

全国计算机等级考试



供2006年全年使用

National Computer
Rank Examination
National Computer
Rank Examination
National Computer
Rank Examination
National Computer
Rank Examination

上机考试习题集

三级汇编语言程序设计

National Computer Rank Examination

全国计算机等级考试命题研究组 编

100
NCRE
南开100题

National Computer
Rank Examination
National Computer
Rank Examination

National Computer
Rank Examination
National Computer
Rank Examination

赠送
参考答案
全新光盘

南开大学出版社

全国计算机等级考试

上机考试习题集

三级汇编语言程序设计

(2006)

全国计算机等级考试命题研究组



南开大学出版社

天津

内容提要

本书提供了全国计算机等级考试三级汇编语言机试题库。本书配套光盘中有 3 部分内容: 本书所有习题的详尽答案; 本书所有习题的源文件以及结果文件; 上机考试的全真模拟环境。读者可以利用配套光盘进行考前训练。

本书所有题目均进行了测试, 保证能够在实际环境中正常运行。

本书针对参加全国计算机等级考试三级汇编语言程序设计(即三级 PC 技术)的考生, 同时也可作为大专院校、成人高等教育以及相关培训班的练习题和考试题使用。

图书在版编目(CIP)数据

全国计算机等级考试上机考试习题集. 三级汇编语言程序设计: 2006 / 全国计算机等级考试命题研究组编.

4 版. —天津: 南开大学出版社, 2006. 3

ISBN 7-310-01630-0

I. 全... II. 全... III. ①电子计算机—水平考试—习题②汇编语言—程序设计—水平考试—习题
IV. TP3-44

中国版本图书馆 CIP 数据核字(2005)第 153916 号

版权所有 侵权必究

南开大学出版社出版发行

出版人: 肖占鹏

地址: 天津市南开区卫津路 94 号 邮政编码: 300071

营销部电话: (022)23508339 23500755

营销部传真: (022)23508542 邮购部电话: (022)23502200

*

河北昌黎太阳红彩色印刷有限责任公司印刷

全国各地新华书店经销

*

2006 年 3 月第 4 版 2006 年 3 月第 5 次印刷

787×1092 毫米 16 开本 13 印张 323 千字

定价: 24.00 元

如遇图书印装质量问题, 请与本社营销部联系调换, 电话: (022)23507125

前 言

全国计算机等级考试（National Computer Rank Examination, NCRE）是由教育部考试中心主办，用于考查应试人员的计算机应用知识与能力的考试。本考试的证书已经成为许多单位招聘员工的一个必要条件，具有相当的“含金量”。

为了帮助考生更顺利地通过计算机等级考试，我们做了大量市场调研，根据考生的备考体会，以及培训教师的授课经验，推出了《上机考试习题集——三级汇编语言程序设计》。本书的主要组成有两部分。

一、三级汇编语言程序设计上机考试题库

对于备战等级考试而言，做题，是进行考前冲刺的最佳方式。这是因为它的针对性相当强，考生可以通过实际练习做题，来检验自己是否真正掌握了相关知识点，了解考试重点，并且根据需要再对知识结构的薄弱环节进行强化。

二、配套光盘

本书配套光盘可用于考前实战训练，有 3 部分内容：

- 本书所有习题的详尽答案。
- 本书所有习题的源文件以及结果文件。
- 上机考试的全真模拟环境。

本书针对参加全国计算机等级考试三级汇编语言程序设计（报考三级 PC 技术）的考生，同时也可以作为普通高校、大专院校、成人高等教育以及相关培训班的练习题和考试题使用。

为了保证本书及时面市和内容准确，很多朋友做出了贡献，陈河南、贺民、廖明武、侯佳宜、齐惠颖、贺军、于樊鹏、任世华、田民、许伟、何雄、赵晓睿、戴文雅、戴军、汤效平、陈占军、李季、梁彩隆、黄志雄、李志云、陈安南、李晓春、王春桥、王雷、韦笑、龚亚萍等老师在编写文档、调试程序、排版、查错、预读、光盘制作与测试等工作中加班加点，付出了很多辛苦，在此一并表示感谢！

在学习的过程中，您如有问题或建议，请与我们联系：book_service@126.com，或登录百分网：www.baifen100.com。

全国计算机等级考试命题研究组

2005 年 12 月

★★

第 1 题

请编制程序，其功能是：将内存中由 SOURCE 指示的 40 个字节有符号数组成的数组分成正数和负数两个数组，并求这两个数组的数据个数，结果存放在 RESULT 指示的内存区域。存放形式为正数个数在前，其后跟正数数组元素，然后是负数个数及负数数组元素。

例如：

内存中有 1EH, 91H, 74H, 91H, 42H, 30H, 81H, F3H, 18H, 25H

结果为 06H, 1EH, 74H, 42H, 30H, 18H, 25H, 04H, 91H, 91H, 81H, F3H

部分程序已经给出，其中原始数据由过程 LOAD 从文件 INPUT1.DAT 中读入 SOURCE 开始的内存单元中，转换结果要求从 RESULT 开始存放，由过程 SAVE 保存到文件 OUTPUT1.DAT 中。

请填空 BEGIN 和 END 之间已经给出的一段源程序使其完整，需填空处已经用横线标出，每个空白一般只需要填一条指令或指令的一部分（指令助记符或操作数），也可以填入功能相当的多条指令，或删除 BEGIN 和 END 之间原有的代码并自行编程来完成所要求的功能。

对程序必须进行汇编，并与 IO.OBJ 链接产生可执行文件，最终运行程序产生结果。调试中若发现整个程序中存在错误之处，请加以修改。

试题程序：

```

                                EXTRN     LOAD:FAR,SAVE:FAR
N                                EQU       40

STAC                            SEGMENT   STACK
                                DB         128 DUP (?)
STAC                            ENDS

DATA                            SEGMENT
SOURCE                          DB        N DUP(0)
RESULT                          DB        N+2 DUP(0)
NAME0                           DB        'INPUT1.DAT',0
NAME1                           DB        'OUTPUT1.DAT',0
NDATA                           DB        N DUP(0)
PDATA                           DB        N DUP(0)
DATA                            ENDS

CODE                            SEGMENT
                                ASSUME     CS:CODE, DS:DATA, SS:STAC
START                          PROC       FAR
```

```

        PUSH    DS
        XOR     AX,AX
        PUSH    AX
        MOV     AX,DATA
        MOV     DS,AX
        MOV     ES,AX          ;置附加段寄存器

        LEA     DX,SOURCE      ;数据区起始地址
        LEA     SI,NAME0       ;原始数据文件名
        MOV     CX,N           ;字节数
        CALL    LOAD           ;从'INPUT1.DAT'中读取数据
;      **** BEGIN ****
        LEA     SI, SOURCE
        (1)      DI,OFFSET PDATA ;PDATA 为正数数组存放缓冲区首址
        MOV     BX,OFFSET NDATA ;NDATA 为负数数组存放缓冲区首址
        XOR     DX,DX
        MOV     CX,N
        CLD
MAIN1:    LODSB
        TEST    AL,(2)
        JZ     MAIN2
        INC     DH
        MOV     [BX],AL
        INC     BX
        (3)
MAIN2:    INC     DL
        MOV     [DI],AL
        INC     DI
MAIN3:    (4)      MAIN1
        LEA     SI,PDATA
        LEA     DI,RESULT
        MOV     [DI],DL
        INC     DI
        XOR     CX,CX
        MOV     CL,DL
MAIN4:    MOV     AL,(5)
        MOV     [DI],AL
        INC     DI
        INC     SI

```

```

                                (6)
                                -----
                                MOV     [DI],DH
                                INC     DI
                                XOR     CX,CX
                                MOV     CL,DH
                                MOV     BX,OFFSET NDATA
MAIN5:  MOV     AL,[BX]
                                MOV     [DI],AL
                                INC     DI
                                -----
                                (7)
                                -----
                                LOOP    MAIN5
;      **** END ****
                                LEA     DX,RESULT      ;结果数据区首址
                                LEA     SI,NAME1        ;结果文件名起始地址
                                MOV     CX,N+2          ;字节数
                                CALL    SAVE            ;保存结果到'OUTPUT1.DAT'文件
                                RET
START  ENDP
CODE   ENDS
                                END      START

```

★★

第 2 题

请编制程序，其功能是：内存中连续存放着 10 个无符号 8 位格雷码表示法的数，现将此十个数转换成十个 8 位二进制数，结果存入内存。其转换方法为二进制数的最高位 d_7 与格雷码的最高位 g_7 相同，二进制数的其余七位 $d_k (k=6, \dots, 0)$ 分别为格雷码的位 $g_k (k=6, \dots, 0)$ 与二进制数的位 $d_{k+1} (k=6, \dots, 0)$ 异或的结果。

例如：

内存中有 00H, 03H, 2BH, 67H, 0CH, 15H, 54H, 02H, D8H, C7H

结果为 00H, 02H, 32H, 45H, 08H, 19H, 67H, 03H, 90H, 85H

部分程序已给出，其中原始数据由过程 LOAD 从文件 INPUT1.DAT 中读入 SOURCE 开始的内存单元中。运算结果要求从 RESULT 开始存放，由过程 SAVE 保存到文件 OUTPUT1.DAT 中。

请填空 BEGIN 和 END 之间已给出的源程序使其完整（空白已用横线标出，每个空白一般只需一条指令，但采用功能相当的多条指令亦可），或删除 BEGIN 和 END 之间原有的代码并自行编程来完成所要求的功能。

对程序必须进行汇编，并与 IO.OBJ 链接产生可执行文件，最终运行程序产生结果。调试中若发现整个程序中存在错误之处，请加以修改。

试题程序:

```

        EXTRN     LOAD:FAR,SAVE:FAR
N       EQU       10

STAC    SEGMENT   STACK
        DB        128 DUP (?)
STAC    ENDS

DATA    SEGMENT
SOURCE  DB        N DUP(?)           ;顺序存放 10 个字节数
RESULT  DB        N DUP(0)          ;存放结果
NAME0   DB        'INPUT1.DAT',0
NAME1   DB        'OUTPUT1.DAT',0
DATA    ENDS

CODE    SEGMENT
        ASSUME     CS:CODE, DS:DATA, SS:STAC
START   PROC       FAR
        PUSH       DS
        XOR        AX,AX
        PUSH       AX
        MOV        AX,DATA
        MOV        DS,AX

        LEA        DX,SOURCE        ;数据区起始地址
        LEA        SI,NAME0         ;原始数据文件名
        MOV        CX,N             ;字节数
        CALL       LOAD             ;从'INPUT1.DAT'中读取数据
;      **** BEGIN ****
        LEA        DI, RESULT
        LEA        SI, SOURCE
        MOV        CX,10
AGN0:   MOV        AL,[SI]
        (1)
        MOV        CX,8
        MOV        BX,0
AGN1:   MOV        AH,0
        SHL        BL,1
        (2)        AL,1

```

```

                                RCL      AH,1
                                CMP      AH,____(3)____
                                _____(4)_____
                                JMP      NEXT
SET_ONE:  OR      BL,01H
NEXT:     MOV      _____(5)____,BL
                                _____(6)_____
                                LOOP     AGN1
                                _____(7)_____
                                MOV      [DI],BL
                                INC      SI
                                INC      DI
                                LOOP     AGN0
;      **** END ****
                                LEA      DX,RESULT      ;结果数据区首址
                                LEA      SI,NAME1        ;结果文件名
                                MOV      CX,N            ;结果字节数
                                CALL     SAVE            ;保存结果到文件
                                RET
START    ENDP
CODE     ENDS
                                END      START

```

☆☆

第 3 题

请编制程序，其功能是：内存中连续存放着 20 个十六位二进制无符号数序列，请将它们排成升序（从小到大）。

例如：

内存中有 7001H, 7004H, 7002H…（假设后 17 个字均大于 7004H）

结果为 7001H, 7002H, 7004H…（后跟 17 个字，按从小到大的顺序排列）

部分程序已给出，其中原始数据由过程 LOAD 从文件 INPUT1.DAT 中读入 SOURCE 开始的内存单元中。运算结果要求从 RESULT 开始存放，由过程 SAVE 保存到文件 OUTPUT1.DAT 中。

请填空 BEGIN 和 END 之间已给出的源程序使其完整（空白已用横线标出，每个空白一般只需一条指令，但采用功能相当的多条指令亦可），或删除 BEGIN 和 END 之间原有的代码并自行编程来完成所要求的功能。

对程序必须进行汇编，并与 IO.OBJ 链接产生可执行文件，最终运行程序产生结果。调试中若发现整个程序中存在错误之处，请加以修改。

试题程序:

```

        EXTRN     LOAD:FAR,SAVE:FAR
N       EQU       20

STAC    SEGMENT   STACK
        DB        128 DUP (?)
STAC    ENDS

DATA    SEGMENT
SOURCE  DW        N DUP(?)
RESULT  DW        N DUP(0)
NAME0   DB        'INPUT1.DAT',0
NAME1   DB        'OUTPUT1.DAT',0
DATA    ENDS

CODE    SEGMENT
        ASSUME    CS:CODE, DS:DATA, SS:STAC
START   PROC      FAR
        PUSH      DS
        XOR        AX,AX
        PUSH      AX
        MOV        AX,DATA
        MOV        DS,AX

        LEA        DX,SOURCE      ;数据区起始地址
        LEA        SI,NAME0       ;原始数据文件名
        MOV        CX,N*2         ;字数
        CALL       LOAD           ;从'INPUT1.DAT'中读取数据
;      **** BEGIN ****
        LEA        SI,SOURCE
        LEA        DI,RESULT
        MOV        CX,N
NEXT0:   MOV        AX,[SI]
        MOV        [DI],AX
        ADD        SI, (1)
        _____ (2) _____
        LOOP       _____ (3) _____
        CLD
        MOV        BX,N-1

```

```

MAL1:    LEA        SI,RESULT
          MOV        CX,____(4)____
NEXT:    LOD        ____ (5) ____
          CMP        [SI],AX
          JAE        CONT
          XCHG       [SI],____(6)____
          MOV        [SI-2],AX
CONT:    LOOP       ____ (7) ____
          ____ (8) ____
          ____ (9) ____    MAL1
;    **** END ****
          LEA        DX,RESULT    ;结果数据区首址
          LEA        SI,NAME1     ;结果文件名
          MOV        CX,N*2       ;结果字数
          CALL       SAVE         ;保存结果到文件
          RET
START    ENDP
CODE     ENDS
          END        START

```

★★

第 4 题

请编制程序，其功能是：内存中连续存放着 24 个无符号二进制字序列，字的最高 3 位为 000，此序列对应某一信号在一段时间内的连续变化，现对第 21 个二进制字前的 20 个二进制字进行移动平均处理，其方法为：将要处理的字 X_i 用以它为开始的连续五个字的平均数 $(X_i + X_{i+1} + X_{i+2} + X_{i+3} + X_{i+4})/5$ 代替（余数舍去），得到新的 20 个无符号二进制字序列，结果存入内存。

例如：

内存中有 0100H, 0200H, 0300H, 0400H, 0500H, 0600H...

结果 0300H, 0400H...

部分程序已经给出，其中原始数据由过程 LOAD 从文件 INPUT1.DAT 中读入 SOURCE 开始的内存单元中，转换结果要求从 RESULT 开始存放，由过程 SAVE 保存到文件 OUTPUT1.DAT 中。

请填空 BEGIN 和 END 之间已经给出的一段源程序使其完整，需填空处已经用横线标出，每个空白一般只需填一条指令或指令的一部分（指令助记符或操作数），也可以填入功能相当的多条指令，或删除 BEGIN 和 END 之间原有的代码并自行编程来完成所要求的功能。

对程序必须进行汇编，并与 IO.OBJ 链接产生可执行文件，最终运行程序产生结果。调试中若发现整个程序中存在错误之处，请加以修改。

试题程序:

```

        EXTRN     LOAD:FAR,SAVE:FAR
N       EQU       24

STAC    SEGMENT   STACK
        DB        128 DUP (?)
STAC    ENDS

DATA    SEGMENT
SOURCE  DW        N DUP(?)           ;顺序存放 24 个字
RESULT  DW        20 DUP(0)         ;存放结果
NAME0   DB        'INPUT1.DAT',0
NAME1   DB        'OUTPUT1.DAT',0
DATA    ENDS

CODE    SEGMENT
        ASSUME    CS:CODE, DS:DATA, SS:STAC
START   PROC      FAR
        PUSH      DS
        XOR       AX,AX
        PUSH      AX
        MOV       AX,DATA
        MOV       DS,AX

        LEA       DX,SOURCE         ;数据区起始地址
        LEA       SI,NAME0          ;原始数据文件名
        MOV       CX,N*2            ;字节数
        CALL      LOAD              ;从'INPUT1.DAT'中读取数据
;      ***** BEGIN *****
        MOV       DI,0
        MOV       SI,0
        MOV       CX,20
        MOV       BX,_____(1)_____
AGN0:   MOV       AX,SOURCE[SI]
        PUSH      SI
        PUSH      CX
        MOV       _____(2)_____, _____(3)_____
AGN1:   INC       SI
        INC       SI

```

```

        ADD     AX,SOURCE[SI]
        LOOP    AGN1
        _____
                (4)
        DIV     BX
        MOV     RESULT[DI],AX
        INC     DI
        _____
                (5)
        POP     CX
        POP     SI
        INC     SI
        _____
                (6)
        LOOP    AGN0
;    **** END ****
        LEA     DX,RESULT      ;结果数据区首址
        LEA     SI,NAME1       ;结果文件名
        MOV     CX,40          ;结果字节数
        CALL    SAVE           ;保存结果到文件
        RET
START    ENDP
CODE     ENDS
        END      START

```

★★

第 5 题

请编制程序，其功能是：内存中从 SOURCE 开始连续存放着 21 个八位有符号数（补码），其相邻两数之间差值不超过 -8 至 7。对这种变化缓慢的数据可采用差分方法进行压缩。即第一个数据不变，其后的数据取与前一数据的差值并用四位二进制补码表示，两个差值拼成一个字节，前一个差值放在高四位，后一个差值放在低四位。

例如：

原数据 (X[n]): 23H, 27H, 2AH, 29H, 22H...

压缩后 (Y[n]): 23H, 43H, F9H...

编程按上述方法进行压缩，结果保存在 RESULT 开始的内存单元中。

部分程序已给出，请填空 BEGIN 和 END 之间已给出的源程序使其完整（空白已用横线标出，每个空白一般只需一条指令，但采用功能相当的多条指令亦可），或删除 BEGIN 和 END 之间原有的代码并自行编写程序片段来完成所要求的功能。

原始数据由过程 LOAD 从文件 INPUT1.DAT 中读入 SOURCE 开始的内存单元中，结果要求从 RESULT 开始存放，由过程 SAVE 保存到文件 OUTPUT1.DAT 中。

对程序必须进行汇编，并与 IO.OBJ 链接产生可执行文件，最终运行程序产生结果。调

试中若发现整个程序中存在错误之处，请加以修改。

试题程序：

```

        EXTRN    LOAD:FAR,SAVE:FAR
N       EQU      10

STAC    SEGMENT STACK
        DB       128 DUP (?)
STAC    ENDS

DATA    SEGMENT
SOURCE  DB       2*N+1  DUP(?)
RESULT  DB       N+1    DUP(0)
NAME0   DB       'INPUT1.DAT',0
NAME1   DB       'OUTPUT1.DAT',0
DATA    ENDS

CODE    SEGMENT
        ASSUME   CS:CODE, DS:DATA, SS:STAC
START   PROC     FAR
        PUSH     DS
        XOR      AX,AX
        PUSH     AX
        MOV      AX,DATA
        MOV      DS,AX
        MOV      ES,AX          ;置附加段寄存器
        LEA      DX,SOURCE      ;数据区起始地址
        LEA      SI,NAME0       ;原始数据文件名起始地址
        MOV      CX,2*N+1       ;字节数
        CALL     LOAD           ;从'INPUT1.DAT'中读取数据
;      **** BEGIN ****
        LEA      SI,SOURCE
        LEA      DI,RESULT
        CLD
        MOVSB                    ;Y[0]=X[0]
        XOR      BX,BX          ;FLAG=0
        MOV      DX,N*2         ;COUNTER
        MOV      CL,4
COMPRESS:

```

```

        LODSB
        SUB  (1)          ;X[n]-X[n-1]
        (2)          ;FLAG=NOT FLAG
        J  (3)      LOW_HEX
        (4)      AL,CL
        MOV      AH,AL
        JMP      NEXT
LOW_HEX:
        (5)
        OR      AL,AH
        STOSB
NEXT:    (6)
        JNE      COMPRESS
;      **** END ****
        LEA      DX,RESULT      ;结果数据区首址
        LEA      SI,NAME1      ;结果文件名起始地址
        MOV      CX,N+1        ;字节数
        CALL     SAVE          ;保存结果到'OUTPUT1.DAT'文件中
        RET
START   ENDP
CODE    ENDS
        END      START

```

★★

第 6 题

请编制程序，其功能是：内存中存放着 20 个 0~9 之间的数字的 ASCII 字符（包括数字 0 和 9 的 ASCII 字符）或 SP 字符（20H），请将 0~9 之间的数字的 ASCII 字符（包括数字 0 和 9 的 ASCII 字符）转换为相应的八位二进制数，并将 SP 字符转换为\$字符（24H）。将按上述方法处理后得到的 20 个字节存入内存中。

例如：

内存中有 20H ('SP'), 30H ('0'), 31H ('1'), 31H ('1'), 31H ('1'), 20H ('SP'),
32H ('2'), ..., 39H ('9') (共 20 个 ASCII 字符)

结果为 24H ('\$'), 00H, 01H, 01H, 01H, 24H ('\$'), 02H ('2'), ..., 09H ('9')
(共 20 个字节)

部分程序已给出，其中原始数据由过程 LOAD 从文件 INPUT1.DAT 中读入 SOURCE 开始的内存单元中。运算结果要求从 RESULT 开始存放，由过程 SAVE 保存到文件 OUTPUT1.DAT 中。

请填空 BEGIN 和 END 之间已给出的源程序使其完整（空白已用横线标出，每个空白

一般只需一条指令,但采用功能相当的多条指令亦可),或删除 BEGIN 和 END 之间原有的代码并自行编程来完成所要求的功能。

对程序必须进行汇编,并与 IO.OBJ 链接产生可执行文件,最终运行程序产生结果。调试中若发现整个程序中存在错误之处,请加以修改。

试题程序:

```

        EXTRN     LOAD:FAR,SAVE:FAR
N       EQU       20

STAC    SEGMENT   STACK
        DB        128 DUP (?)
STAC    ENDS

DATA    SEGMENT
SOURCE  DB        N DUP(?)
RESULT  DB        N DUP(0)
NAME0   DB        'INPUT1.DAT',0
NAME1   DB        'OUTPUT1.DAT',0
DATA    ENDS

CODE    SEGMENT
        ASSUME    CS:CODE, DS:DATA, SS:STAC
START   PROC      FAR
        PUSH     DS
        XOR      AX,AX
        PUSH     AX
        MOV      AX,DATA
        MOV      DS,AX

        LEA      DX,SOURCE      ;数据区起始地址
        LEA      SI,NAME0       ;原始数据文件名
        MOV      CX,N           ;字节数
        CALL     LOAD           ;从'INPUT1.DAT'中读取数据
;      **** BEGIN ****
        (1)
        MOV      DI,0
        (2)
CHAN:   (3)
        SUB      AL,20H
        JZ       CHANGE

```

```

SUB      AL, (4)
MOV      RESULT[DI],AL
(5)
(6)
DEC      CX
JZ       EXIT
JMP      CHAN
CHANGE:  MOV      RESULT[DI],24H
INC      DI
INC      SI
JMP      (7)
EXIT:    NOP
;      **** END ****
LEA      DX,RESULT      ;结果数据区首址
LEA      SI,NAME1       ;结果文件名
MOV      CX,N           ;结果字节数
CALL     SAVE           ;保存结果到文件
RET
START    ENDP
CODE     ENDS
END      START

```

★★

第 7 题

请编制程序，其功能是：内存中共有 30 个字节型数据，找出其中的两个 ASCII 字符串并进行校验。欲寻找的 ASCII 字符串由 13 个字符组成：#(23H)，7 个 ASCII 字符，*(2AH)，2 个 ASCII 字符，回车符 (0DH)，换行符 (0AH)。

校验方法为：对字符#及字符*之间的 7 个 ASCII 字符进行异或操作，若异或操作结果的 ASCII 字符表示（异或操作结果高 4 位的 ASCII 字符表示在前，低 4 位的 ASCII 字符表示在后）与原字符串中字符*之后的两个 ASCII 字符相同，则将原字符串原样保存；反之，则将原字符串中的所有字符（共 13 个）均用字符!(21H)代替。

例如：

内存中有 33H, 35H, 23H ('#', 第一个字符串开始), 46H, 41H, 30H, 2EH, 34H, 3DH, 31H, 2AH (*), 32H, 31H, 0DH, 0AH, 46H, 23H ('#', 第二个字符串开始), 46H, 41H, 30H, 2EH, 34H, 3DH, 30H, 2AH (*), 32H, 31H, 0DH, 0AH, 55H

结果为 23H, 46H, 41H, 30H, 2EH, 34H, 3DH, 31H, 2AH (*), 32H, 31H (校验正确，该字符串原样保持), 0DH, 0AH, 21H, 21H, 21H, 21H,