

安徽大学人工智能学院

实验报告



课程名称: 《计算机组成原理与汇编语言》

专 业: 人工智能

学 号: WA2214014

姓 名: 杨跃浙

指导老师: 杜库

实验项目	实验 2-第二次上机实验			实验次序	2
实验地点	笃行南楼 A104	参与人员	杨跃浙	实验日期	4.23

一、实验目的

- 1、复习 DEBUG 的常用命令
- 2、做 HELLO 程序
- 3、查看汉字的不同内码练习：在记事本中输入” 1 汉 A” 字符串，分别以 ansi, Unicode,utf-8 三种编码（如下图所示）保存为 r1.txt,r2.txt,r3.txt。
- 4、在 debug 中查看 100,-1,-5 的补码。可分别执行如下三段代码，查看 ax 寄存器。
- 5、编写小程序实现 1+2，并显示结果。

二、实验内容

- 1、复习 DEBUG 的常用命令

①A

```

DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
Drive C is mounted as local directory e:\masm\
Z:\>c:

C:\>debug
-A CS:05
073F:0005 MOV CX,0
073F:0008 MOV BYTE PTR[0001],12
073F:000D JMP 20
073F:000F DB 01,02,'STRING'
073F:0017
-D CS:0
073F:0000 CD 20 3E A7 00 B9 00 00-C6 06 01 00 12 EB 11 01 . >.....
073F:0010 02 53 54 52 49 4E 47 01-01 01 01 00 02 FF FF FF .STRING.....
073F:0020 FF FF FF FF FF FF FF-FF FF FF FF 00 00 00 00 .....
073F:0030 00 00 14 00 18 00 3F 07-FF FF FF FF 00 00 00 00 .....?.....
073F:0040 05 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....
073F:0050 CD 21 CB 00 00 00 00 00-00 00 00 00 00 20 20 20 .!.....
073F:0060 20 20 20 20 20 20 20 20-00 00 00 00 00 20 20 20 .....
073F:0070 20 20 20 20 20 20 20 20-00 00 00 00 00 00 00 00 .....
-R
AX=0000 BX=0000 CX=0000 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=073F IP=0100 NU UP EI PL NZ NA PO NC
073F:0100 0000 ADD [BX+SI],AL DS:0000=CD
t

```

②D

```

DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
Drive C is mounted as local directory e:\masm\

Z:\>c:

C:\>debug
-A CS:05
073F:0005 MOV CX,0
073F:0008 MOV BYTE PTR[00011],12
073F:000D JMP 20
073F:000F DB 01,02,'STRING'
073F:0017
-D CS:0
073F:0000 CD 20 3E A7 00 B9 00 00-C6 06 01 00 12 EB 11 01 . >.....
073F:0010 02 53 54 52 49 4E 47 01-01 01 01 00 02 FF FF FF .STRING.....
073F:0020 FF FF FF FF FF FF FF FF-FF FF FF FF 00 00 00 00 .....
073F:0030 00 00 14 00 18 00 3F 07-FF FF FF FF 00 00 00 00 .....?.....
073F:0040 05 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....
073F:0050 CD 21 CB 00 00 00 00 00-00 00 00 00 00 20 20 20 .!.....
073F:0060 20 20 20 20 20 20 20 20-00 00 00 00 00 20 20 20 .....
073F:0070 20 20 20 20 20 20 20 20-00 00 00 00 00 00 00 00 .....
-R
AX=0000 BX=0000 CX=0000 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=073F IP=0100  NU UP EI PL NZ NA PO NC
073F:0100 0000          ADD     [BX+SI],AL          DS:0000=CD

```

③R

```

DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
Drive C is mounted as local directory e:\masm\

Z:\>c:

C:\>debug
-A CS:05
073F:0005 MOV CX,0
073F:0008 MOV BYTE PTR[00011],12
073F:000D JMP 20
073F:000F DB 01,02,'STRING'
073F:0017
-D CS:0
073F:0000 CD 20 3E A7 00 B9 00 00-C6 06 01 00 12 EB 11 01 . >.....
073F:0010 02 53 54 52 49 4E 47 01-01 01 01 00 02 FF FF FF .STRING.....
073F:0020 FF FF FF FF FF FF FF FF-FF FF FF FF 00 00 00 00 .....
073F:0030 00 00 14 00 18 00 3F 07-FF FF FF FF 00 00 00 00 .....?.....
073F:0040 05 00 00 00 00 00 00 00-00 00 00 00 00 00 00 00 .....
073F:0050 CD 21 CB 00 00 00 00 00 00-00 00 00 00 00 20 20 20 .!.....
073F:0060 20 20 20 20 20 20 20 20-00 00 00 00 00 20 20 20 .....
073F:0070 20 20 20 20 20 20 20 20-00 00 00 00 00 00 00 00 .....
-R
AX=0000 BX=0000 CX=0000 DX=0000 SP=00FD BP=0000 SI=0000 DI=0000
DS=073F ES=073F SS=073F CS=073F IP=0100  NU UP EI PL NZ NA PO NC
073F:0100 0000          ADD     [BX+SI],AL          DS:0000=CD

```

④T

```

DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
-N EDIT.COM
-L
-U 100 10B
1696:0100 FB      STI
1696:0101 8BC5     MOV     AX,BP
1696:0103 2E       CS:
1696:0104 FF2F     JMP     FAR [BX]
1696:0106 B440     MOV     AH,40
1696:0108 BB0200   MOV     BX,0002
1696:010B B91600   MOV     CX,0016
-R
AX=FFFF BX=0001 CX=10FE DX=6E65 SP=0080 BP=2031 SI=700A DI=0A0D
DS=075A ES=075A SS=16C2 CS=1696 IP=0010  NU UP EI PL NZ NA PO NC
1696:0010 8BE8     MOV     BP,AX
-T
AX=FFFF BX=0001 CX=10FE DX=6E65 SP=0080 BP=FFFF SI=700A DI=0A0D
DS=075A ES=075A SS=16C2 CS=1696 IP=0012  NU UP EI PL NZ NA PO NC
1696:0012 8CC0     MOV     AX,ES
-T
AX=075A BX=0001 CX=10FE DX=6E65 SP=0080 BP=FFFF SI=700A DI=0A0D
DS=075A ES=075A SS=16C2 CS=1696 IP=0014  NU UP EI PL NZ NA PO NC
1696:0014 051000   ADD     AX,0010
-

```

⑤P

```

DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
-R
AX=FFFF BX=0001 CX=10FE DX=6E65 SP=0080 BP=2031 SI=700A DI=0A0D
DS=075A ES=075A SS=16C2 CS=1696 IP=0010  NU UP EI PL NZ NA PO NC
1696:0010 8BE8     MOV     BP,AX
-T
AX=FFFF BX=0001 CX=10FE DX=6E65 SP=0080 BP=FFFF SI=700A DI=0A0D
DS=075A ES=075A SS=16C2 CS=1696 IP=0012  NU UP EI PL NZ NA PO NC
1696:0012 8CC0     MOV     AX,ES
-T
AX=075A BX=0001 CX=10FE DX=6E65 SP=0080 BP=FFFF SI=700A DI=0A0D
DS=075A ES=075A SS=16C2 CS=1696 IP=0014  NU UP EI PL NZ NA PO NC
1696:0014 051000   ADD     AX,0010
-P
AX=076A BX=0001 CX=10FE DX=6E65 SP=0080 BP=FFFF SI=700A DI=0A0D
DS=075A ES=075A SS=16C2 CS=1696 IP=0017  NU UP EI PL NZ NA PE NC
1696:0017 0E       PUSH    CS
-P
AX=076A BX=0001 CX=10FE DX=6E65 SP=007E BP=FFFF SI=700A DI=0A0D
DS=075A ES=075A SS=16C2 CS=1696 IP=0018  NU UP EI PL NZ NA PE NC
1696:0018 1F       POP     DS
-

```

⑥U

```

DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
-Q
C:\>DEBUG HELLO.ASM
-U
073F:0100 61          DB      61
073F:0101 7373       JNB     0176
073F:0103 756D       JNZ     0172
073F:0105 65          DB      65
073F:0106 206373     AND     [BP+DI+73],AH
073F:0109 3A7031     CMP     DH,[BX+SI+31]
073F:010C 0D0A70     OR      AX,700A
073F:010F 3120       XOR     [BX+SI],SP
073F:0111 7365       JNB     0178
073F:0113 67          DB      67
073F:0114 6D          DB      6D
073F:0115 65          DB      65
073F:0116 6E          DB      6E
073F:0117 740D       JZ      0126
073F:0119 0A20     OR      AH,[BX+SI]
073F:011B 69          DB      69
073F:011C 6E          DB      6E
073F:011D 66          DB      66
073F:011E 6F          DB      6F
073F:011F 206462     AND     [SI+62],AH

```

⑦Q

```

DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX
DS=075A ES=075A SS=16C2 CS=1696 IP=0010  NV UP EI PL NZ NA PO NC
1696:0010 8BE8       MOV     BP,AX
-T
AX=FFFF BX=0001 CX=10FE DX=6E65 SP=0080 BP=FFFF SI=700A DI=0A0D
DS=075A ES=075A SS=16C2 CS=1696 IP=0012  NV UP EI PL NZ NA PO NC
1696:0012 8CC0       MOV     AX,ES
-T
AX=075A BX=0001 CX=10FE DX=6E65 SP=0080 BP=FFFF SI=700A DI=0A0D
DS=075A ES=075A SS=16C2 CS=1696 IP=0014  NV UP EI PL NZ NA PO NC
1696:0014 051000     ADD     AX,0010
-P
AX=076A BX=0001 CX=10FE DX=6E65 SP=0080 BP=FFFF SI=700A DI=0A0D
DS=075A ES=075A SS=16C2 CS=1696 IP=0017  NV UP EI PL NZ NA PE NC
1696:0017 0E          PUSH    CS
-P
AX=076A BX=0001 CX=10FE DX=6E65 SP=007E BP=FFFF SI=700A DI=0A0D
DS=075A ES=075A SS=16C2 CS=1696 IP=0018  NV UP EI PL NZ NA PE NC
1696:0018 1F          POP     DS
-Q
C:\>_

```

2、做 HELLO 程序

The image consists of two screenshots from a DOSBox window. The top screenshot shows a text editor (EDIT) with the file C:\H2.ASM. The code defines a data segment with a string 'HELLO!', sets up a code segment, and contains assembly instructions to load the string into the AX register and output it using INT 21H. The bottom screenshot shows the command prompt where the MASM assembler and the LINK linker are used to create an executable file H2.EXE. The linker output shows a warning about no stack segment.

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: EDIT
File Edit Search View Options Help
C:\H2.ASM
DATA SEGMENT
    INFO DB 'HELLO!','$'
DATA ENDS

CODE SEGMENT
ASSUME CS:CODE
START:
    MOV AX,DATA
    MOV DS,AX
    MOV DX,OFFSET INFO
    MOV AX,0900H
    INT 21H
    MOV AX,4C00H
    INT 21H
CODE ENDS
END START

F1=Help | Line:16 Col:12
```

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX
C:\>MASM H2.ASM
Microsoft (R) Macro Assembler Version 5.00
Copyright (C) Microsoft Corp 1981-1985, 1987. All rights reserved.

Object filename [H2.OBJ]:
Source listing [NUL.LST]:
Cross-reference [NUL.CRF]:

51728 + 464816 Bytes symbol space free

0 Warning Errors
0 Severe Errors

C:\>LINK H2.OBJ

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

Run File [H2.EXE]:
List File [NUL.MAP]:
Libraries [.LIB]:
LINK : warning L4021: no stack segment

C:\>
```

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX
Microsoft (R) Macro Assembler Version 5.00
Copyright (C) Microsoft Corp 1981-1985, 1987. All rights reserved.

Object filename [H2.OBJ]:
Source listing [NUL.LST]:
Cross-reference [NUL.CRF]:

51728 + 464816 Bytes symbol space free

0 Warning Errors
0 Severe Errors

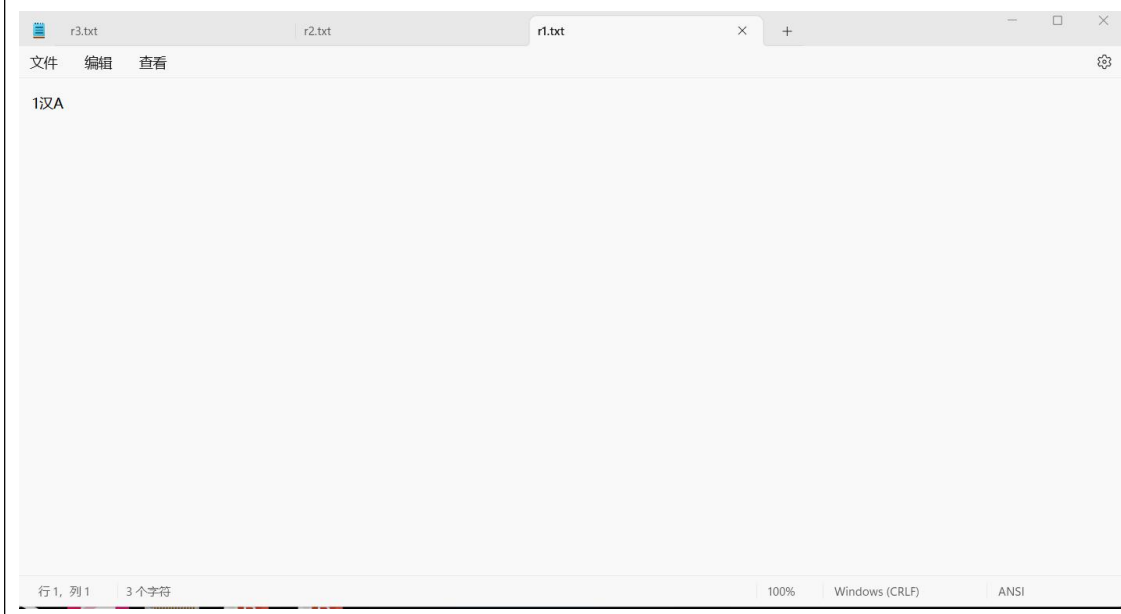
C:\>LINK H2.OBJ

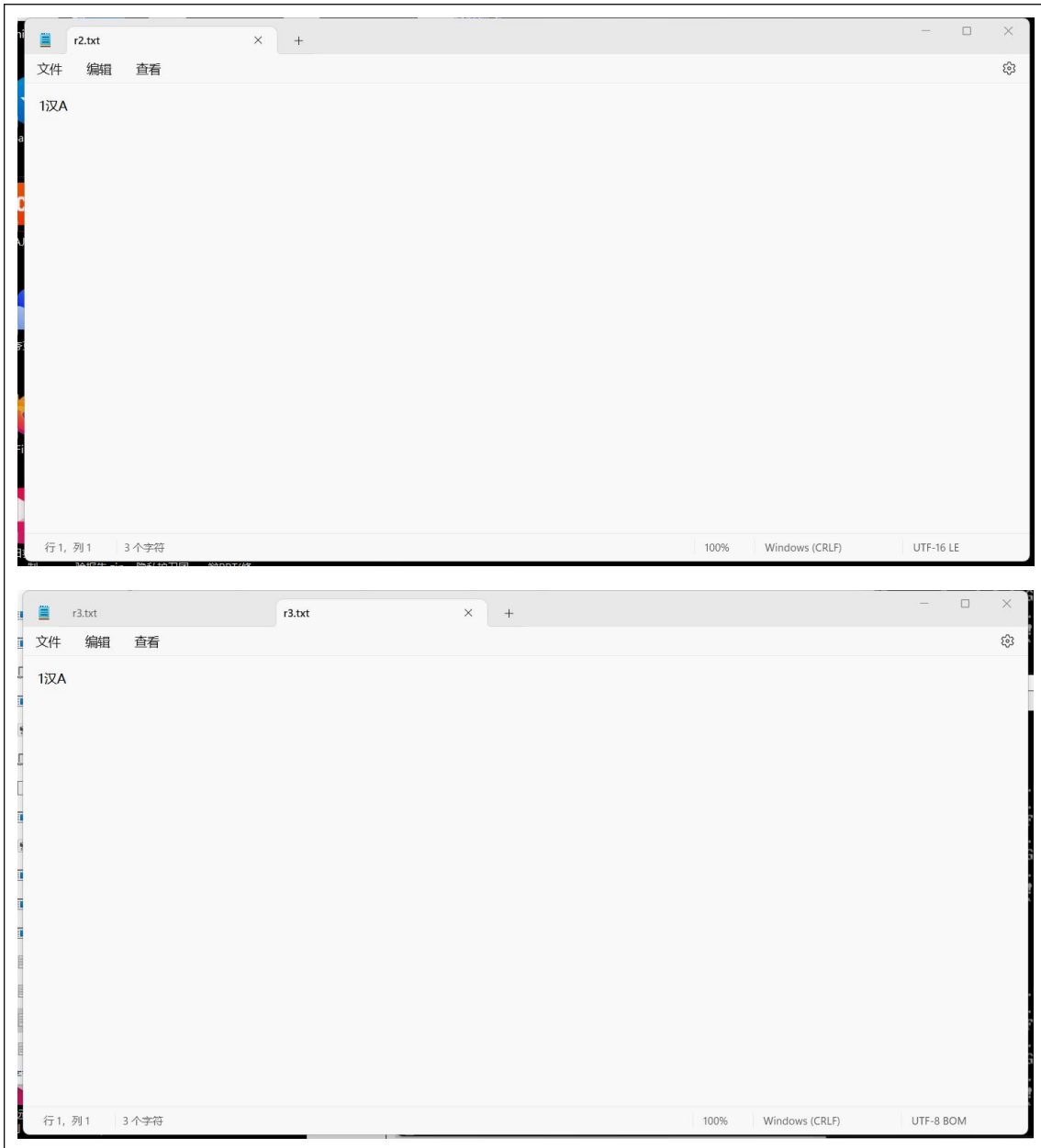
Microsoft (R) Overlay Linker Version 3.60
Copyright (C) Microsoft Corp 1983-1987. All rights reserved.

Run File [H2.EXE]:
List File [NUL.MAP]:
Libraries [.LIB]:
LINK : warning L4021: no stack segment

C:\>H2.EXE
HELLO!
C:\>
```

3、查看汉字的不同内码练习：在记事本中输入” 1 汉 A” 字符串，分别以 ansi, utf-16 LE,utf-8 三种编码（如下图所示）保存为 r1.txt,r2.txt,r3.txt。






```

DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
0 Severe Errors

C:\>LINK H2.OBJ

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

Run File [H2.EXE]:
List File [NUL.MAP]:
Libraries [.LIB]:
LINK : warning L4021: no stack segment

C:\>H2.EXE
HELLO!
C:\>DEBUG R1.TXT
-D
073F:0100 31 BA BA 41 00 B8 05 00-50 FF 36 3C 21 FF 36 3A 1..A....P.6<!.6:
073F:0110 21 E8 4C 14 83 C4 06 89-46 F8 89 56 34 00 2E 07 !.L....F..U4...
073F:0120 0B 06 BA 17 75 1C 83 7E-0A FF 74 08 8B 46 06 0B ....u...~..t..F..
073F:0130 46 08 75 0E A1 3A 21 8B-16 3C 21 A3 B8 17 89 16 F.u...!...<!....
073F:0140 BA 17 C4 5E F8 2B C0 26-89 47 10 26 89 47 0E 8B ...^..+.&.G.&.G..
073F:0150 46 06 8B 56 08 26 89 47-0A 26 89 57 0C 8A 46 0A F..U.&.G.&.W..F..
073F:0160 26 88 47 22 8A 46 0C 26-88 47 05 A1 32 21 8B 16 &.G".F.&.G..2!..
073F:0170 34 21 89 46 FC 89 56 FE-A1 5E 27 8B 16 60 27 A3 4!.F..U..^'...'..

```

```

DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
C:\>H2.EXE
HELLO!
C:\>DEBUG R1.TXT
-D
073F:0100 31 BA BA 41 00 B8 05 00-50 FF 36 3C 21 FF 36 3A 1..A....P.6<!.6:
073F:0110 21 E8 4C 14 83 C4 06 89-46 F8 89 56 34 00 2E 07 !.L....F..U4...
073F:0120 0B 06 BA 17 75 1C 83 7E-0A FF 74 08 8B 46 06 0B ....u...~..t..F..
073F:0130 46 08 75 0E A1 3A 21 8B-16 3C 21 A3 B8 17 89 16 F.u...!...<!....
073F:0140 BA 17 C4 5E F8 2B C0 26-89 47 10 26 89 47 0E 8B ...^..+.&.G.&.G..
073F:0150 46 06 8B 56 08 26 89 47-0A 26 89 57 0C 8A 46 0A F..U.&.G.&.W..F..
073F:0160 26 88 47 22 8A 46 0C 26-88 47 05 A1 32 21 8B 16 &.G".F.&.G..2!..
073F:0170 34 21 89 46 FC 89 56 FE-A1 5E 27 8B 16 60 27 A3 4!.F..U..^'...'..
-Q
C:\>DEBUG R2.TXT
-D
073F:0100 FF FE 31 00 49 6C 41 00-50 FF 36 3C 21 FF 36 3A ..1.IIA.P.6<!.6:
073F:0110 21 E8 4C 14 83 C4 06 89-46 F8 89 56 34 00 2E 07 !.L....F..U4...
073F:0120 0B 06 BA 17 75 1C 83 7E-0A FF 74 08 8B 46 06 0B ....u...~..t..F..
073F:0130 46 08 75 0E A1 3A 21 8B-16 3C 21 A3 B8 17 89 16 F.u...!...<!....
073F:0140 BA 17 C4 5E F8 2B C0 26-89 47 10 26 89 47 0E 8B ...^..+.&.G.&.G..
073F:0150 46 06 8B 56 08 26 89 47-0A 26 89 57 0C 8A 46 0A F..U.&.G.&.W..F..
073F:0160 26 88 47 22 8A 46 0C 26-88 47 05 A1 32 21 8B 16 &.G".F.&.G..2!..
073F:0170 34 21 89 46 FC 89 56 FE-A1 5E 27 8B 16 60 27 A3 4!.F..U..^'...'..

```



ANSI(即 cp936 编码字符集) 存储一个英文用__1__字节， 存储一个汉字用__2__字节。

utf-16 LE 存储一个英文用__2__字节， 存储一个汉字用__2__字节。

从图中可见，记事本为标记 utf-16 LE 文件， 会在文件头部存入 __FF FE__ 两个字节。

Utf-8 存储一个英文用__1__字节， 存储一个汉字用__3__字节。从图中可见，记事本为标记 utf-8 文件， 会在文件头部存入 __EF BB BF__ 三个字节，这三个字节被称为 BOM 头。

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
-Q
C:\>DEBUG r3.txt
-d
073F:0100 EF BB BF 31 E6 B1 89 41-99 FF 36 3C 21 FF 36 3A ...1...A..6<?.6:
073F:0110 21 E8 4C 14 83 C4 06 89-46 F8 89 56 34 00 2E 07 ?..L....F..U4...
073F:0120 0B 06 BA 17 75 1C 83 7E-0A FF 74 08 8B 46 06 0B ....u...~.t..F..
073F:0130 46 08 75 0E A1 3A 21 8B-16 3C 21 A3 B8 17 89 16 F.u...?!<?.....
073F:0140 BA 17 C4 5E F8 2B C0 26-89 47 10 26 89 47 0E 8B ...^..+.&.G.&.G..
073F:0150 46 06 8B 56 08 26 89 47-0A 26 89 57 0C 8A 46 0A F..U.&.G.&.W..F.
073F:0160 26 88 47 22 8A 46 0C 26-88 47 05 A1 32 21 8B 16 &.G".F.&.G..2?...
073F:0170 34 21 89 46 FC 89 56 FE-A1 5E 27 8B 16 60 27 A3 4?.F..U..^'...'..
-q
C:\>debug name.txt
-d
073F:0100 FF FE 68 67 C3 8D 59 6D-99 FF 36 3C 21 FF 36 3A ..hg...Ym..6<?.6:
073F:0110 21 E8 4C 14 83 C4 06 89-46 F8 89 56 34 00 2E 07 ?..L....F..U4...
073F:0120 0B 06 BA 17 75 1C 83 7E-0A FF 74 08 8B 46 06 0B ....u...~.t..F..
073F:0130 46 08 75 0E A1 3A 21 8B-16 3C 21 A3 B8 17 89 16 F.u...?!<?.....
073F:0140 BA 17 C4 5E F8 2B C0 26-89 47 10 26 89 47 0E 8B ...^..+.&.G.&.G..
073F:0150 46 06 8B 56 08 26 89 47-0A 26 89 57 0C 8A 46 0A F..U.&.G.&.W..F.
073F:0160 26 88 47 22 8A 46 0C 26-88 47 05 A1 32 21 8B 16 &.G".F.&.G..2?...
073F:0170 34 21 89 46 FC 89 56 FE-A1 5E 27 8B 16 60 27 A3 4?.F..U..^'...'..
```

写出你的名字对应的 utf-16 LE 编码串是____68 67 C3 8D 59
6D_____

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: EDIT
File Edit Search View Options Help
C:\>BUMA.ASM
assume cs:p1
p1 segment
start:
    mov ax,100
    mov ax,0
    sub ax,1
    mov ax,0
    sub ax,5
p1 ends
end start
-
```

F1=Help | Line:11 Col:3

```

DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DEBUG
AX=0064 BX=0000 CX=000F DX=0000 SP=0000 BP=0000 SI=0000 DI=0000
DS=075A ES=075A SS=0769 CS=076A IP=0003  NV UP EI PL NZ NA PO NC
076A:0003 B80000      MOV     AX,0000
-t
AX=0000 BX=0000 CX=000F DX=0000 SP=0000 BP=0000 SI=0000 DI=0000
DS=075A ES=075A SS=0769 CS=076A IP=0006  NV UP EI PL NZ NA PO NC
076A:0006 2D0100      SUB     AX,0001
-t
AX=FFFF BX=0000 CX=000F DX=0000 SP=0000 BP=0000 SI=0000 DI=0000
DS=075A ES=075A SS=0769 CS=076A IP=0009  NV UP EI NG NZ AC PE CY
076A:0009 B80000      MOV     AX,0000
-t
AX=0000 BX=0000 CX=000F DX=0000 SP=0000 BP=0000 SI=0000 DI=0000
DS=075A ES=075A SS=0769 CS=076A IP=000C  NV UP EI NG NZ AC PE CY
076A:000C 2D0500      SUB     AX,0005
-t
AX=FFFB BX=0000 CX=000F DX=0000 SP=0000 BP=0000 SI=0000 DI=0000
DS=075A ES=075A SS=0769 CS=076A IP=000F  NV UP EI NG NZ AC PO CY
076A:000F 01B85C00     ADD     [BX+SI+005C],DI      DS:005C=0000

```

4、在 debug 中查看 100, -1, -5 的补码。可分别执行如下三段代码，查看 ax 寄存器。

```
mov ax, 100
```

```
mov ax, 0
```

```
sub ax, 1
```

```
mov ax, 0
```

```
sub ax, 5
```

-1 的 16 位长补码是_____FFFF_____

-5 的 16 位长补码是_____FFFB_____

5、编写小程序实现 2*3，并显示结果。

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: EDIT
File Edit Search View Options Help
C:\MUL.ASM
assume cs:p1,ds:data
data segment
    info db '2*3='
    result db 0,'$'
data ends
p1 segment
start:
    mov ax,data
    mov ds,ax
    mov a1,2
    mov b1,3
    imul a1,b1
    add a1,30h
    mov result[0],a1

    mov dx,offset info
    mov ax,0900h
    int 21h
    mov ax,4c00h
    int 21h
p1 ends
end start
F1=Help Line:22 Col:12
```

```
DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX
076A:000F 01B85C00 ADD [BX+SI+005C],DI DS:005C=0000
-q
C:\>edit
C:\>masm mul.asm
Microsoft (R) Macro Assembler Version 5.00
Copyright (C) Microsoft Corp 1981-1985, 1987. All rights reserved.

Object filename [mul.OBJ]:
Source listing [NUL.LST]:
Cross-reference [NUL.CRF]:
mul.asm(3): warning A4001: Extra characters on line
mul.asm(10): error A2009: Symbol not defined: A1
mul.asm(11): error A2009: Symbol not defined: B1
mul.asm(12): error A2009: Symbol not defined: A1
mul.asm(13): error A2009: Symbol not defined: A1
mul.asm(14): error A2009: Symbol not defined: A1

51652 + 464892 Bytes symbol space free

1 Warning Errors
5 Severe Errors
C:\>_
```

调试发现错误 修改代码:

```
NUL.ASK13
1  assume cs:pl,ds:data
2  data segment
3      info db '2*3='
4      result db 0,'q'
5  data ends
6  pl segment
7  start:
8      mov ax,data
9      mov ds,ax
10     mov al,2
11     mov bl,3
12     mul bl
13     add al,30h
14     mov result[0],al
15
16     mov dx,offset info
17     mov ax,0900h
18     int 21h
19     mov ax,4c00h
20     int 21h
21 pl ends
22 end start
23
```

DOSBox 0.74-3, Cpu speed: 3000 cycles, Frameskip 0, Program: DOSBOX

Microsoft (R) Macro Assembler Version 5.00
Copyright (C) Microsoft Corp 1981-1985, 1987. All rights reserved.

Object filename [mul.OBJ]:
Source listing [NUL.LST]:
Cross-reference [NUL.CRF]:

51652 + 464892 Bytes symbol space free

0 Warning Errors
0 Severe Errors

C:\>link mul.obj

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

Run File [MUL.EXE]:
List File [NUL.MAP]:
Libraries [.LIB]:
LINK : warning L4021: no stack segment

C:\>mul.exe
2*3=6
C:\>_