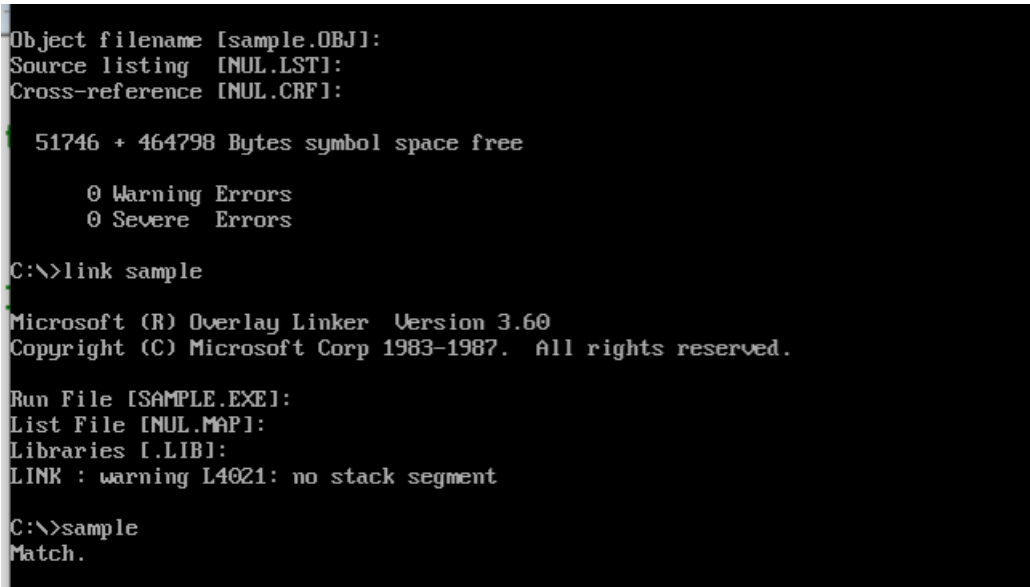


山东大学 计算机科学与技术 学院

汇编语言 课程实验报告

学号：202200130048	姓名：陈静雯	班级：6 班
实验题目：实验一工具链使用：示例 1.1		
实验学时：2	实验日期：20241014	
实验目的：熟悉实验环境、调试学习课本示例学习使用和熟悉 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. sample1_1.asm（示例 1.1 源程序）		
编译及运行结果： 编译链接执行： 		

Debug

```
C:\>sample
Match.
```

```
C:\>debug sample.exe
-g
Match.
```

u 显示程序

```
Program terminated normally
-u
076F:0000 1E          PUSH    DS
076F:0001 2BC0          SUB     AX,AX
076F:0003 50          PUSH    AX
076F:0004 B86A07      MOV     AX,076A
076F:0007 8ED8          MOV     DS,AX
076F:0009 8EC0          MOV     ES,AX
076F:000B 8D360000      LEA     SI,[0000]
076F:000F 8D3E1900      LEA     DI,[0019]
076F:0013 FC          CLD
076F:0014 B91900      MOV     CX,0019
076F:0017 F3          REPZ
076F:0018 A6          CMPSB
076F:0019 7406          JZ      0021
076F:001B 8D163B00      LEA     DX,[003B]
076F:001F EB04          JMP     0025
```

```
076F:001F EB04          JMP     0025
-u
076F:0021 8D163200      LEA     DX,[0032]
076F:0025 B409          MOV     AH,09
076F:0027 CD21      INT     21
076F:0029 CB          RETF
076F:002A 7383          JNB     FFAF
076F:002C C4068BB6      LES     AX,[B68B]
076F:0030 FA          CLI
076F:0031 FE81E6FF  INC     BYTE PTR [BX+DI+FFE6]
076F:0035 00C6          ADD     DH,AL
076F:0037 82FBFE          CMP     BL,FE
076F:003A 002B          ADD     [BP+DI],CH
076F:003C C0          DB      C0
076F:003D 50          PUSH    AX
076F:003E 8D86FBFE      LEA     AX,[BP+FEFB]
A
```

断点设置，查看数据段

```
076F:003E 8D86FBFE      LEA     AX,[BP+FEFB]
d-gOb
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
d-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 66 6F 72 77 61 72 64 e cursor forward
076A:0030 2E 4D 61 74 63 68 2E 0D-0A 24 4E 6F 20 6D 61 74 .Match...$No mat
076A:0040 63 68 21 0D 0A 24 00 00-00 00 00 00 00 00 00 ch!..$.
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 3A 00 EB >.....t.....
076A:0070 04 8D 16 31 00 B4 09 CD-21 CB 73 83 C4 06 8B B6 ...1....!.s....
```

e 修改字符串, d 再次查看

```
-e29
076A:0029  62.66  61.6f  63.72  6B.77  77.61  61.72  72.64
076A:0030  64.2e  2E.20

-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 66 6F 72 77 61 72 64  e cursor forward
076A:0030  2E 20 4D 61 74 63 68 2E-0D 0A 24 4E 6F 20 6D 61  . 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 00 00 00 00 00 00  ...2....!.....
```

另一种 e 命令修改, 避免 acs 码

```
C:\>debug sample.exe
-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
-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 00 00 00 00 00 00  ...2....!.....

-e29 'forward.'20
-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 66 6F 72 77 61 72 64  e cursor forward
076A:0030  2E 20 4D 61 74 63 68 2E-0D 0A 24 4E 6F 20 6D 61  . 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 00 00 00 00 00 00  ...2....!.....

-g
No match!

Program terminated normally
```

用 a 命令把数据区恢复原状

```
-a076a:29
076A:0029  db 'backward.'
076A:0032
-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 00 00 00 00 00 00  ...2....!.....

-g
Match.
```

用 f 命令修改

```
-f29 1 9 'forward.'20
-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 66 6F 72 77 61 72 64 e cursor forward
076A:0030 2E 20 4D 61 74 63 68 2E-0D 0A 24 4E 6F 20 6D 61 . 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 00 00 00 00 00 00 ...2....!.....
-g
No match!

Program terminated normally
```

在源文件中把 jz match 改成 jnz match

```
C:\>debug sample.exe
-g
No match!

Program terminated normally
- $
```

设置断点，发现 003b 的内容不对

```
-g19
AX=076A BX=0000 CX=0000 DX=0000 SP=FFFC BP=0000 SI=0019 DI=0032
DS=076A ES=076A SS=0769 CS=076F IP=0019 NU UP EI PL ZR NA PE NC
076F:0019 7506 JNZ 0021
-p
AX=076A BX=0000 CX=0000 DX=0000 SP=FFFC BP=0000 SI=0019 DI=0032
DS=076A ES=076A SS=0769 CS=076F IP=001B NU UP EI PL ZR NA PE NC
076F:001B 8D163B00 LEA DX,[003B] DS:003B=6F4E
-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 75 06 8D 16 3B 00 EB >.....u...;...
076A:0070 04 8D 16 32 00 B4 09 CD-21 CB 73 83 C4 06 8B B6 ...2....!..s....
-g
```

用 a 命令修改, u 查看

```
-a19
076F:0019 jz 0021
076F:001B

-u0
076F:0000 1E          PUSH    DS
076F:0001 2BC0        SUB     AX,AX
076F:0003 50          PUSH    AX
076F:0004 B86A07      MOV     AX,076A
076F:0007 8ED8        MOV     DS,AX
076F:0009 8EC0        MOV     ES,AX
076F:000B 8D360000    LEA     SI,[0000]
076F:000F 8D3E1900    LEA     DI,[0019]
076F:0013 FC          CLD
076F:0014 B91900      MOV     CX,0019
076F:0017 F3          REPZ
076F:0018 A6          CMPSB
076F:0019 7406        JZ      0021
076F:001B 8D163B00    LEA     DX,[003B]
076F:001F EB04        JMP     0025

-rip
IP 001B
:0
-g
Match.
```

用 f 命令修改, t 命令逐条查看

```
076A:0070 04 8D 16 32 00 B4 09 CD-21 CB 00 00 00 00 00 00 ...2....!.....
-f29 1 9 'forward.'20
-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 66 6F 72 77 61 72 64 e cursor forward
076A:0030 2E 20 4D 61 74 63 68 2E-0D 0A 24 4E 6F 20 6D 61 . 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 75 06 8D 16 3B 00 EB >.....u...;..
076A:0070 04 8D 16 32 00 B4 09 CD-21 CB 00 00 00 00 00 00 ...2....!.....
```

```
-t
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=000F NU UP EI PL ZR NA PE NC
076F:000F 8D3E1900 LEA DI,[0019] DS:0019=6F4D
2-t
AX=076A BX=0000 CX=007A DX=0000 SP=FFFC BP=0000 SI=0000 DI=0019
DS=076A ES=076A SS=0769 CS=076F IP=0013 NU UP EI PL ZR NA PE NC
076F:0013 FC CLD
1-t
AX=076A BX=0000 CX=007A DX=0000 SP=FFFC BP=0000 SI=0000 DI=0019
DS=076A ES=076A SS=0769 CS=076F IP=0014 NU UP EI PL ZR NA PE NC
076F:0014 B91900 MOV CX,0019
-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
```

```

-t
AX=076A BX=0000 CX=0008 DX=0000 SP=FFFC BP=0000 SI=0011 DI=002A
DS=076A ES=076A SS=0769 CS=076F IP=0019  NU UP EI NG NZ AC PE CY
076F:0019 7406      JZ      0021
-t
AX=076A BX=0000 CX=0008 DX=0000 SP=FFFC BP=0000 SI=0011 DI=002A
DS=076A ES=076A SS=0769 CS=076F IP=001B  NU UP EI NG NZ AC PE CY
076F:001B 8D163B00    LEA     DX,[003B]      DS:003B=6F4E
-t
AX=076A BX=0000 CX=0008 DX=003B SP=FFFC BP=0000 SI=0011 DI=002A
DS=076A ES=076A SS=0769 CS=076F IP=001F  NU UP EI NG NZ AC PE CY
076F:001F EB04      JMP     0025
-t
AX=076A BX=0000 CX=0008 DX=003B SP=FFFC BP=0000 SI=0011 DI=002A
DS=076A ES=076A SS=0769 CS=076F IP=0025  NU UP EI NG NZ AC PE CY
076F:0025 B409      MOV     AH,09
-t
AX=096A BX=0000 CX=0008 DX=003B SP=FFFC BP=0000 SI=0011 DI=002A
DS=076A ES=076A SS=0769 CS=076F IP=0027  NU UP EI NG NZ AC PE CY
076F:0027 CD21      INT     21
-g
No match!

Program terminated normally
-q

```

问题及收获：

1. 对照中断 21 的表格，分析程序中使用到的系统调用，说明程序是怎么调用 DOS 系统调用的

程序使用了 DOS 中断 21H 的功能 09H 来输出字符串。具体实现步骤如下：

- (1) 将要输出的字符串地址加载到 DX 寄存器。
- (2) 设置 AH 寄存器为 09H，表示调用输出字符串的功能。
- (3) 执行 int 21h 指令，触发中断调用，DOS 内核会根据 AH 寄存器的值执行相应的服务。
- (4) 字符串输出完成后，程序继续执行后续的指令，最终通过 ret 指令返回到 DOS。

```
    jz     match
    lea     dx, mess2
    jmp     short disp
match:
    lea     dx, mess1
disp:
    mov     ah, 09
    int     21h
    ret     ;return to DOS
```

2. 熟悉 DOSBox 的使用，包括其运行、配置分辨率、配置 autoexec 区以免每次打开都需要执行 mount 语句。

3. 熟悉在 DOSBox 环境下使用 masm 进行编译、使用 link 进行链接、使用 debug 或 td 进行断点调试。

4. 熟悉汇编程序的编写框架，尝试改变汇编代码重新编译链接运行观察寄存器的变化。

5. 尝试使用汇编开发的 IDE 例如 MASMPPlus 进行编程的高效编程开发、调试。

6. 运行了课本实例 1.1 的代码，熟悉了一些转移指令、文本比较指令、条件跳转指令、无条件跳转指令等。