

**课 程 实 验 报 告**

**课程名称： 汇编语言程序设计实验**

**实验名称： 实验二** 分支循环程序设计

**实验时间： 2017-10-24，18：30-21：50 实验地点： 南一楼804室**

**指导教师： 朱虹**

**专业班级：信息安全 201602班**

**学 号： U201614836 姓 名： 林晓斌**

**同组学生： 无 报告日期： 2017年 10 月 24日**

**原创性声明**

本人郑重声明：本报告的内容由本人独立完成，有关观点、方法、数据和文献等的引用已经在文中指出。除文中已经注明引用的内容外，本报告不包含任何其他个人或集体已经公开发表的作品或成果，不存在剽窃、抄袭行为。

特此声明！

学生签名：

日期：2017.10.17

|  |  |  |
| --- | --- | --- |
| 实验完成质量得分（70分）（实验步骤清晰详细深入，实验记录真实完整等） | 报告撰写质量得分（30分）（报告规范、完整、通顺、详实等） | 总成绩（100分） |
|  |  |  |

指导教师签字：

                     日期：

目录

[实验二 分支循环程序设计 3](#_Toc500881715)

[2.1 实验目的和要求 3](#_Toc500881716)

[2.2 实验内容 3](#_Toc500881717)

[2.2.1 任务1 4](#_Toc500881718)

[2.3 实验小结 11](#_Toc500881719)

[2.4 参考文献 12](#_Toc500881720)

# 实验二 分支循环程序设计

# 2.1 实验目的和要求

1. 掌握汇编源程序编辑工具、汇编程序、连接程序、调试工具TD的使用；
2. 理解数、符号、寻址方式等在计算机内的表现形式；
3. 熟悉常用的DOS功能调用；
4. 熟悉分支、循环程序的结构及控制方法，掌握分支、循环程序的调试方法；
5. 加深对转移指令及一些常用的汇编指令的理解。

# 2.2 实验内容

设计实现一个学生成绩查询的程序。

1、实验背景

在以BUF为首址的字节数据存储区中，存放着n个学生的课程成绩表（百分制），每个学生的相关信息包括：姓名（占10个字节，结束符为数值0，即串长最大为9），语文成绩（1个字节），数学成绩（1个字节），英语成绩（1个字节），平均成绩（1个字节）。

2、功能一：提示并输入待查询成绩的学生姓名

（1）使用9号DOS系统功能调用，提示用户输入学生姓名。

（2）使用10号DOS系统功能调用，输入学生姓名。输入的姓名字符串放在以in\_name为首址的存储区中。

3、功能二：以学生姓名查询有无该学生

（1）使用循环程序结构，在成绩表中查找该学生。

（2）若未找到，就提示用户该学生不存在。

（3）若找到，则将该学生课程成绩表的起始偏移地址保存到POIN字变量中。

4、功能三：计算所有学生的平均成绩

使用算数运算相关指令计算并保存每一个学生的平均成绩。

平均成绩计算公式：(A\*2+B+C/2)/3.5，即将语文成绩A乘以权重2、英语成绩C除以权重2后，与数学成绩B一起求和，再计算该生的平均成绩。

5、功能四：将功能二查到的学生的平均成绩进行等级判断，并显示判断结果。若平均成绩大于等于90分，显示“A”；大于等于80分，显示“B”；大于等于70分，显示“C”；大于等于60分，显示“D”；小于60分，显示“F”。

## 2.2.1 任务1

2.2.1.1 实验设计思路及寄存器分配

查找学生成绩的方法大致是参照了c语言中利用数组查找学生成绩的方法，只不过汇编写的程序里面关于地址的计算比数组更加麻烦，而且也需要更加小心翼翼。任务一的实现运用了9号调用和10号调用。任务二运用了两重循环，一是将两个字符串进行一一对比，二是进行未进行比较的同学的计数，比对完之后将该同学的数据的偏移地址保存在poin里。任务三是将每位同学的所有成绩求加权平均，si直接加上偏移地址取出成绩，并将其左移进行倍数放大，再除以7，最后将平均成绩保存后再将si+14进行下一位同学的求平均值。任务四将所搜素的人的平均值进行判定，最后用2号调用输出成绩等级。

存储单元分配

* + 1. BUF1 db变量 用于存储所有学生的信息
    2. IN\_NAME db变量 10个空间的0
    3. POIN dw变量 10个空间的0
    4. BUF3 db变量 存储提示找不到学生的字符串
    5. BUF2 db变量 存储请输入要查找学生的字符串

寄存器分配

任务一：DX:用于存储buf2偏移地址

AH：用于存放所调用的操作的数

任务二：SI:用于存放buf1的偏移地址

DI:用于存放in\_name的偏移地址

DX:用于剩余学生个数的计数

CL:用于比较字母数的计数

AL:存放SI中的数据

BL:存放DI中的数据

任务三：AL:用于存放第一个成绩

BL:用于存放第二个成绩

CL:用于存放第三个成绩

AX:存放第一个成绩及总成绩

BX:存放运算后的第二个成绩

CX:存放第三个成绩

CL:存放除数7

任务四：AL:存放查找的同学的平均成绩

DL:存放所要输出的ASCII码

2.2.2.2 流程图

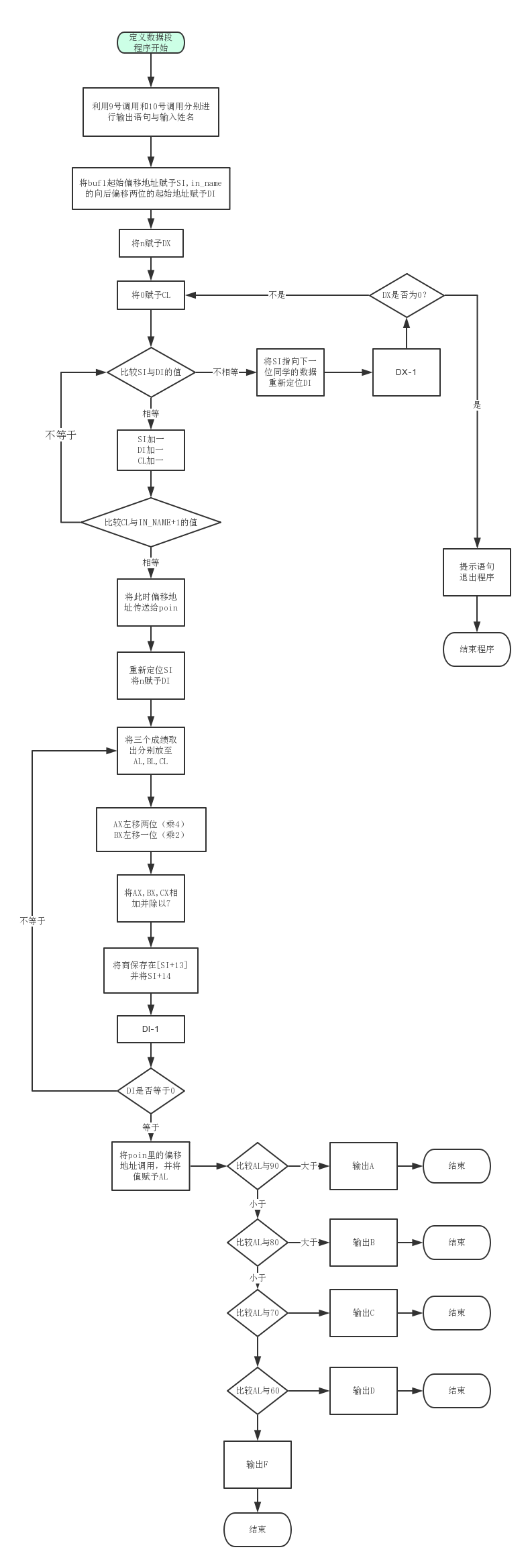


图2.1流程图

2.2.2.3 源程序

;----------------------------

.386

STACK SEGMENT STACK USE16

DB 200 DUP(0)

STACK ENDS

;----------------------------

DATA SEGMENT USE16

N EQU 5

BUF1 DB 'zhangsan',0,0

DB 100,85,90,?

DB 'lisi',6 DUP(0)

DB 80,90,90,?

DB N-3 DUP('TempValue',0,80,90,95,?)

DB 'linxiaob',0,0

DB 85,95,100,?

in\_name DB 10,11 DUP(0)

POIN DW 10,11 dup(0)

BUF3 DB 0AH,0DH,'can not find the student$'

BUF2 DB 0AH,0DH,'Please enter your name:$'

DATA ENDS

;------------------------------

CODE SEGMENT USE16

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

BEGIN: MOV AX,DATA

MOV DS,AX

LEA DX,BUF2

MOV AH,9

INT 21H

LEA DX,IN\_NAME

MOV AH,10

INT 21H ;任务一

MOV SI,OFFSET BUF1

MOV DI,OFFSET IN\_NAME

INC DI

INC DI

MOV DX,N

LP2: MOV CL,0

LP: MOV AL,[SI]

MOV BL,[DI]

CMP AL,BL

JNE LP1

INC SI

INC DI

INC CL

CMP CL,IN\_NAME+1

JNE LP

CMP BYTE PTR [SI],0

JNE LP

MOV CH,0

SUB SI,CX

MOV POIN,SI

JMP AVE

LP1: MOV CH,0

MOV BX,14

SUB BX,CX

ADD SI,BX

MOV DI,OFFSET IN\_NAME

INC DI

INC DI

DEC DX

JNZ LP2

LEA DX,BUF3

MOV AH,9H

INT 21H

MOV AH,4CH ;EXIT

INT 21H;任务二

AVE: MOV SI,OFFSET BUF1

MOV DI,N

LP3: MOV AL,[SI+10]

MOV BL,[SI+11]

MOV CL,[SI+12]

MOV AH,0

MOV BH,0

MOV CH,0

SAL AX,2

SAL BX,1

ADD AX,BX

ADD AX,CX

MOV CL,7

DIV CL

MOV [SI+13],AL

ADD SI,14

DEC DI

JNZ LP3 ;renwu3

MOV DL,0AH

MOV AH,2H

INT 21H

MOV BX,POIN;ERROR

MOV AX,DS:[BX+13]

CMP AL,90

JGE LEVEL1

CMP AL,80

JGE LEVEL2

CMP AL,70

JGE LEVEL3

CMP AL,60

JGE LEVEL4

LEVEL1: MOV DL,'A'

MOV AH,2H

INT 21H

MOV AH,4CH ;EXIT

INT 21H

LEVEL2: MOV DL,'B'

MOV AH,2H

INT 21H

MOV AH,4CH ;EXIT

INT 21H

LEVEL3: MOV DL,'C'

MOV AH,2H

INT 21H

MOV AH,4CH ;EXIT

INT 21H

LEVEL4: MOV DL,'D'

MOV AH,2H

INT 21H

MOV AH,4CH ;EXIT

INT 21H

;-----------------------------

CODE ENDS

END BEGIN

2.2.2.4 问题回答

1. 如何让9号功能调用显示的信息放在自己希望的位置？

可以在想显示的信息前加0ah，0dh等进行换行及移位。

2. 如果在9号功能调用时，带显示字符串的结尾没有“$”结束符会怎样？

会将之后内存中的数据一起显示直至数据段结束。如图2.1所示

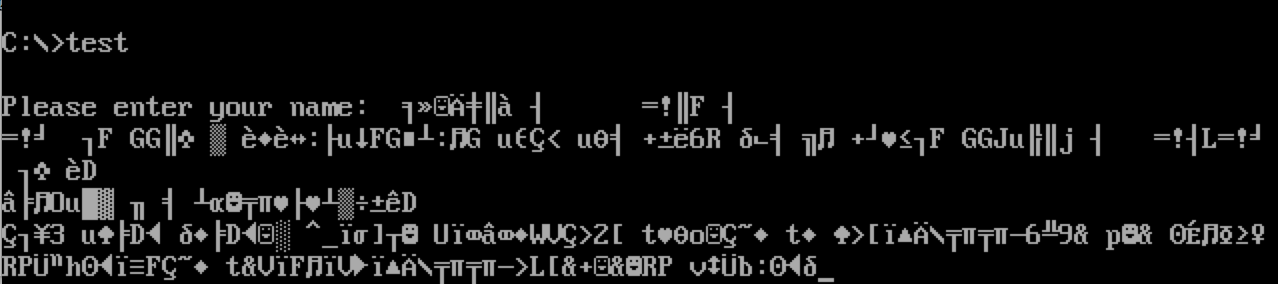


图2.2未输入$

3. 如果在9号功能调用前，未对DS赋值，或者未对DX给予正确的值，结果会如何？

会将内存中所有的内容都显示出来。如图2.2

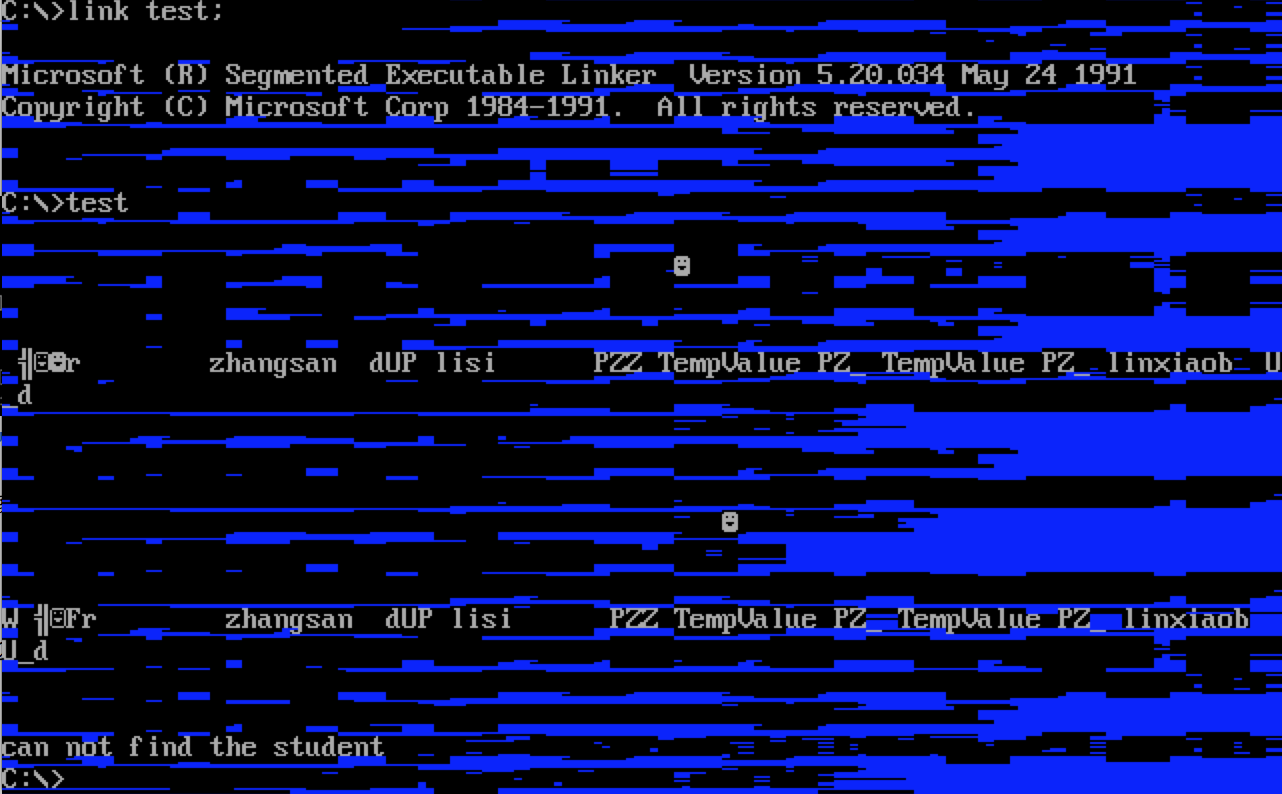


图2.3未对ds进行赋值

4. 10号功能调用时，输入的字符数超过定义的数量时，它是如何处理的？

不能再输入（或按不下回车键），如图2.3

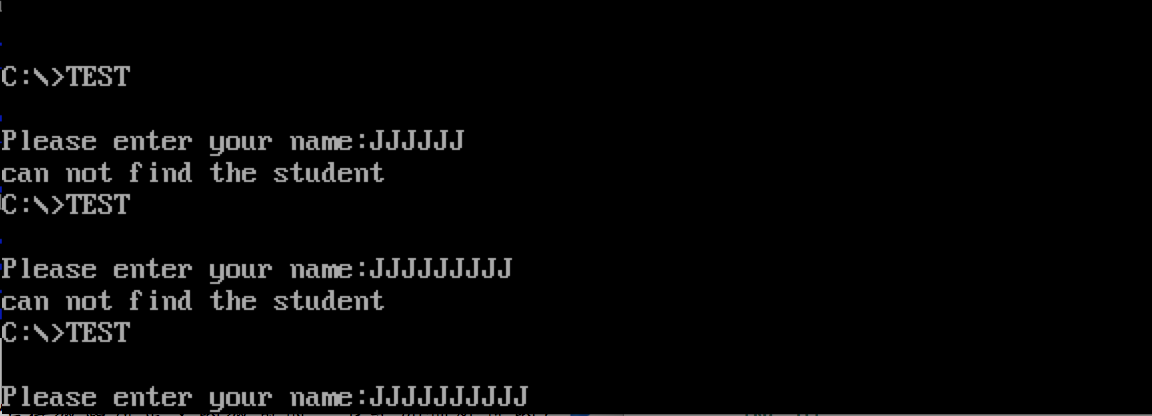


图2.4超过定义数量

2.2.2.5 程序运行结果

输入zhangsan，输出结果为A，如图2.4所示，zhangsan的成绩为100，85，80，平均成绩为91，符合结果。

输入lisi，输出结果为B，如图2.4所示，lisi的成绩为85，90，90，平均成绩为88，符合结果。

输入linxiaob，输出结果为A，如图2.4所示，linxiaob的成绩为85，95，100，平均成绩为90，符合结果。

输入hhhshish，输出结果为can not find the student，如图2.4，符合结果。

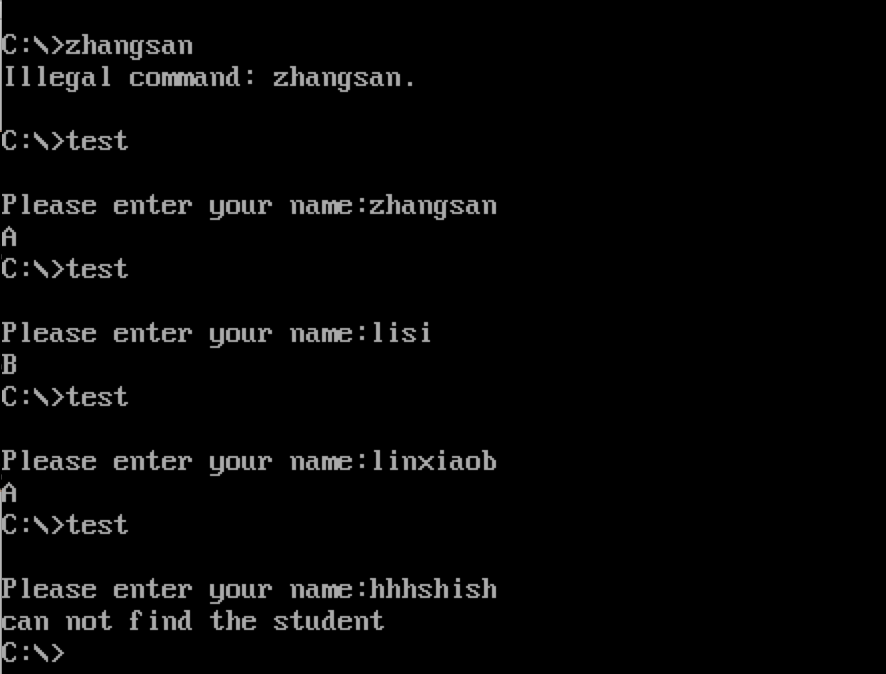


图2.5程序运行结果

# 2.3 实验小结

本次试验相对于上次试验实在是拔高了不少。我觉得这次试验的主要难点是任务二的循环部分，虽然只是两重循环，但是由于对汇编语言不熟悉，实现起来真的是磕磕绊绊，各种跳转眼花缭乱。有时候寄存器用着用着就重了，计数寄存器跟存放偏移地址寄存器抢着用，导致调试的时候输出了好多奇奇怪怪的结果，令人百思不得其解。再有就是汇编语言的长相相似（？），有的时候一下子不好找到自己想要找的语句，经常改了之后才发现改错了，有的时候是隔一段时间再看自己的程序，突然不知道自己想表达什么意思。针对于以上几点错误以及易混淆的地方，我觉得今后我应该要养成两个良好的习惯：1是将所用的寄存器及其功能记下，最好就写在程序旁，这样方便辨识可用的剩余寄存器。2是将自己写的代码按意群标号分段，方便阅读及检查程序。

# 2.4 参考文献

[1] 王元珍、曹忠升、韩宗芬.80X86汇编语言程序设计(第一版).华中科技大学出版社，2005年:1-348

[2] 许向阳.80X86汇编语言程序设计上机指南(第一版).华中科技大学出版社,2007年:1-224