4.4

STACK SEGMENT USE16 STACK

DB 200 DUP(0)

STACK ENDS

DATA SEGMENT USE16

STR DW 3 DUP(0)

A DW ?

B DW ?

DATA ENDS

CODE SEGEMENT USE16

ASSUME CS:CODE,DS:DATA,SS:STACK

BEGIN:

MOV SI,OFFSET STR

MOV AX,[SI]

CMP AX,[SI]+2;

JNE L1

MOV AX,[SI]+4

MOV A,AX

MOV B, SI+4;

JMP EXIT

L1: CMP AX,[SI]+4

JNE L2

MOV AX,[SI]+2

MOV A, AX

MOV B,SI+2

JMP EXIT

L2:

MOV AX,[SI]

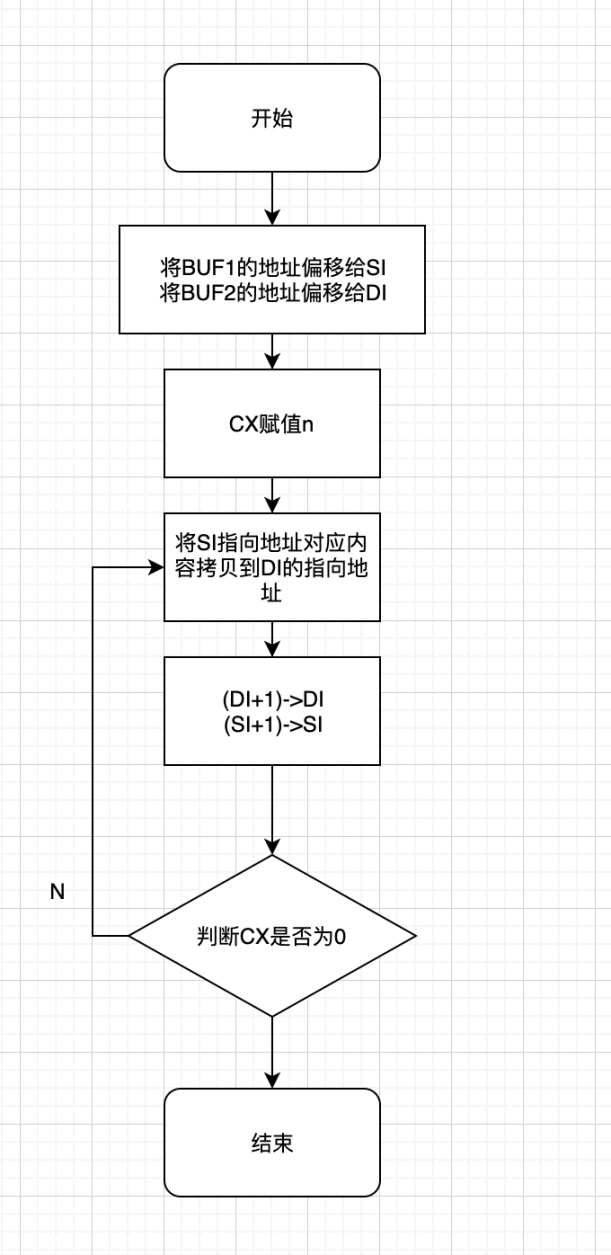
MOV A, AX

MOV B,SI

JMP EXIT

EXIT

4.5



1. 功能：将BUF1的内容复制到BUF2中
2. 0FFFFH+1次
3. 不能，CX的值不确定，所以执行次数无法确定
4. SI指向的内容会被拷贝到BUF2的n个字节存储区
5. n=1,循环体被执行一次,n != 1 则循环体被无限次执行，即造成死循环

4.7

STACK SEGMENT USE16 STACK

DB 200 DUP(0)

STACK ENDS

DATA SEGMENT USE16

STR DB “THE ASCII CODE OF$”

IS DB “IS”

A DB ?

BUF DB “123456789abcdef”

RES DW ?

CRLF DB 0DH,0AH,’$’

DATA ENDS

CODE SEGEMENT USE16

ASSUME CS:CODE,DS:DATA,SS:STACK

L\_INPUT:

LEA DX,STR

MOV AH 9

INT 21H

MOV AH,1

INT 21H

CMP AL,0

JE EXIT

LEA DX,IS

MOV AH 9

INT 21H

JMP SHOW

SHOW:

MOV AH,0

MOV BX,16

LOOP\_SHOW:

MOV DX,0

DIV BX;

PUSH DX;

CMP AX,0;

JNE LOOP\_SHOW;

MOV CX,2

LOOPA:

POP SI;

MOVZX DX,BUF[SI];

MOV AH,2

INT 21H

LOOP LOOPA

LEA DX,CRLF

    MOV AH,9

    INT 21H

JMP L\_INPUT

EXIT:

4.12

思路：将输入的字符串转化为整数，然后不断模16，最后输出

STACK SEGMENT USE16 STACK

DB 200 DUP(0)

STACK ENDS

DATA SEGMENT USE16

    INPUT DB "INPUT: $"

    OUTPUT DB "OUTPUT: $"

    BUF DB "123456789abcdef"

    BUF2 DB 11

         DB ?

         DB 11 DUP(0)

    CRLF DB 0DH,0AH,'$'

DATA ENDS

CODE SEGEMENT USE16

ASSUME CS:CODE,DS:DATA,SS:STACK

NF10T16 PROC

L\_INPUT:

    LEA DX,INPUT

    MOV AH 9

    INT 21H

    CMP AL,0

    JE RETURN

    LEA DX,BUF2;输入字符串

    MOV AH,10

    INT 21H

    CMP IN\_GOOD+1,0;比较长度

    JE L\_INPUT;

    ;字符串转化为数字

    MOV CX,IN\_GOOD+1;

    LEA SI,IN\_GOOD;

    ADD SI,2

    MOV AX,0;

LOOP\_S2I:

    ;字符转化为整数

    MOV DX,[SI]

    SUB DX,'0';

    MOV DI,CX-1;

    IMUL DI,10

    IMUL DX,DI

    ADD AX,DX

    LOOP LOOP\_S2I

    MOV BX,16;

LOOP\_SHOW:

    ;整数转化为16进制

    MOV DX,0

    DIV BX;

    PUSH DX;

    INC CX;

    CMP AX,0;

    JNE LOOP\_SHOW;

    LEA DX,CRLF

    MOV AH,9

    INT 21H

    LEA DX,INPUT

    MOV AH 9

    INT 21H

LOOPA:

    ;输出

    POP SI;

    MOVZX DX,BUF[SI];

    MOV AH,2

    INT 21H

    LOOP LOOPA

RETURN:   RET

NF10T16 ENDP

**附加题：**

1. 编写子程序OPPOSITE，实现求相反数的功能，子程序OPPOSITE用AX返回相反数的值。主程序的调用格式如下：

… …

X DW ?

… …

PUSH X

CALL NEAR PTR OPPOSITE

ADD SP, 2

解：

OPPOSITE NEAR PROC

POP AX;

NEG AX;

RET

OPPOSITE ENDP

1. 编写子程序PROG，实现如下功能：(X - Y) / Z（X、Y、Z是有符号数，每一步都不需考虑运算溢出的情况），子程序将商保存到AX、余数保存到DL。主程序的调用格式如下：：

… …

X DW ?

Y DB ?

Z DB ?

… …

MOV AL, Y

MOV AH, Z

PUSH AX

PUSH X

CALL FAR PTR SUBPROM

ADD SP, 4

… …

解：

PROG PROC FAR

POP BX;Y,Z

POP CX;X

MOVZX DX,BL;

MOVZX SI,BH;

SUB CX,DX;

MOV AX,CX;

DIV SI;

RET

PROG ENDP

1. 阅读下面的程序，分析程序运行的结果。

.386

CODE SEGMENT USE16

ASSUME CS: CODE

MSG DB '123$ '

L0: PUSH OFFSET L1

CALL DO

PUSH CS

POP DS

MOV DX, OFFSET MSG

MOV AH, 9

INT 21H

L1: MOV AH, 4CH

INT 21H

DO PROC

POP AX;IP

POP BX;OFFSET L1

PUSH BX

MOV DL, MSG

MOV AH, 2

INT 21H

INC MSG

RET

DO ENDP

CODE ENDS

END L0

输出：1