

基本方式：

1. 编写完成源代码

```
; 数据段，用于定义数据常量
datas segment
    string db 'Hello,World!$' ; 定义字符串常量
datas ends

; 代码段，包含程序的执行代码
code segment
    assume cs:code,ds:datas

; 程序入口点
start:
    mov ax,datas ; 将数据段的段基址移入AX寄存器
    mov ds,ax ; 将AX中的段基址送入DS寄存器，使DS指向数据段
    mov dx,offset string ; 将字符串的偏移地址移入DX寄存器
    mov ah,9 ; 设置AH寄存器为9，准备调用DOS中断服务打印字符串
    int 21h ; 调用DOS中断服务，打印字符串
    mov ah,4ch ; 设置AH寄存器为4CH，准备调用DOS中断服务退出程序
    int 21h ; 调用DOS中断服务，退出程序
code ends
; 程序结束，告诉编译器到达代码段的终点
end start
```

2. 汇编

```
C:\>masm hello.asm
Microsoft (R) Macro Assembler Version 5.10
Copyright (C) Microsoft Corp 1981, 1988. All rights reserved.

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

50084 + 463321 Bytes symbol space free

0 Warning Errors
0 Severe Errors
```

3. 链接

```
C:\>dir
Directory of C:\.
. <DIR> 23-09-2024 10:41
.. <DIR> 23-09-2024 10:40
DEBUG EXE 20,634 15-11-2019 13:22
EXE2BIN EXE 8,424 15-11-2019 13:22
HELLO ASM 972 18-09-2024 16:13
HELLO OBJ 126 23-09-2024 10:41
LINK EXE 65,475 15-11-2019 13:22
MASM EXE 110,703 15-11-2019 13:22
6 File(s) 206,334 Bytes.
2 Dir(s) 262,111,744 Bytes free.

C:\>link hello.obj

Microsoft (R) Overlay Linker Version 3.64
Copyright (C) Microsoft Corp 1983-1988. All rights reserved.

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

4. 结果如下：

```
C:\>dir
Directory of C:\.
. <DIR> 23-09-2024 10:42
.. <DIR> 23-09-2024 10:40
DEBUG EXE 20,634 15-11-2019 13:22
EXE2BIN EXE 8,424 15-11-2019 13:22
HELLO ASM 972 18-09-2024 16:13
HELLO EXE 544 23-09-2024 10:42
HELLO OBJ 126 23-09-2024 10:41
LINK EXE 65,475 15-11-2019 13:22
MASM EXE 110,703 15-11-2019 13:22
7 File(s) 206,878 Bytes.
2 Dir(s) 262,111,744 Bytes free.

C:\>hello.exe
Hello,World!
```

```

C:\>hello.exe
Hello,World!
C:\>debug hello.exe
-u
076B:0000 B86A07      MOV     AX,076A
076B:0003 8ED8          MOV     DS,AX
076B:0005 BA0000      MOV     DX,0000
076B:0008 B409          MOV     AH,09
076B:000A CD21      INT     21
076B:000C B44C          MOV     AH,4C
076B:000E CD21      INT     21
076B:0010 0050B8      ADD     [BX*SI-4B1],DL
076B:0013 AA          STOSB
076B:0014 07          POP     ES
076B:0015 50          PUSH    AX
076B:0016 B8C3      MOV     AX,BX
076B:0018 034604      ADD     AX,[BP+04]
076B:001B 40          INC     AX
076B:001C 50          PUSH    AX
076B:001D EB6078      CALL    7880

```

另类方式：直接将代码和书籍写到内存中去执行。

```

C:\>debug hello.exe
-e 076a: 0
076A:0000 48.48 65.65 6C.6c 6C.6c 6F.6f 2C.24 57.

-e 076b: 0
076B:0000 B8.b8 6A.6b 07.07 8E.be D8.d8 BA.ba 00.02 00.00
076B:0008 B4.b4 09.09 CD.cd 21.21 B4.b8 4C.00 CD.4c 21.cd
076B:0010 00.21 00.

-r cs
CS 076B
:076b
-r ds
DS 075A
:076a
-r ip
IP 0000
:0
-g
Hello

```

1. 字符串 "Hello\$" 的 ASCII 码为 48 65 6C 6C 6F 24，这会将字符串的 ASCII 值写入 076A:0 开始的内存地址。

2. 机器码 b8 6b 07 be d8 ba 02 00 b4 09 cd 21 b8 00 4c cd 21 是源代码的对应机器码，写入到内存 CS:076B 开始的内存地址。

3. 设置 CS 和 IP（代码段寄存器和指令指针）指向程序的入口点。

4. 设置 DS 寄存器，指向数据段基址。

5. 最后，执行程序 -g。