**西安电子科技大学**

**微机原理综合实验 课程实验报告**

**实验名称 顺序、分支、循环、子程序设计**

xxxxxxxxxxxxx 学院 xxxxxxx 班

成 绩

姓名 xxxxxx 学号 xxxxxxxxxxx

同作者 无

实验日期 2021 年 5 月 28 日

|  |
| --- |
| 指导教师评语：  指导教师：  年 月 日 |
| **实验报告内容基本要求及参考格式**  一、实验目的  二、实验所用仪器（或实验环境）  三、实验基本原理及步骤（或方案设计及理论计算）  四、实验数据记录（或仿真及软件设计）  五、实验结果分析及回答问题（或测试环境及测试结果） |

1. **实验目的**

学习顺序、分支、循环结构以及子程序的编程设计。

1. **实验内容**

1．编写Q=a+(b-c)\*d的程序。（a、b、c、d为0-9间的正整数）要求：a、b、c、d均从键盘输入，计算结果在虚拟终端上显示。

2．在DS数据段偏移地址为DATA开始的顺序80个单元中，存放着某班40个同学的微机原理考试成绩。请编程统计>=90分，80分-89分，70分-79分，60分-69分以及<60分的人数，将统计结果放在当前数据段偏移地址为BUFFER的顺序单元中。并在虚拟终端上显示各分数段的人数。

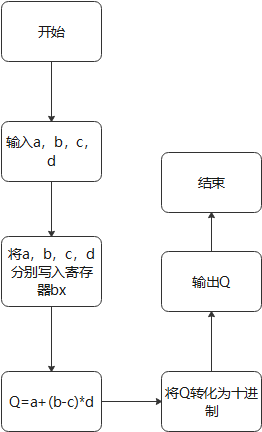
3．从键盘上输入1个数（不大于5位，以回车键结束），判断其奇偶性，如果是奇数，屏幕上显示”It is odd”，否则显示”It is even”。

4．从键盘上输入N个字符（N<16），求这N个字符中’A’的个数，并将’A’的个数显示在屏幕上。

5. 已知BUF1中有N1个按从小到大的顺序排列的互不相等的无符号数，BUF2中有N2个从小到大的顺序排列的互不相等的无符号数。编写程序将BUF1和BUF2中的数合并到BUF3中，使在BUF3中存放的数互不相等且按从小到大的顺序排列。

1. **程序清单及运行结果**

**1.**



data segment

string1 db 'number a:$'

string2 db 'number b:$'

string3 db 'number c:$'

string4 db 'number d:$'

string5 db 'a+(b-c)\*d=$'

a dw 0

b dw 0

c dw 0

d dw 0

e dw 0 ;结果存在e

data ends

code segment

main proc

assume cs:code,ds:data

start:

mov ax,data

mov ds,ax

lea dx,string1

mov ah,9h

int 21h

call input ;输入a

mov a,bx

lea dx,string2

mov ah,9h

int 21h

call input ;输入b

mov b,bx

lea dx,string3

mov ah,9h

int 21h

call input ;输入c

mov c,bx

lea dx,string4

mov ah,9h

int 21h

call input ;输入d

mov d,bx

lea dx,string5

mov ah,9h

int 21h ;输出结果

;计算过程

mov ax,b

sub ax,c

mov bx,ax

mov ax,d

;计算（b-c）\*d

imul bx ;这里是16位，8位乘以8位,其中一个乘数默认放在AL寄存器中,乘法的结果放在AX寄存器之中

adc ax,a

mov e,ax

;输出结果

test ax,8000h

jz printe2 ;正数

jnz printe1 ;负数

printe1:

mov dl,'-'

mov ah,02h ;显示输出。入口参数：AH=02H;功能号 DL=要显示的字符

int 21h

mov ax,e

neg ax

mov bl,0ah

div bl

mov bh,ah

mov dl,al

add dl,30H

mov ah,02h ;显示输出。入口参数：AH=02H;功能号 DL=要显示的字符

int 21h

mov dl,bh

add dl,30H

mov ah,02h ;显示输出。入口参数：AH=02H;功能号 DL=要显示的字符

int 21h

MOV AX,4C00H ;程序中止

INT 21H

printe2:

mov ax,e

mov bl,0xah

div bl

mov bh,ah

mov dl,al

add dl,30H

mov ah,02h ;显示输出。入口参数：AH=02H;功能号 DL=要显示的字符

int 21h

mov dl,bh

add dl,30H

mov ah,02h ;显示输出。入口参数：AH=02H;功能号 DL=要显示的字符

int 21h

MOV AX,4C00H ;程序中止

INT 21H

main endp

;call 子程序

input proc

mov ah,01h;输入一个字符，AL等于输入的字符

int 21h

sub al,30h ;转换为十进制

mov bl,al ;将al转移到bl中

and bx,00FFh ;将bx高位清零

jmp exit

exit:

call newline

ret

input endp

;换行操作

newline proc

mov dl,0ah;换行

mov ah,2h

int 21h

;显示输出，dl为要显示的字符

mov dl,0dh;回车

mov ah,2h

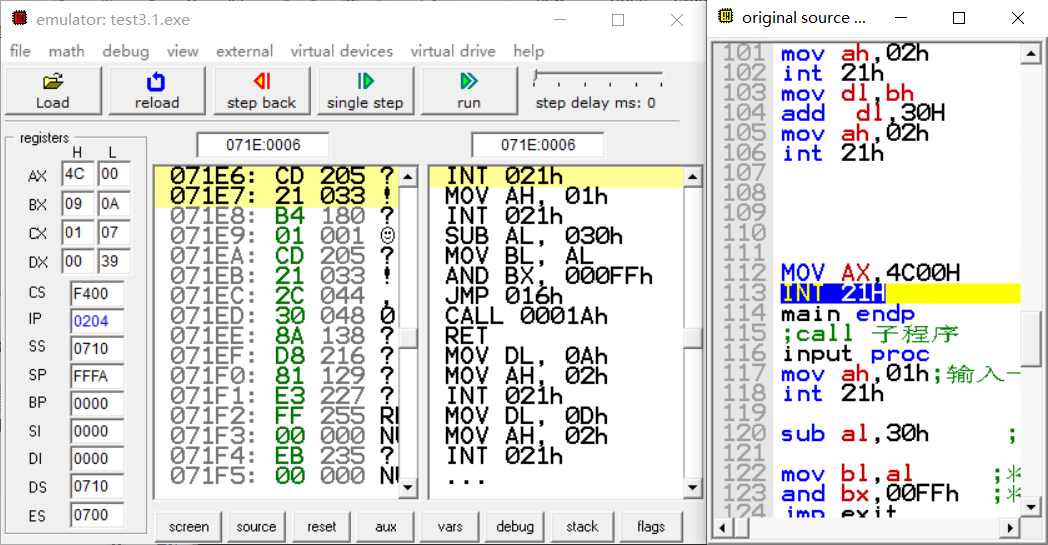
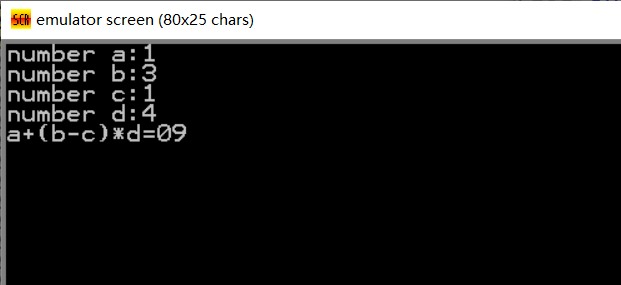
int 21h

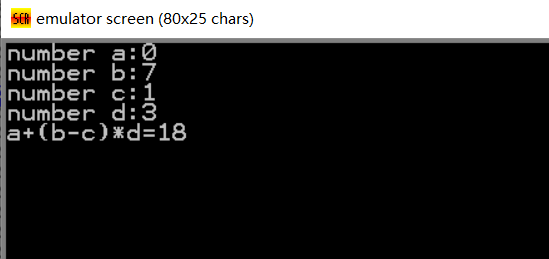
ret

newline endp

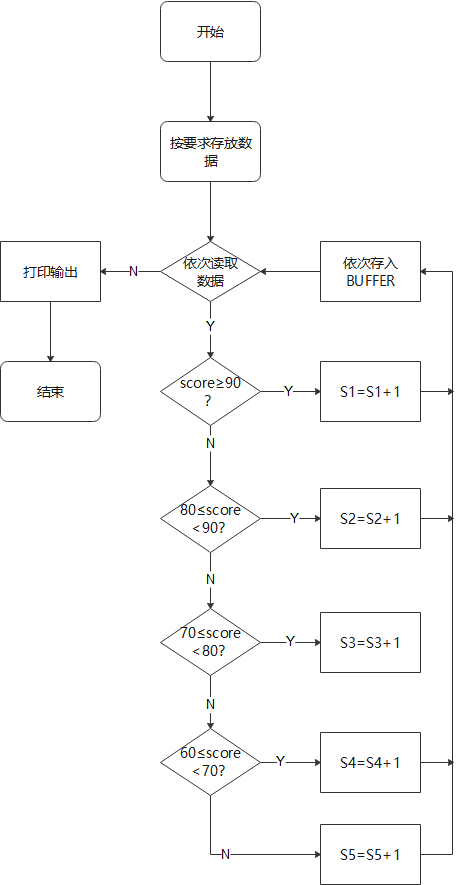
code ends

end start



**2.**



data segment

score db 67,91,84,86,72,85,58,98,66,54,77,88,69,60,48,64,40,81,79,49,65,87,66,88,98,100,66,58,96,76,85,81,82,87,78,49,68,75,48,95

buffer 5 dup(0)

str0 db " 40 scores : $"

str1 db " >=90 : $"

str2 db " 80-89 : $"

str3 db " 70-79: $"

str4 db " 60-69 : $"

str5 db " <60 : $"

data ends

code segment

assume CS:code,DS:data

main:

mov ax,data

mov ds,ax

mov dx,0000h

mov bx,0000h

mov bp,0000h

mov cx,40

mov si,offset score

mov di,offset buffer

nine:

mov al,[si]

cmp al,90

jc eight

inc dh

jmp stor

eight:

cmp al,80

jc seven

inc dl

jmp stor

seven:

cmp al,70

jc six

inc bh

jmp stor

six:

cmp al,60

jc five

inc bl

jmp stor

five:

inc bp

stor:

inc si

loop nine

mov [di],dh

mov [di+1],dl

mov [di+2],bh

mov [di+3],bl

mov [di+4],bp

lea dx,str0

mov ah,09h

int 21h

mov cx,40

lea si,score

show:

mov ax,0

mov al,[si]

call printf

call space

inc si

loop show

call crlf

lea dx,str1

mov ah,09h

int 21h

mov ax,0

mov al,[di]

call printf

call crlf

lea dx,str2

mov ah,09h

int 21h

mov ax,0

mov al,[di+1]

call printf

call crlf

lea dx,str3

mov ah,09h

int 21h

mov ax,0

mov al,[di+2]

call printf

call crlf

lea dx,str4

mov ah,09h

int 21h

mov ax,0

mov al,[di+3]

call printf

call crlf

lea dx,str5

mov ah,09h

int 21h

mov ax,0

mov al,[di+4]

call printf

call crlf

mov ax,4c00h

int 21h

printf proc

push cx

push bx

push dx

mov cx,0

mov bx,10

disp1:

mov dx,0

div bx

push dx

inc cx

or ax,ax

jne disp1

disp2:

mov ah,2

pop dx

add dl,30H

int 21h

loop disp2

pop dx

pop bx

pop cx

ret

printf endp

;换行

crlf proc near

push ax

push dx

mov dl,0ah

mov ah,2h

int 21h

mov dl,0dh

mov ah,2h

int 21h

pop dx

pop ax

ret

crlf endp

space proc

push ax

push dx

mov dl,20h

mov ah,2h

int 21h

pop dx

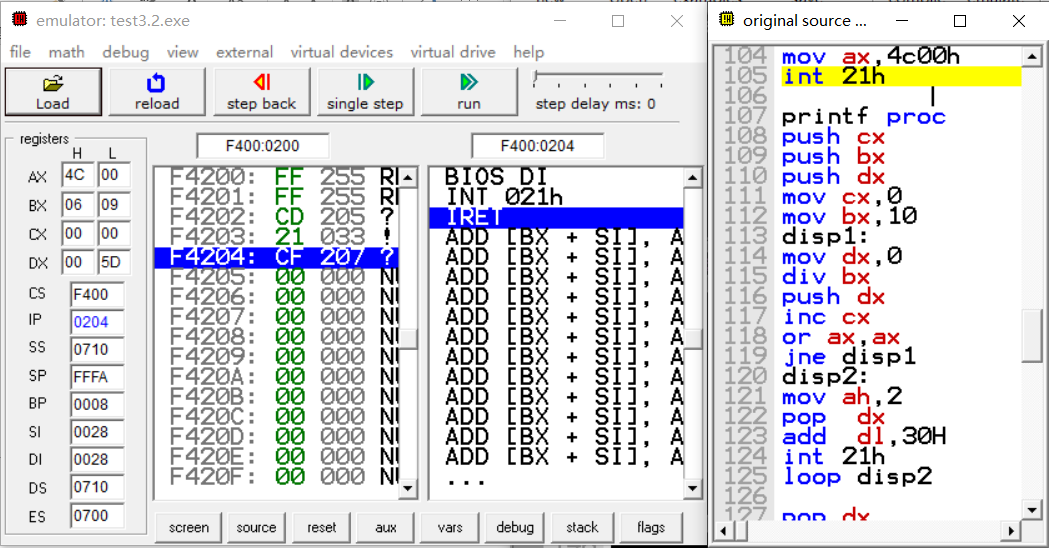
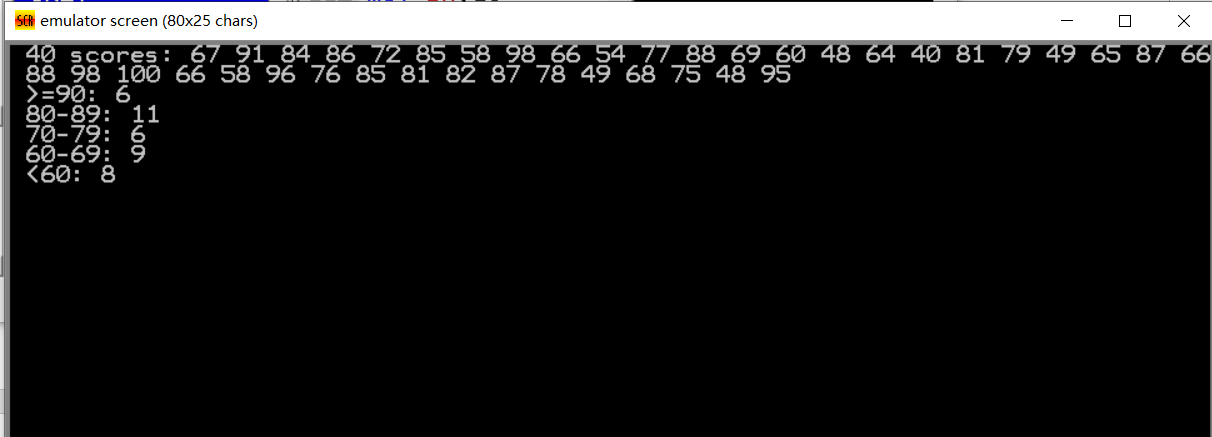
pop ax

ret

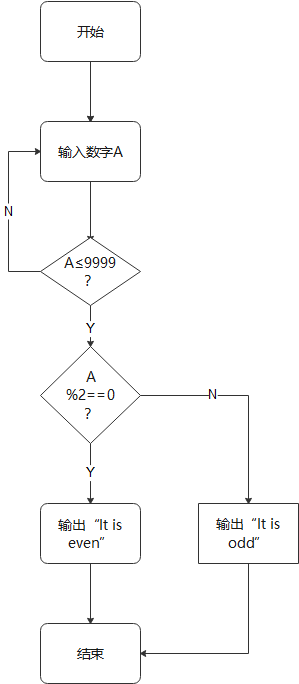
space endp

code ends

end main

**3.**



data segment

odd db 'It is odd$'

even db 'It is even$'

data ends

stack segment

dw 64 dup(0)

ends

code segment

start:

mov ax,data

mov ds,ax

mov bx,0

input1:

mov ah,01h

int 21h ;键盘输入一个数

cmp al,0dh

jz select ;数据写入后进行奇数偶数判断

sub al,30h

mov cl,al

mov ch,00

mov ax,bx

mov bx,0ah

mul bx

mov bx,ax

add bx,cx

jmp input1

select:

mov bx,0

mov ah,2

mov dl,0ah ;回车后进行判断

int 21h

mov dl,0dh

int 21h

sub ah,ah

mov al,cl

mov bl,2

div bl

cmp ah,1 ;ah存余数

jnc next1

jmp next2

next1: ;打印结果是奇数

lea dx, odd

mov ah, 9

int 21h

mov ah,4ch

int 21h

next2: ;打印结果是偶数

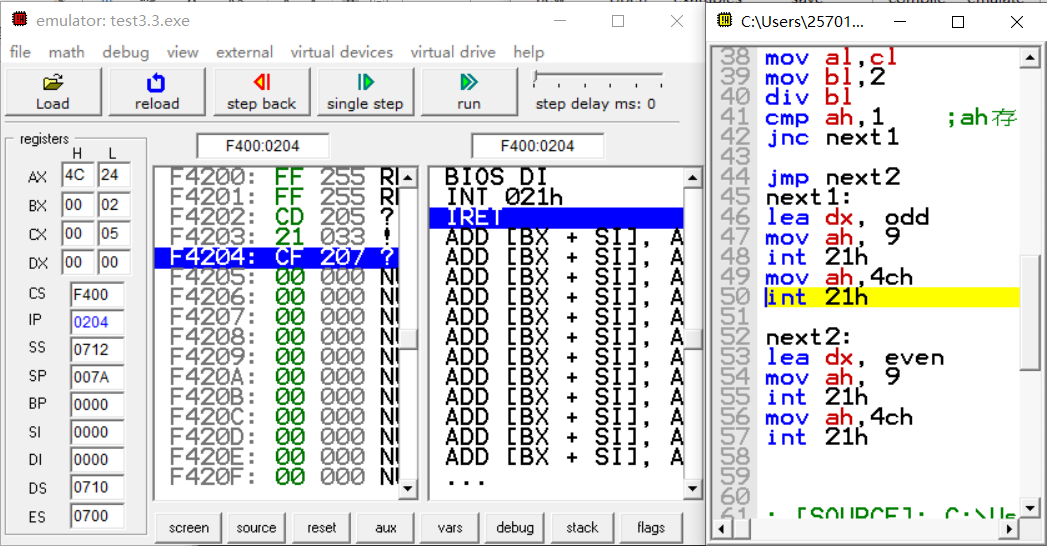
lea dx, even

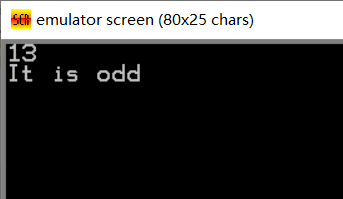
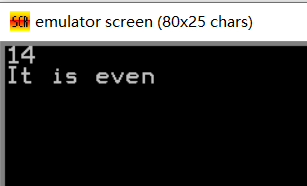
mov ah, 9

int 21h

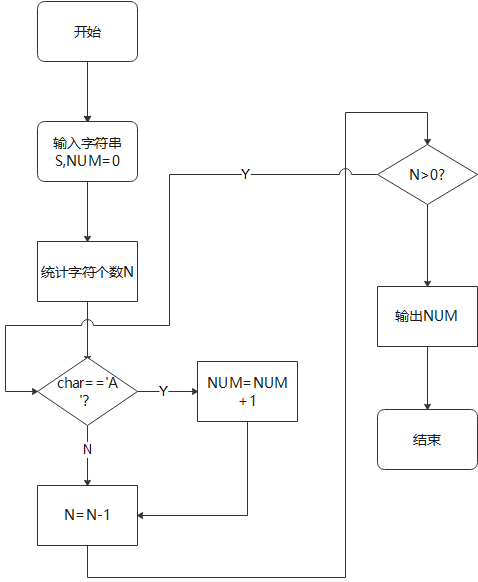
mov ah,4ch

int 21h



**4.**



stack segment

dw 120 dup(0)

ends

code segment

start:

mov ds, ax

mov bx, 0

mov dl, 0

input:

mov ah, 01h

int 21h

cmp al, 0dh ;

jz write

cmp al, 'A'

je next

jmp input

write:

mov ah, 2

mov dl, 0ah ;换行

int 21h

mov dl, 0dh ;回车

int 21h

mov ax, bx

mov dl, 10 ;做除法

div dl

mov dl, al ;余数

mov bl, ah ;商

mov ah, 2

add dl, 30h

int 21h

mov dl, bl

mov ah, 2

add dl, 30h

int 21h

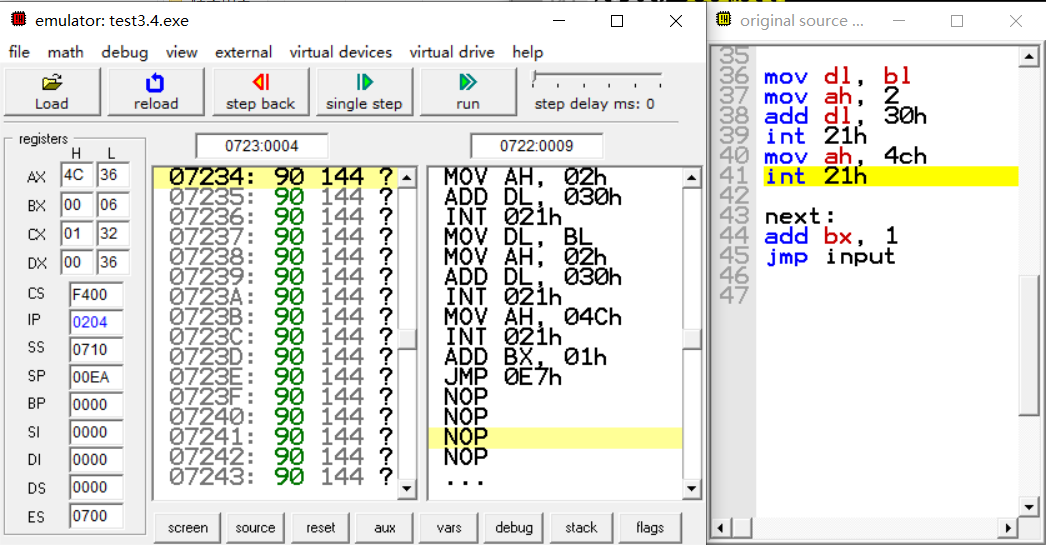
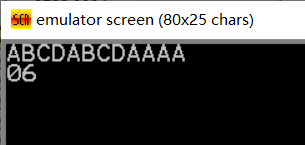
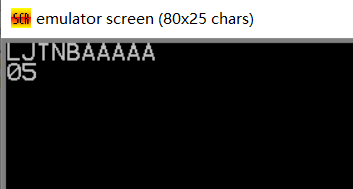
mov ah, 4ch

int 21h

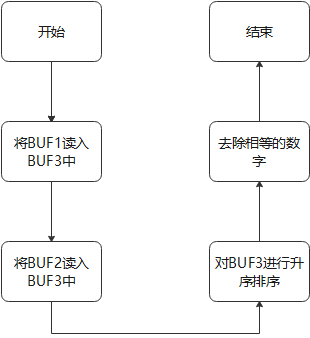
next:

add bx, 1

jmp input

**5.**



data segment

BUF1 db 1,2,3,4,5,6

BUF2 db 5,6,7,8,9

BUF3 db 32 dup(204)

N1 db 6

N2 db 5

string db 'Array merged completed!$'

data ends

code segment

ASSUME CS:CODE,DS:DATE

start:

mov ax,data

mov ds,ax

lea si,BUF1 ;pointer of buffer1

lea di,BUF2 ;pointer of buffer2

lea bx,BUF3 ;pointer of buffer3

mov ax,0

mov cx,0

mov dx,0

mov ah,N1 ;number of N1

mov al,N2 ;number of N2

mov cl,N1

add cl,N2 ;number of N1+N2

mov dx,word ptr N2 ;number of N1+N2-1

dec dx

mov dh,0

add dl,ah

mov ax,word ptr N1 ;number of N1-1

dec ax

mov ah,0

while:

cmp si,ax

ja b2 ;大于跳转

cmp di,dx

ja b1

push dx

mov dx,[di]

cmp [si],dl

pop dx

ja b2 ;buf1[si] > buf2[di]

jb b1 ;buf1[si] < buf2[di]

push dx

push ax

mov dx,[si]

mov [bx],dl ;buf1[si] == buf2[di]

add dl,30h

mov ah,2h

int 21h

pop ax

pop dx

inc si

inc di

inc bx

dec cx

jmp tail

b2:

push dx

push ax

mov dx,[di]

mov [bx],dl

add dl,30h

mov ah,2h

int 21h

pop ax

pop dx

inc bx

inc di

jmp tail

b1:

push dx

push ax

mov dx,[si]

mov [bx],dl

add dl,30h

mov ah,2h

int 21h

pop ax

pop dx

inc bx

inc si

tail:

loop while

call crlf

lea dx,string

mov ah,09h

int 21h

mov ax,4c00h

int 21h

crlf proc near

push dx

mov dl,0ah

mov ah,2h

int 21h

mov dl,0dh

mov ah,2h

int 21h

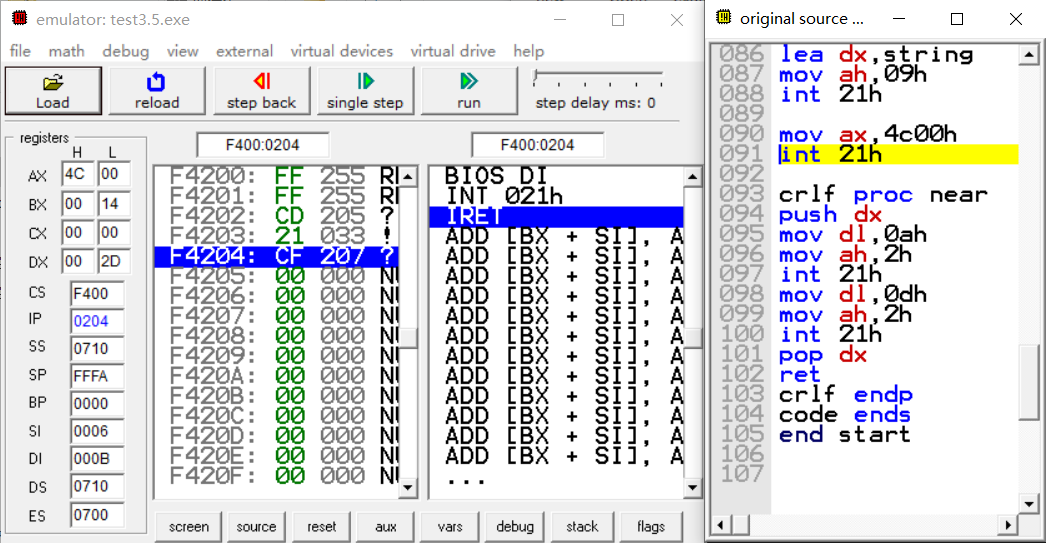
pop dx

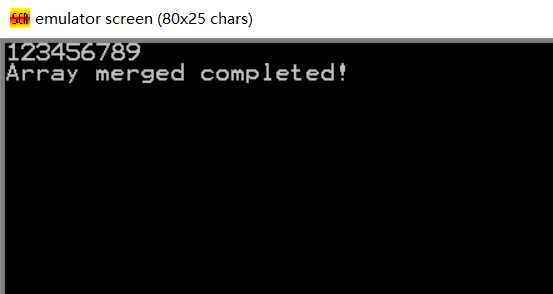
ret

crlf endp

code ends

end start





1. **心得体会**

相比于前几次实验，这次实验明显对于代码要求更高了，这就要求我思维必须更加活跃严谨，不过幸好问题属于比较中规中矩，解题的思路不是很难。本次实验是对我能力一次很重要的提升。