

**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 分支程序、循环程序的设计**

**实验时间： 2016-3-29，14：00-17：30 实验地点： 南一楼804室63号实验台**

**指导教师： 张勇 专业班级：计算机科学与技术201409班**

**学 号： U201414813 姓 名： 唐礼威**

**同组学生： 无 报告日期： 2016年 3 月 30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：

成绩评定

|  |  |  |
| --- | --- | --- |
| 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**

[1 实验目的与要求 1](#_Toc451727238)

[2 实验内容 1](#_Toc451727239)

[3 实验过程 2](#_Toc451727240)

[3.1 任务1 2](#_Toc451727241)

[3.1.1 设计思想及存储单元分配 2](#_Toc451727242)

[3.1.2 流程图 3](#_Toc451727243)

[3.1.3 源程序 3](#_Toc451727244)

[3.1.4 实验步骤 6](#_Toc451727245)

[3.1.5 实验记录 7](#_Toc451727246)

[4 体会 8](#_Toc451727247)

[参考文献 9](#_Toc451727248)

# 实验目的与要求

1、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；

2、加深对转移指令及一些常用的汇编指令的理解；

3、加深对常用DOS功能调用指令的理解，进一步熟悉TD的使用。

# 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），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）。

例如：

N EQU 1000

BUF DB ‘zhangsan’,0,0 ;学生姓名，不足10个字节的部分用0填充

DB 100, 85, 80,？ ; 平均成绩还未计算

DB ‘lisi’,6 DUP(0)

DB 80, 100, 70,?

DB N-3 DUP( ‘TempValue’,0,80,90,95,?) ;除了3个已经具体定义了学生信息的成绩表以外，其他学生的信息暂时假定为一样的。

DB ‘wangwu’，0，0，0，0

DB 85, 85, 100, ?

2、功能一：提示并输入学生姓名

（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）若只是输入了回车，则回到“(1)”处重新提示与输入；若仅仅输入字符q，则程序退出,否则，准备进入下一步处理。

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，并回到“功能一（1）”的位置，提示并重新输入姓名。

（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。

**提示：**字符串比较时，当采用输入串的长度作为循环次数时，若因循环次数减为0而终止循环，则还要去判断成绩表中名字串的下一个字符是否是结束符0，若是，才能确定找到了。

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。**要求避免溢出**。

**提示：**使用循环程序结构，注意寻址方式的灵活使用。把小数3.5转换成分数后再运算避免使用浮点数指令。

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。

（1）平均成绩等级显示方式：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”。

**提示：**使用分支程序结构，采用2号DOS系统功能调用显示结果。

（2）使用转移指令回到“功能一（1）”处（提示并输入姓名）。

# 实验过程

## 任务1

### 设计思想及存储单元分配

设计思想：

本实验主要是要解决3个问题：一是提示并输入学生姓名；二是如何进行字符串的比较从而查询学生；三是如何计算平均分从而分出等级。依据本次实验的要求，此处用分支、循环程序的结构及控制方法来解决。

寄存器分配：

CX：存放学生数量（默认3），计数器作用；

BX：存放目标学生下标值，得到分数缓冲区首地址；

BP:存放缓冲区基地址；

AL:存放平均成绩；

AX，DX,SI：临时寄存器；

### 流程图



### 源程序

.386

STACK SEGMENT USE16 STACK

DB 200 DUP(0)

STACK ENDS

DATA SEGMENT USE16

N EQU 3

BUF DB 'zhangsan',0,0

DB 100,85,80,?

DB 'lisi',6 DUP(0)

DB 80,100,70,?

DB 'wangwu',4 DUP(0)

DB 85,85,100,?

MSG DB 'Input the name : $'

CONTINU DB 'Enter any keys to continue(if enter q,exit!):$'

FAIL DB 'Not Exist!$'

INPUT DB 10

DB ?

DB 10 DUP(0)

DATA ENDS

CODE SEGMENT USE16

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

START: MOV AX, DATA

MOV DS, AX

JMP BEGIN

FAILED: LEA DX, OFFSET FAIL ; 查找失败

MOV AH, 9H

INT 21H

LOOPA: MOV DL, 0AH ;换行符

MOV AH, 2H

INT 21H

LEA DX, OFFSET CONTINU ;继续

MOV AH, 9H

INT 21H

MOV DL, 0AH ;换行符

MOV AH, 2H

INT 21H

MOV DL, 0DH

MOV AH, 2H

INT 21H

MOV AH, 8H

INT 21H

CMP AL, 71H

JE OVER ;输入 'q', 退出程序

BEGIN: MOV CX, N ; 学生个数

LEA DX, OFFSET MSG ;输出提示信息

MOV AH, 9H

INT 21H

MOV DL, 0AH ; 换行符

MOV AH, 2H

INT 21H

LEA DX, OFFSET INPUT ; 读入学生姓名

MOV AH, 0AH

INT 21H

MOV DL, 0AH ; 换行符

MOV AH, 2H

INT 21H

LEA BP, OFFSET INPUT ; 将 INPUT 基址存放至 BP

ADD BP,2

CMP DS:BYTE PTR [BP-1],0H ;空字符串

JE LOOPA

INC CX

COMPA: DEC CX

JE FAILED ; 查找失败, 重新输入

MOV BX, N ; 计算目标学生下标值, 存放至 BX

SUB BX, CX

IMUL BX, 14 ; 根据目标学生下标值, 找到分数缓冲区首地址

MOV AX, 10 ; 临时计数器

MOV SI, 0

COMPB: MOV DL, [BX + SI]

MOV DH, BYTE PTR DS:[BP + SI]

CMP DL, 0 ; 如果缓冲区姓名已结束,说明查找成功

JE CAL ; 跳转至平均成绩计算处

CMP DH, DL ; 比较 当前缓冲区姓名 与 输入姓名 字符

JNE COMPA ; 当前字符相同,继续循环以比较下一字符

INC SI

DEC AX

JNE COMPB

CAL: MOV BX, N ; 计算目标学生下标值, 存放至 BX

SUB BX, CX

IMUL BX, 14

ADD BX, 10 ; 根据目标学生下标值, 找到分数缓冲区首地址 BX = m \* 14 + 10

MOV AX, 0

MOV DX, 0

MOV AL, [BX] ; 计算平均成绩

ADD AX, AX

MOV DL, [BX +1]

ADD AX, DX

MOV DL, [BX +2]

SAR DL, 1

ADD AX, DX ;AL=A\*2+B+C/2

SAL AX, 1

MOV DL, 7

IDIV DL ; AL = 2 \* AL / 7

MOV [BX + 3], AL; AVG = AL ( AL / 3.5)

CMP AL, 90

JGE LEVELA

CMP AL, 80

JGE LEVELB

CMP AL, 70

JGE LEVELC

CMP AL, 60

JGE LEVELD

JMP LEVELF

LEVELA:MOV DL, 41H ;输出成绩等级

MOV AH, 2H

INT 21H

JMP LOOPA

LEVELB:MOV DL, 42H

MOV AH, 2H

INT 21H

JMP LOOPA

LEVELC:MOV DL, 43H

MOV AH, 2H

INT 21H

JMP LOOPA

LEVELD:MOV DL, 44H

MOV AH, 2H

INT 21H

JMP LOOPA

LEVELF:MOV DL, 46H

MOV AH, 2H

INT 21H

JMP LOOPA

OVER: MOV AH, 4CH

INT 21H

CODE ENDS

END START

### 实验步骤

1. 使用编辑程序EDIT.EXE录入源程序，存盘文件名为shiyan.ASM。

2. 使用MASM 6.0汇编源文件。即MASM shiyan；

3. 观察提示信息，若出错，则用编辑程序修改错误，存盘后重新汇编，直至不再报错为止。

4. 使用连接程序LINK.EXE将汇编生成的shiyan.OBJ文件连接成执行文件。

即LINK shiyan；

5. 若连接时报错，则依照错误信息修改源程序。之后重新汇编和连接，直至不再报错并生成shiyan.EXE文件。

6. 执行该程序。即在命令行提示符后输入shiyan后回车，观察执行现象。

7. 让9号功能调用显示的信息放在自己希望的位置。

8. 在9号功能调用时，尝试带显示字符串的结尾没有“$”结束符。

9. 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？

### 实验记录

（包括实验条件、输入/输出、错误和修改等信息的记录）

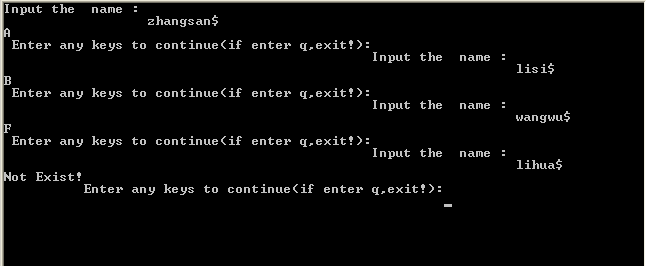
1. 实验环境条件：P3 1GHz，256M内存；WINDOWS 2000命令行窗口；EDIT.EXE 2.0；MASM.EXE 6.0； LINK.EXE 5.2; TD.EXE 5.0。

2、按照流程图编写完程序，编译和连接都没问题，运行时出现



后来发现是9号调用的字符串在之前定义时没有以$结尾，改正后程序运行正确

3、程序运行正确，但格式很差，如下图：



改进方法：在LOOPA里1号调用的换行符后面加上一个回车换行符，即

MOV DL, 0AH ;换行符

MOV AH, 2H

INT 21H

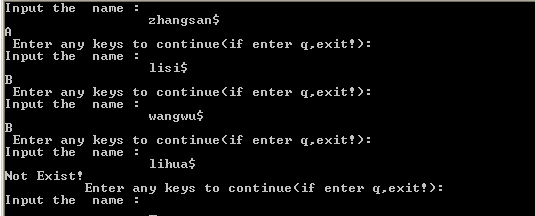
后面加上：

MOV DL, 0DH

MOV AH, 2H

INT 21H

改正后运行如下图：



分析：存放的三个人平均成绩依次为zhangsan：88 等级A； lisi：83 等级B； wangwu：90 等级B，可见与输出结果相符合

4、对于输入字符超过定义的数量时，会出现无法再继续输入的情况如下图：



# 体会

这次上机，进一步提高了自己的汇编语言编程能力，通过实验对这门语言有了更深的理解，更加熟悉了调试工具td的使用以及对dos功能调用的理解。这次上机主要是熟悉分支、循环程序的结构和控制方法，掌握了它们的调试方法，对转移指令有了更深的理解，是从比高级语言更底层的角度去理解了其含义，对其它的汇编指令有了初步的认识，尽管有些指令还是需要查表才能写出来，但相信在今后的几次实验中会对更多的指令更加的熟悉和掌握。

# 参考文献

[1] 王元珍等.80x86汇编语言程序设计.版本(第1版)

[2] 王晓虹等.汇编语言程序设计教程.版本(第1版)