**电工电子实验中心**

**实验报告**

课程名称： 微机原理与接口技术实验

实验名称： 数据统计

姓名： 李应飞 学号: 161610338

评定成绩： 审阅教师： 卓然

实验时间： 2018.10.30

南京航空航天大学

# 实验目的要求

1. 熟悉汇编语言程序的框架结构，掌握循环程序的设计方法。
2. 熟悉常用的条件跳转指令。
3. 熟悉有符号数的运算。

# 实验任务

本实验要求通过求某数据区内负数的个数来表现循环程序的结构形式。要求实验程序在数据段中存放一组数据，分类统计数据中正数、负数和零的个数，并分别存入内存变量 Positive、Negative 和 Zero 中。将所有数据累加求和，存入 SUM 中。

# 实验代码（写出自己补全的代码，包含适当注释）

MOV SI,OFFSET NUM

MOV CX,N ;总数

CLD ;清除方向标志

MOV BX,0

NEXT0:

LODSB ;扫描字节 将si指向的一个字节放入al中

CBW ; 字节转换为字

ADD BX,AX

CMP AL,0

JG NEXT1

JL NEXT2

INC ZERO

JMP NEXT4

NEXT1:

INC POSITIVE

JMP NEXT4

NEXT2:

INC NEGATIVE

NEXT4:

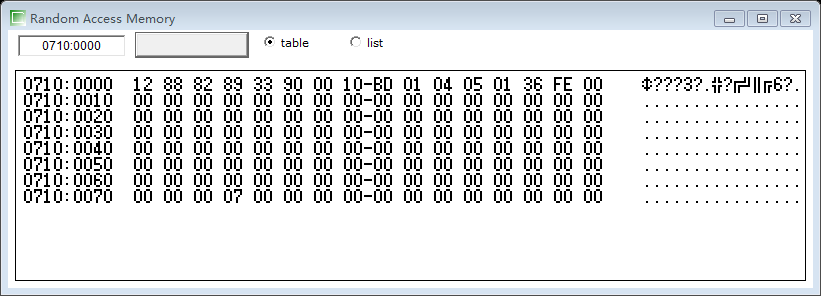
LOOP NEXT0

MOV SUM,BX

# 实验的运行数据及分析

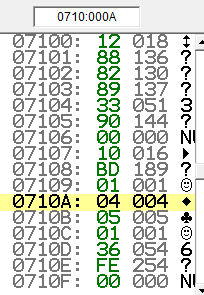
1. 实验数据记录

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| DS:0000 | 12 | 88 | 82 | 89 | 33 | 90 | 00 | 10 |
| DS:0008 | BD | 01 | 04 | 05 | 01 | 36 | FE | 00 |

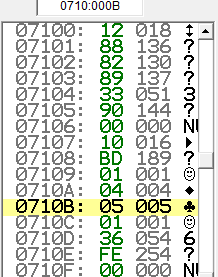


黄色的部分为结果

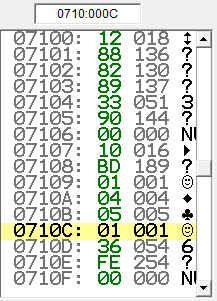
POSITIVE的结果：正数有4个



NEGATIVE的结果：负数有5个



ZERO的结果：有一个0



SUM的结果：所有数的和FE36，没有区分有无符号

![E:\qq下载\MobileFile\Image\WB8KDZP](]TQ88KVL26MX8V.png](data:image/png;base64,)

# 探究内容（选做）

1. 若需要将程序得到的结果显示在屏幕上，如何处理？

1.在数据段加入提示信息

DATA SEGMENT

NUM DB 12H,88H,82H,89H,33H,90H,0H,10H,0BDH,01H

N EQU $-NUM

POSITIVE DB 0

NEGATIVE DB 0

ZERO DB 0

SUM DW 0

NOTE DB "POSITIVE:",0DH,0AH,'$'

NOTE1 DB "NEGATIVE:",0DH,0AH,'$'

NOTE2 DB "ZERO:",0DH,0AH,'$'

NOTE3 DB "SUM:",0DH,0AH,'$'

DATA ENDS

2.然后利用DOS提供的功能，写子程序，调用

MOV DX,OFFSET NOTE3

MOV AH,9 ;输出字符串

INT 21H

MOV AX，SUM

CALL DSPAXS

MOV DX,OFFSET NOTE

MOV AH,9

INT 21H

MOV AL,POSITIVE

MOV AH,0

CALL DSPAXS

MOV DX,OFFSET NOTE1

MOV AH,9

INT 21H

MOV AL,NEGATIVE

MOV AH,0

CALL DSPAXS

MOV DX,OFFSET NOTE2

MOV AH,9

INT 21H

MOV AL,ZERO

MOV AH,0

CALL DSPAXS

POP AX

POP DS

; RET

START ENDP

;============================

DSPAXS PROC NEAR

PUSH AX

TEST AX,8000H ;影响SF标志

JNS @DSPAXS1;为正转移SF

PUSH AX

PUSH DX

MOV AH,2 ;输出负号

MOV DL,'-'

INT 21H

POP DX

POP AX

NEG AX

@DSPAXS1:

CALL DSPAX

POP AX

RET

DSPAXS ENDP

;============================

DSPAX PROC NEAR

PUSH AX

PUSH BX

PUSH CX

PUSH DX

PUSHF

XOR CX,CX

MOV BX,10

@DSPAX1:

XOR DX,DX

DIV BX ;商在AX 余数在DX

INC CX

OR DX,30H;转换为字符，相当于加‘0’

PUSH DX ;不断入栈

CMP AX,0

JNE @DSPAX1

MOV AH,2 ;不断出栈 然后输出

@DISPAX2:

POP DX

INT 21H

LOOP @DISPAX2

MOV DX,0DH;

INT 21H

MOV DX,0AH;

INT 21H

POPF

POP DX

POP CX

POP BX

POP AX

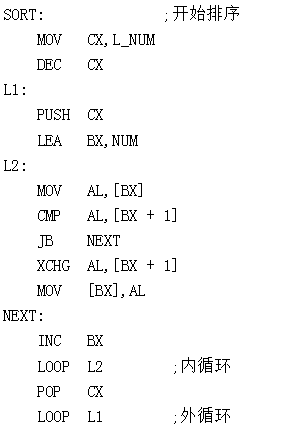
RET

DSPAX ENDP

2）利用某种排序算法，对原始数据进行排序。

冒泡排序

L\_NUM是数组的长度



# 心得体会

1.认识了无条件转移指令和条件转移指令。

2. 加强了我对条件跳转指令的运用以及和与无条件转移指令的对比。

3.循环分支的灵活运用和如何区分有符号数和无符号数。