

****

《汇编语言》实验报告

（五）

**姓　　名 王宇翔**

**学　　号 13720192200474**

**学 院 信息学院**

**专 业 软件工程类**

**2020 年 11 月18日**

# 实验目的

学习利用 8086 的硬指令完成基本的程序设计，进一步熟悉程序结构和指令的具 体功能；

# 实验环境

MASM集成开发环境，DOSBox虚拟机运行

# 实验内容

(1) 根据下述情况，分别编写程序，记录 BX 中 1 的个数（需要考虑 BX 中二进制

串的特殊情况），要求如下：

 循环次数已知

 循环次数未知

(2) 按照下列要求，编写相应程序段。

1) 起始地址为 string 的主存单元中存放一个字符串（长度大于 6），把该字

符串中的第 1 个和第 6 个字符（字节量）传送给 DX 寄存器；

2) 从主存 buffer 开始的 4 字节中保存了 4 个非压缩 BCD 码，现按低（高）

地址对低（高）位的原则，将他们合并到 DX 中。

3) 假设从 B800H:0 开始存放有 100 个 16 位无符号数，编程求它们的和，并

存在 DX.AX 中

4) 一个 100 字节元素的数组首地址为 array，将每个元素减 1（不考虑溢出）。

(3) 把内存中从 PACKED 开始的 10 个字节单元中的 16 位压缩 BCD 数转换成非压

缩 BCD 数，并把结果存放在 UNPACKED 开始的 20 个字节单元中；将下列代

码补充完整，并且自己定义 PACKED 中的数据，将 UNPACKED 中的结果展示

出来。

MOV DX, \_\_\_\_\_

MOV CL, \_\_\_\_\_\_

MOV SI, 0

MOV DI, \_\_\_

CONVERT: MOV AL, [SI+PACKED]

MOV AH, AL

AND AL, 0FH

\_\_\_\_\_\_\_\_\_\_

MOV [DI+UNPACKED], \_\_\_\_

ADD DI, \_\_\_\_

\_\_\_\_\_\_

DEC DX

JNZ CONVERT

(4) 给定一个有序数组（均小于 FFH 例如 02H, 07H, 0BH, 0FH, 13H, 1CH, 24H, 39H,

40H, 57H, 68H）和一个目标值（例如 79H），请判断数组中是否含有两个数的

和为目标值，请设计一个算法，将时间复杂度控制在 O(n)，编程实现并验证

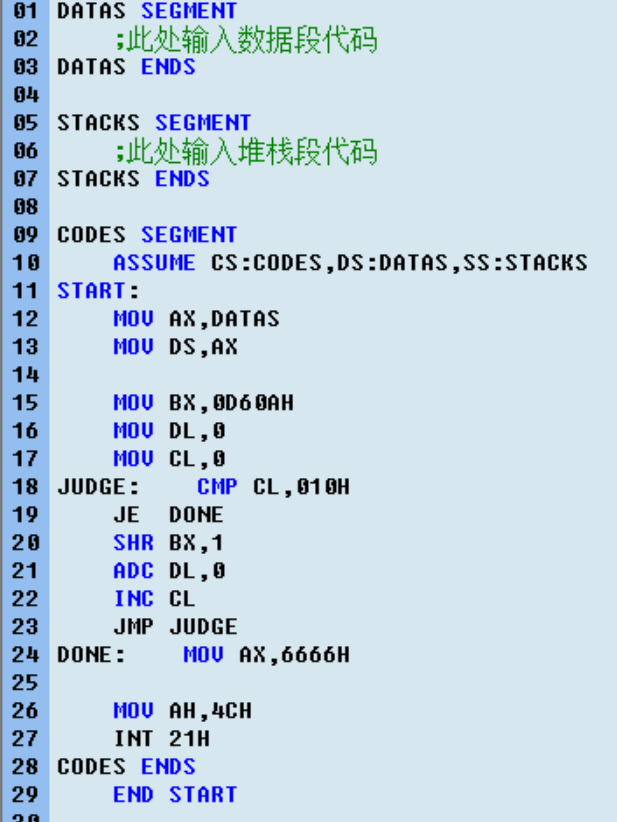
你的算法。

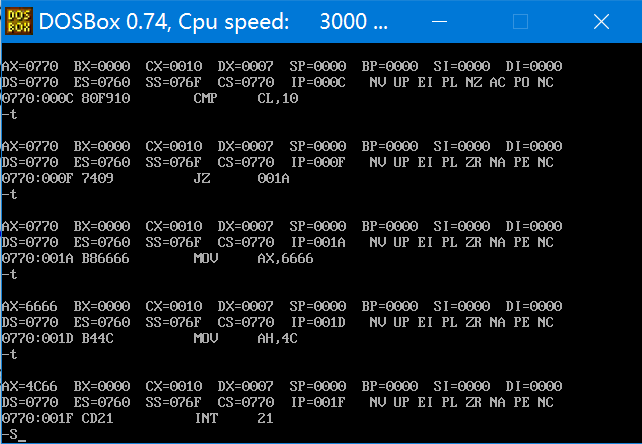
# 实验具体实现

**1.记录BX中1的个数：**

我定义了一个BX为D60AH，二进制为1101 0110 0000 1010 共有7个1

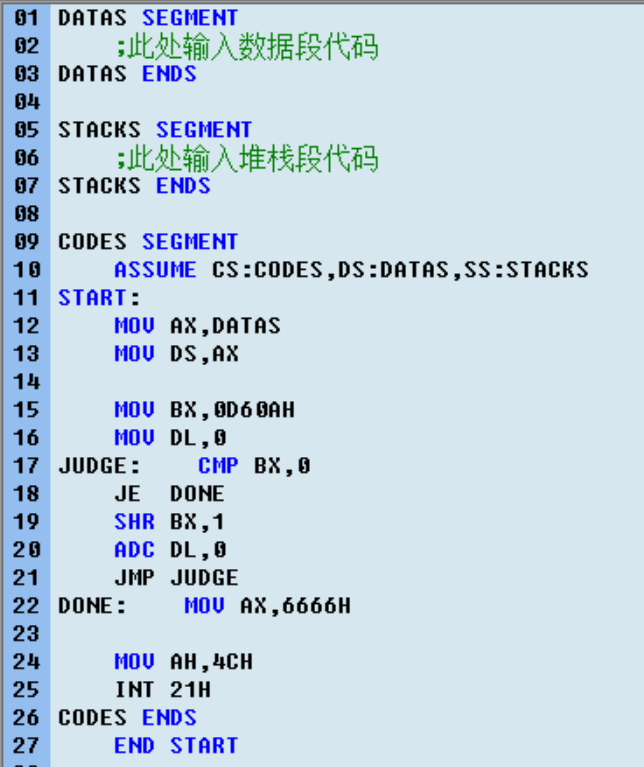
a.指定循环次数（用CL判断循环次数）：

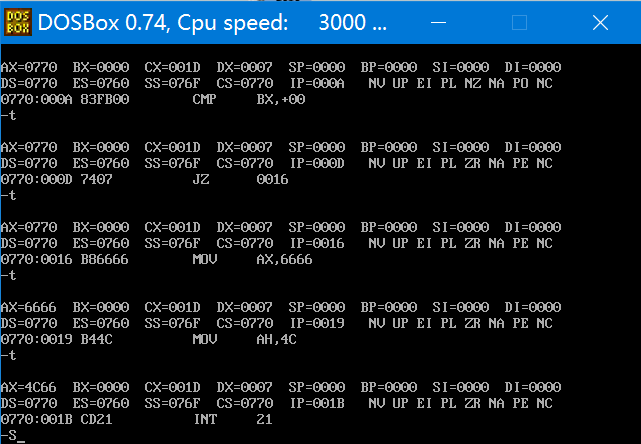




答案在CL中，答案正确。

b.不指定循环次数，我判断BX来跳出循环：



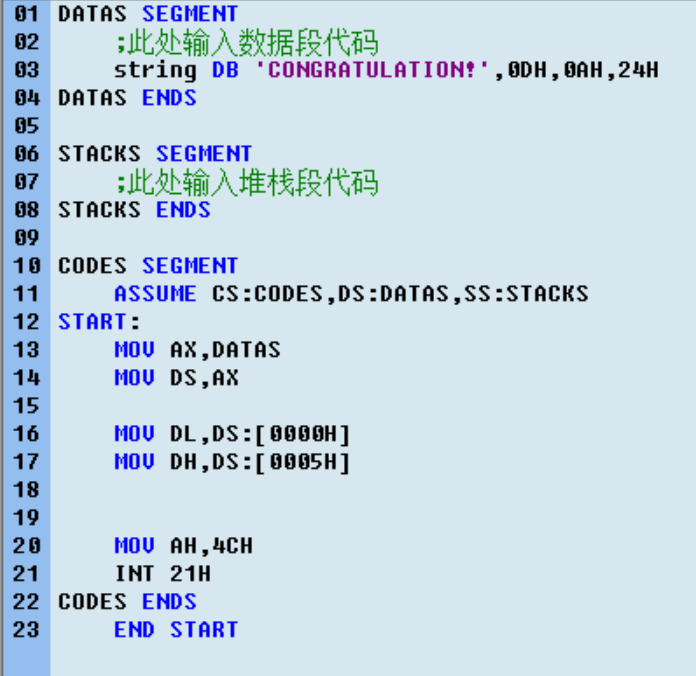


答案正确。

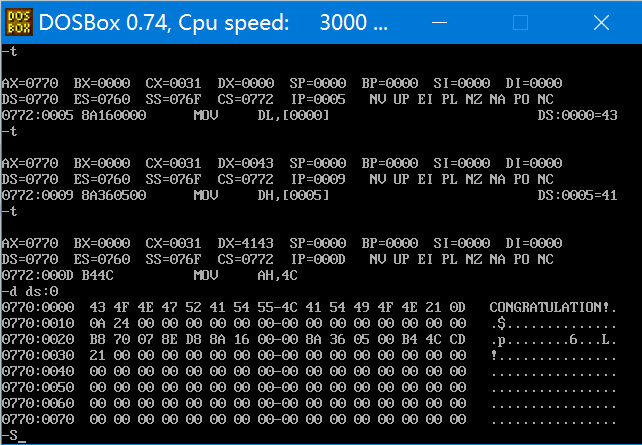
**2.按要求编写程序段**

**（1）在字符串中取出第一个和第六个字符：**

直接寻址，代码：



结果验证：



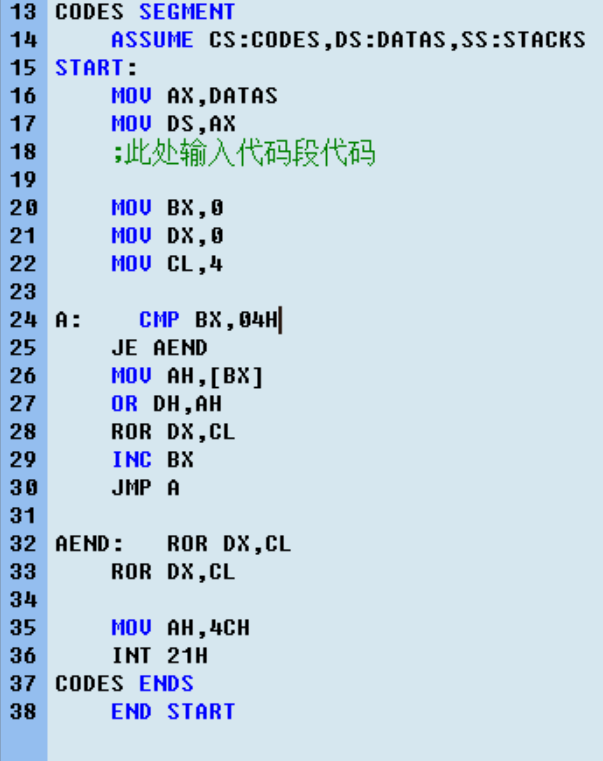
第一个字符C，ASCII为43H，第六个字符A，ASCII为41H，分别存在DL和DH里，结果正确。

**（2）从主存中取4个非压缩BCD码存入DX**

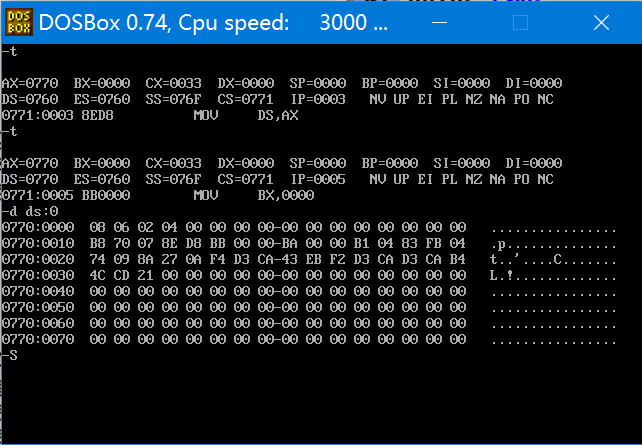
数据段代码：



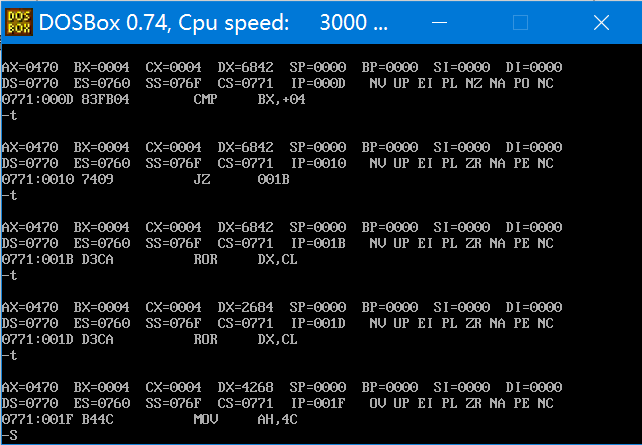
代码段代码：



验证：

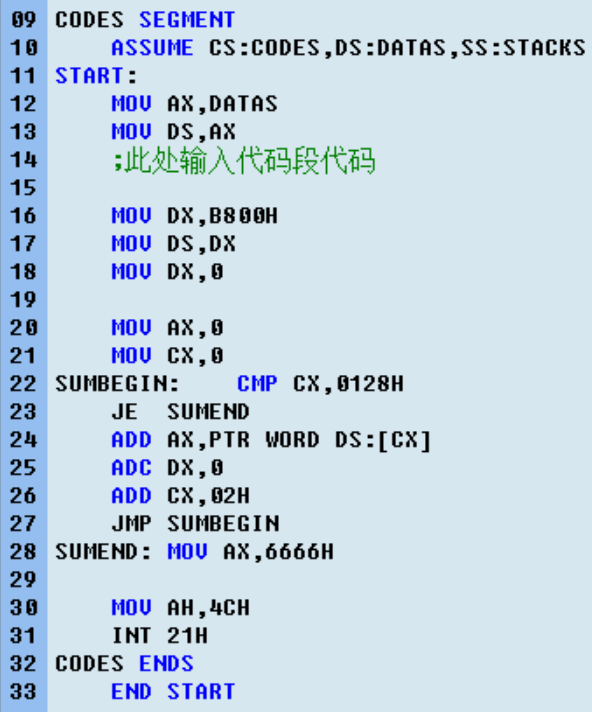


非压缩BCD码依次为8、6、2、4

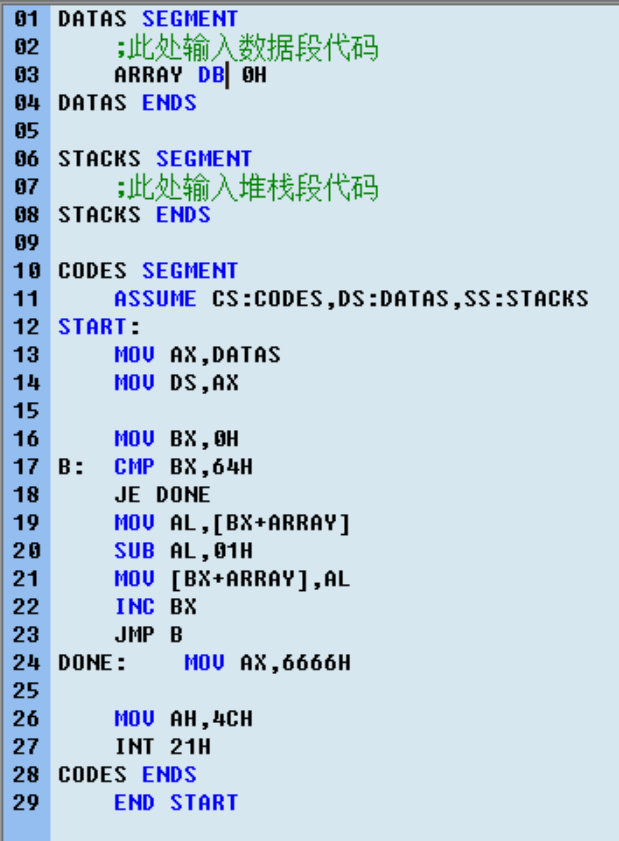


**（3）计算从B800H：0开始的100个16位无符号数和，存入DX.AX**

代码：



**（4）将以array为首地址的100个byte减一**

****

**3.填空**

MOV DX,0AH

MOV CL,4

MOV SI,0

MOV DI,0

CONVERT: MOV AL,[SI+PACKED]

MOV AH,AL

AND AL,0FH

SHR AH,CL

MOV [DI+UNPACKED],AX

ADD DI,2

MOV AX,0

DEC DX

JNZ CONVERT

**4.寻找两数和**

DATAS SEGMENT

DTA DB 02H,07H,0BH,0FH,13H,1CH,24H,39H,40H,57H,68H

TAR DW 79H

DATAS ENDS

STACKS SEGMENT

STACKS ENDS

CODES SEGMENT

ASSUME CS:CODES,DS:DATAS,SS:STACKS

START:

MOV AX,DATAS

MOV DS,AX

MOV BX,0

MOV CX,0AH

LP1: CMP BX,CX

JE NOANS

MOV AX,DS:[0+BX]

MOV DX,DS:[0+CX]

ADD AX,DX

CMP AX,TAR

JE ANS

ADD BX,1

CMP AX,TAR

JB LP1

SUB BX,1

SUB CX,1

CMP AX,TAR

JA LP1

JMP NOANS

NOANS: MOV BX,0

MOV CX,0

MOV AX,0

ANS: MOV AL,BL

MOV AH,CL

MOV AH,4CH

INT 21H

CODES ENDS

END START

# 实验分析与总结

我逐渐掌握了使用JMP系语句进行复杂的程序设计，但LOOP系语句仍不熟练。用汇编语言进行程序设计非常困难，有好多语法限制，需要长期练习。