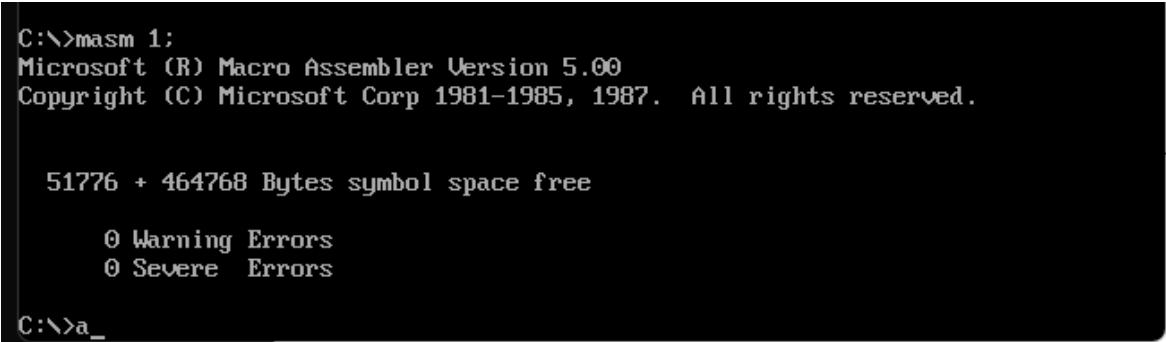


山东大学 计算机科学与技术 学院

汇编语言 课程实验报告

学号：202200130053	姓名：陈红瑞	班级：3 班
实验题目：实验 1：例 1.1		
实验学时：2	实验日期：20241014	
实验目的：学习使用和熟悉 MASM、LINK、DEBUG、EDIT、TD 等汇编工具。掌握一般汇编语言程序的编程框架。学习汇编程序的基本编写习惯，包括但不限于寄存器使用规范、变量/标号命名、注释、对齐、分段、缩进等。		
实验环境：Windows10、DOSBox-0.74、Masm64		
源程序清单： 1. 1.asm （示例 1.1）		
编译及运行结果： 编译结果如下图： 		

```

C:\>link 1:

Microsoft (R) Overlay Linker Version 3.60
Copyright (C) Microsoft Corp 1983-1987. All rights reserved.

LINK : warning L4021: no stack segment

C:\>_

```

运行结果如下图：

```

C:\>1
Match.

C:\>a

```

下面再对程序进行调试：

使用 g 指令来运行，并且可以设置断点的位置。如，g0b 将断点设置在运行部分代码之前，同时可以查看寄存器的内容。

```

-g0b

AX=076A BX=0000 CX=007A DX=0000 SP=FFFC BP=0000 SI=0000 DI=0000
DS=076A ES=076A SS=0769 CS=076F IP=000B  NU UP EI PL ZR NA PE NC
076F:000B 8D360000 LEA SI,[0000] DS:0000=6F4D
- ▲

```

使用 r 指令可以查看和修改寄存器，如在下图中实现了查看和修改 ax 寄存器的内容。

```

-r
AX=076A BX=0000 CX=007A DX=0000 SP=FFFC BP=0000 SI=0000 DI=0000
DS=076A ES=076A SS=0769 CS=076F IP=000B  NU UP EI PL ZR NA PE NC
076F:000B 8D360000 LEA SI,[0000] DS:0000=6F4D
-rax
AX 076A
:0
-a

```

使用 d 指令可以查看内存单元的内容，默认为数据段。如图，使用 d0 指令来查看数据段中从 0 地址开始的内容，当程序的主要部分没有运行时，可以找到初始设置的字符串的内容。

```

-d0
076A:0000 4D 6F 76 65 20 74 68 65-20 63 75 72 73 6F 72 20 Move the cursor
076A:0010 62 61 63 6B 77 61 72 64-2E 4D 6F 76 65 20 74 68 backward.Move th
076A:0020 65 20 63 75 72 73 6F 72-20 62 61 63 6B 77 61 72 e cursor backwar
076A:0030 64 2E 4D 61 74 63 68 2E-0D 0A 24 4E 6F 20 6D 61 d.Match...$No ma
076A:0040 74 63 68 21 0D 0A 24 00-00 00 00 00 00 00 00 00 tch!...$.....
076A:0050 1E 2B C0 50 B8 6A 07 8E-D8 8E C0 8D 36 00 00 8D .+.P.j.....6...
076A:0060 3E 19 00 FC B9 19 00 F3-A6 74 06 8D 16 3B 00 EB >.....t...;...
076A:0070 04 8D 16 32 00 B4 09 CD-21 CB 73 83 C4 06 8B B6 ...2....!.s....

```

下面对这些内存中的内容进行修改，使用 e 指令。如将第二个字符串的首字母从 'M' 改为 'm'，再次查看时，以及成功将对应的部分修改了。

```

-e19 'm'
-d0
076A:0000 4D 6F 76 65 20 74 68 65-20 63 75 72 73 6F 72 20 Move the cursor
076A:0010 62 61 63 6B 77 61 72 64-2E 6D 6F 76 65 20 74 68 backward.move th
076A:0020 65 20 63 75 72 73 6F 72-20 62 61 63 6B 77 61 72 e cursor backwar
076A:0030 64 2E 4D 61 74 63 68 2E-0D 0A 24 4E 6F 20 6D 61 d.Match...$No ma
076A:0040 74 63 68 21 0D 0A 24 00-00 00 00 00 00 00 00 00 tch!...$.....
076A:0050 1E 2B C0 50 B8 6A 07 8E-D8 8E C0 8D 36 00 00 8D .+.P.j.....6...
076A:0060 3E 19 00 FC B9 19 00 F3-A6 74 06 8D 16 3B 00 EB >.....t...;...
076A:0070 04 8D 16 32 00 B4 09 CD-21 CB 73 83 C4 06 8B B6 ...2....!.s....

```

下面对程序进程单步调试，使用 t 指令。每次执行后，可以看到当前这一步执行的指令并显示寄存器的内容。

```

-t
AX=076A BX=0000 CX=0019 DX=0000 SP=FFFC BP=0000 SI=0000 DI=0019
DS=076A ES=076A SS=0769 CS=076F IP=0017  NU UP EI PL ZR NA PE NC
076F:0017 F3          REPZ
076F:0018 A6          CMPSB

```

最后，在修改了内存以后，输出的结果如图。

```

-g
No match!

Program terminated normally

```

代码中使用了 INT 21H 中的 09H 的中断操作，在计算完匹配结果后，根据结果选择将 mess1 或者 mess2 的有效地址加载到 dx 中，并在加载完成后均执行 disp 部分的代码段，将 09H 传入 AX 中，并调用 INT 21 指令，此时根据 INT 的操作数

来计算 CS 和 IP 的值，根据这两个值在系统中找到对应中断向量的类型，并在对应的中断向量中根据 AX 的值找到最终要执行的中断操作，这里是输出字符串的操作。

```
3 ;MAIN PART OF PROGRAM GOES HERE
1     lea     si, string1
2     lea     di, string2
3     cld
4     mov     cx, 25
5     repz    cmpsb
6     jz      match
7     lea     dx, mess2
8     jmp     short disp
9
3 match:
1     lea     dx, mess1
2
3 disp:
4     mov     ah, 09
5     int     21h
6     ret
7                                     ;return to DOS
```

问题及收获：

1. 掌握汇编代码的编写过程以及语法特点，根据例 1.1 了解汇编语言代码的执行过程。
2. 掌握 DOSBOX 编译，链接，运行的整个过程。
3. 掌握 DOSBOX 在进行 DEBUG 调试时使用的指令以及对内存和寄存器进行修改的方法。