

1. 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。

2. 实验内容

设计实现一个学生成绩查询的程序

1. 实验说明

在以 BUF 为首地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, ?); 除了已经定义了的学生的信息及成绩表
                                                外，其他学生的暂时成绩假定是一样的
        db 'WangWu', 4 dup(0) ;最后一个必须修改为自己名字的拼音
        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。输入的学生姓名字符串存放在以 in_name 为首地址的存储区中。
- (3) 若只是输入了回车，则回到“（1）”处重新提示并输入；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，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

- (1) 使用算术运算相关指令计算并保存每一个学生的平均成绩。
- (2) 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，再与数学成绩 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运用。把小数 3.5 转换成分数后再运算以避免使用浮点指令。

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）。

3. 实验过程

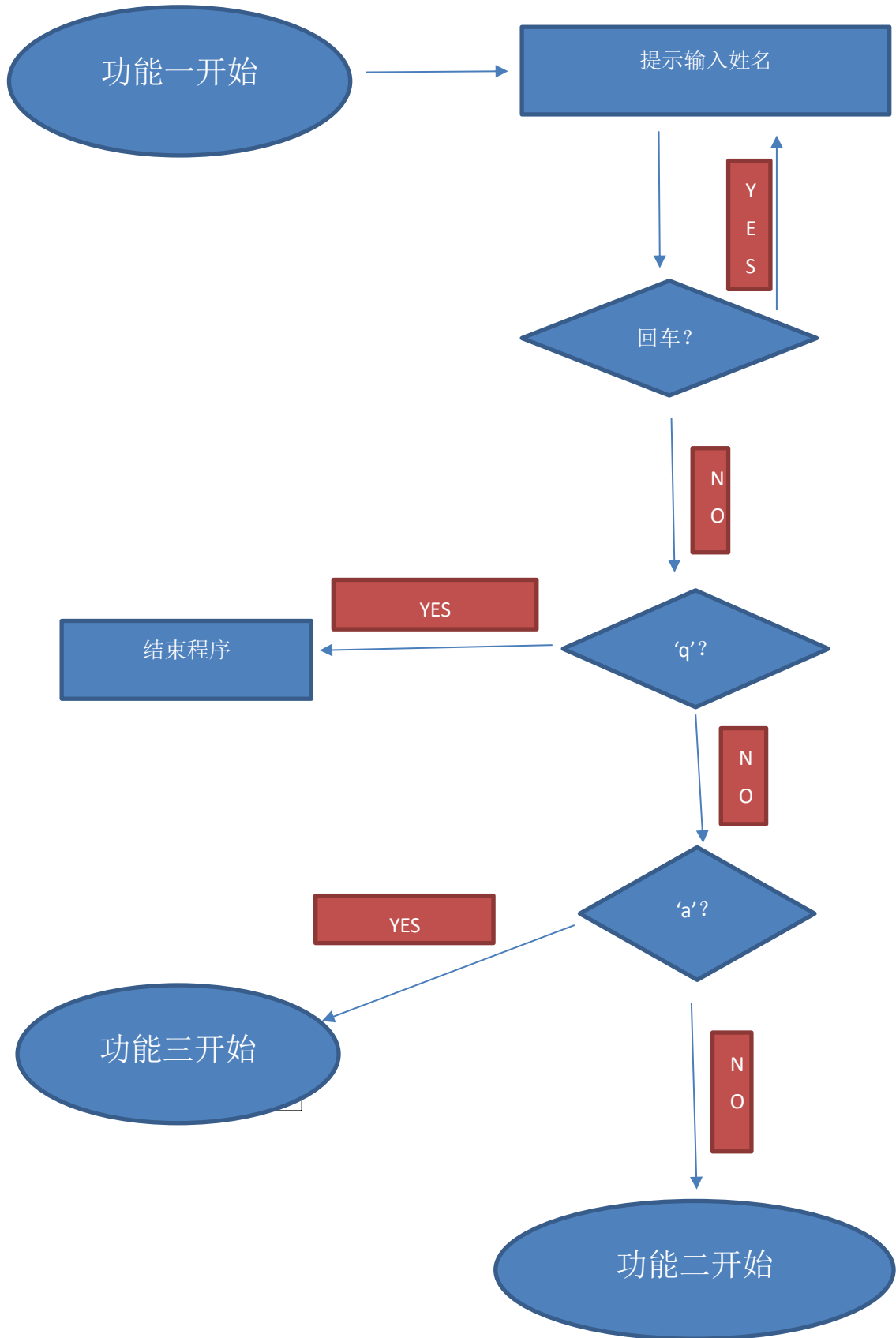
一. 提纲:

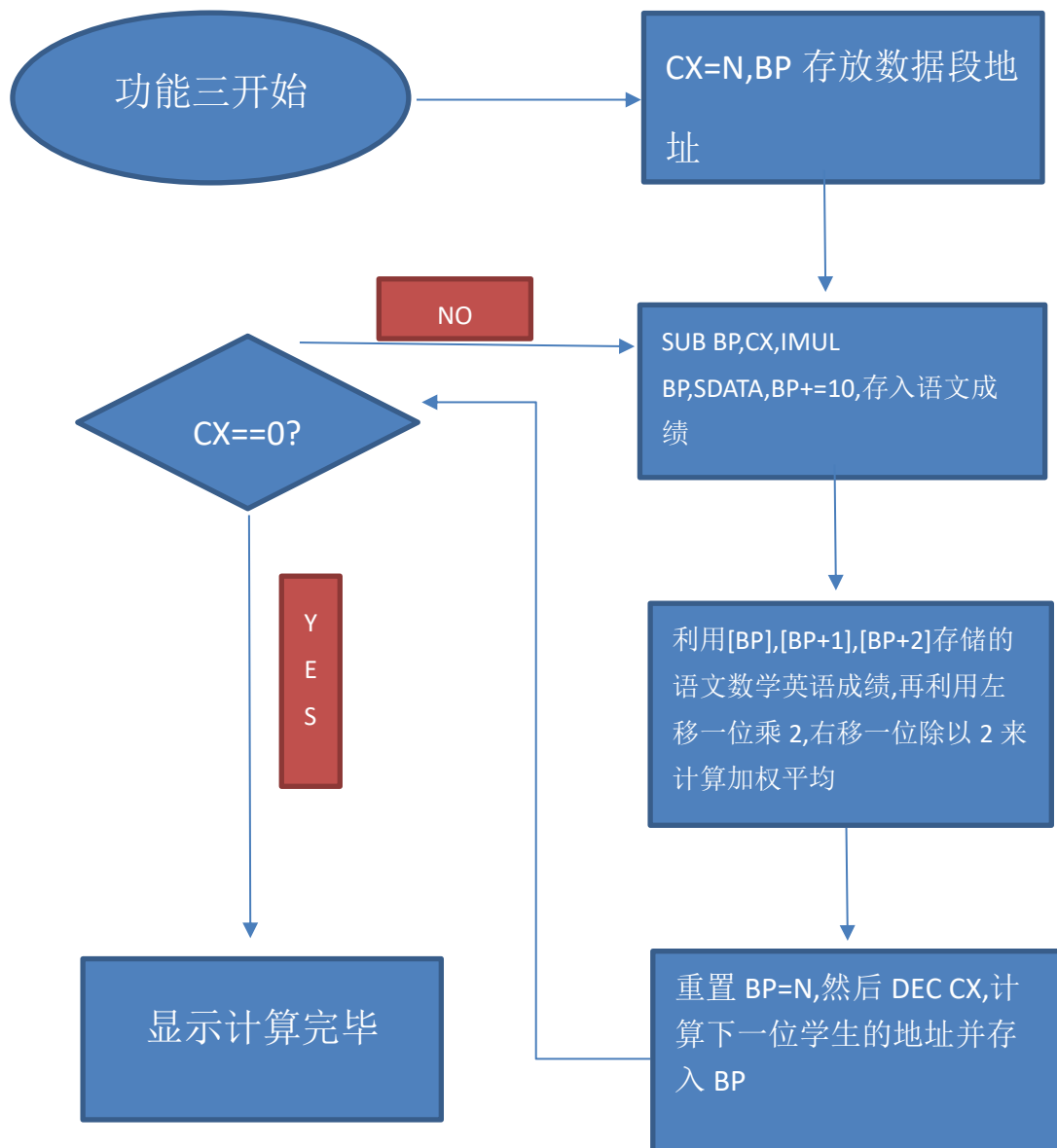
利用分支、循环程序的结构及控制方法, 和一些常用的汇编指令, 设计实现一个学生成绩查询的程序。查找学生姓名, 计算平均成绩, 对成绩进行评级等功能分段实现, 进行程序的模块化, 完善实验报告。

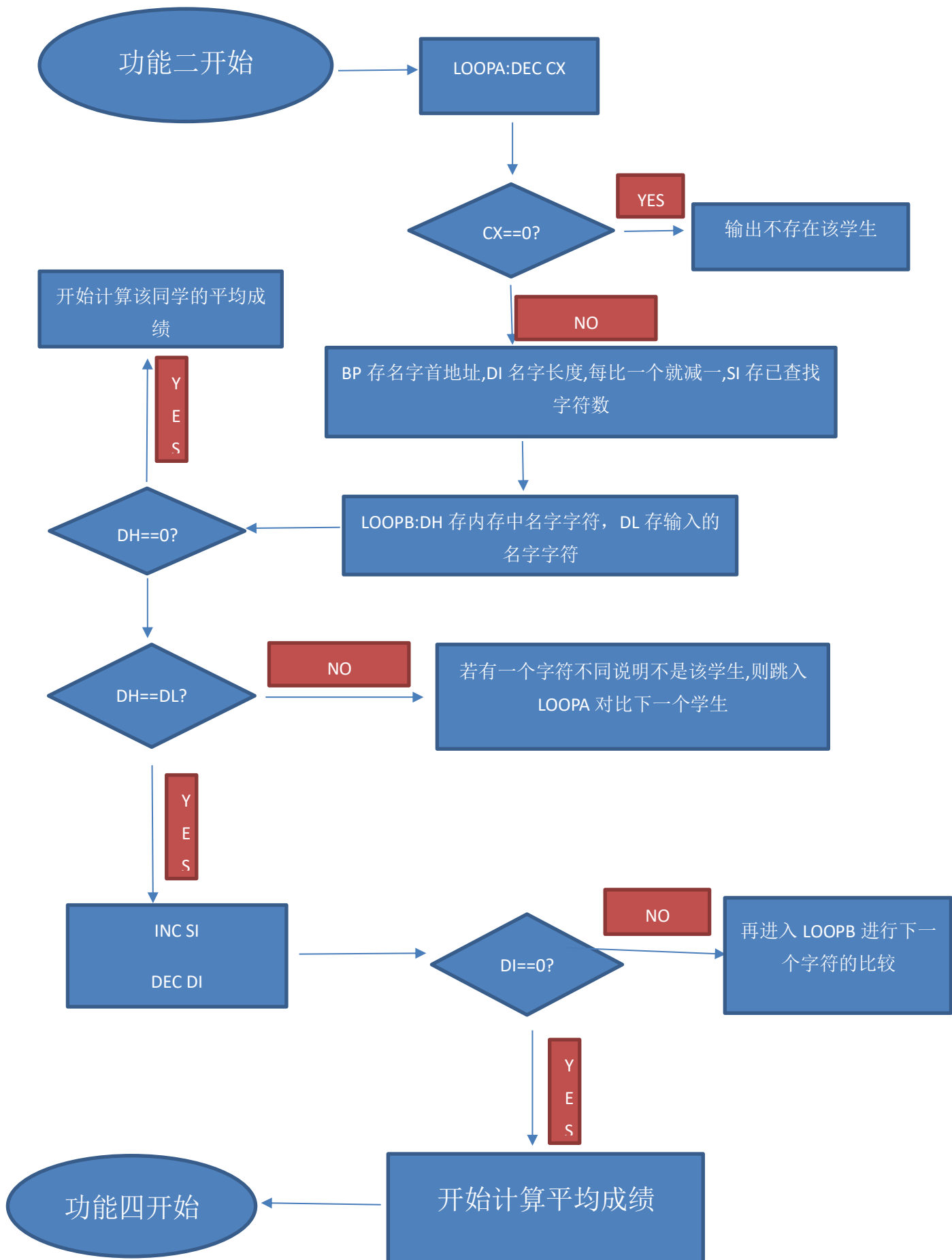
二. 设计思想:

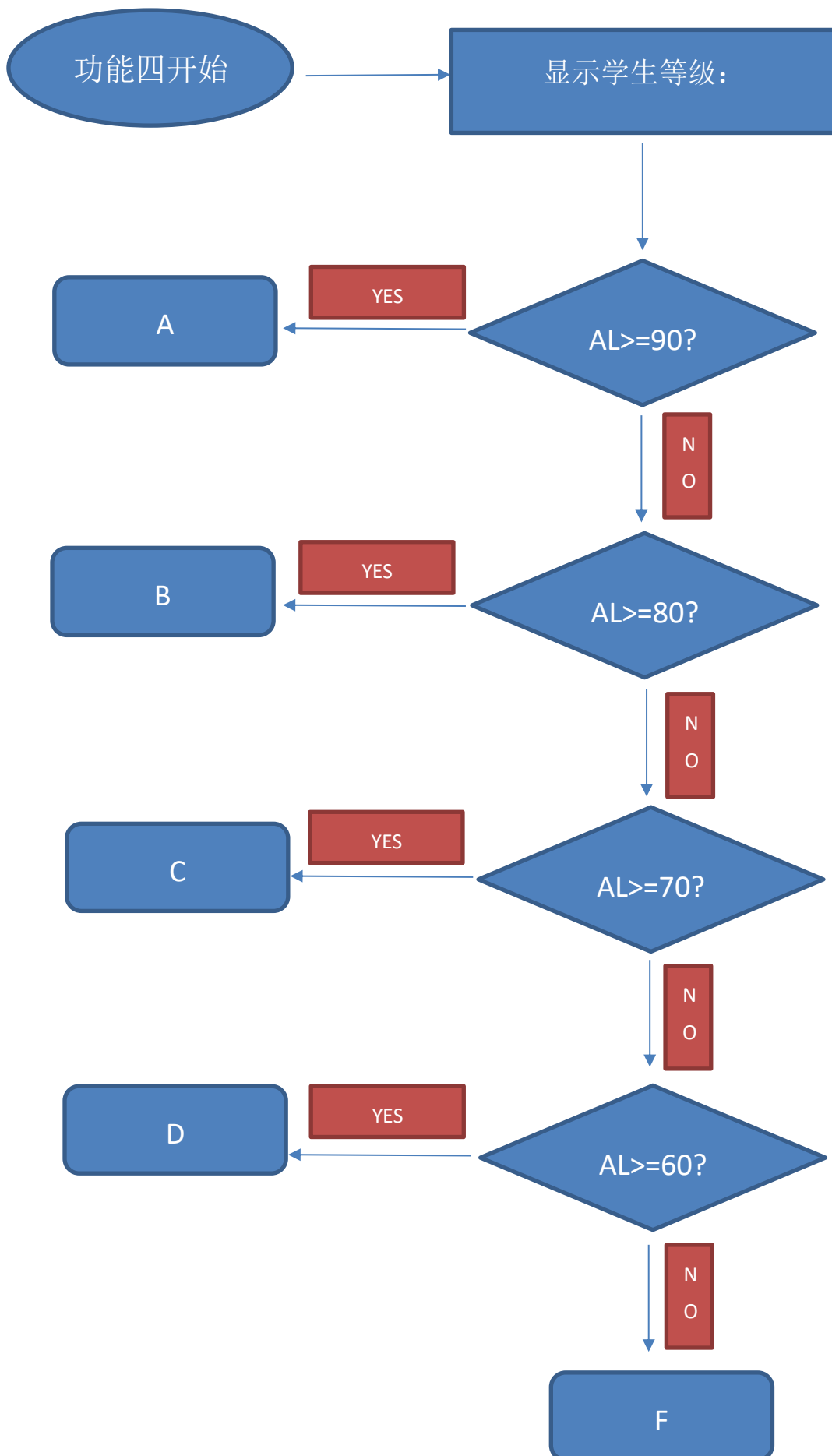
1. 定义数据段中的各项数据, 学生的姓名, 成绩, 以及各项提示语句。
2. 完成功能一, 利用 9 号 DOS 系统功能调用, 进行提示输入姓名。利用 10 号 DOS 系统功能调用, 输入学生姓名, 并存放在以 IN_NAME 为首地址的存储区中。若输入回车, 则回到功能一开始部分, 若输入 q 则程序退出, 否则进行下一步处理。若输入 a 则计算所有同学的平均成绩。
3. 实现功能二, 利用循环程序结构, 在成绩表中查找该学生, 若未找到, 则提示没有该学生, 并返回功能一的位置, 重新提示输入姓名, 若找到, 则将该学生的成绩表起始偏移地址保存到 PION 字变量中。
4. 实现功能三, 使用循环结构, 注意寻址方式的灵活运用, 利用算术运算相关指令计算并保存每一个学生的平均成绩。
5. 实现功能四, 利用分支程序结构, 采用 2 号 DOS 系统功能调用, 对功能二查到的学生的平均成绩进行等级判断, 若 ≥ 90 分, 则显示 A, 若 ≥ 80 分, 则显示 B, 若 ≥ 70 分, 则显示 C, 若 ≥ 60 分, 则显示 D, 若 < 60 分, 则显示 F。最后回到功能一。

三. 流程图









四. 源程序

```
.386
DATA SEGMENT USE16
N EQU 1000
NLENG EQU 10; 一个学生名字的长度
SDATA EQU 14; 一个学生数据的长度
BUF DB 'ZhangSan',0,0; 学生姓名: 张三
    DB 100,85,80,?; 张三的成绩, 平均成绩未计算
    DB 'LiSi',6 DUP(0)
    DB 80,100,70,?
    DB N-3 DUP('TempValue',0,80,90,95,?)
    DB 'XuZiChuan',0; 本人名字
    DB 85,90,95,?; 本人的成绩

IN_NAME DB 10; 最多欲接收字符个数
        DB ?; 实际接收字符个数
        DB 10 DUP(0); 第三字节开始存放输入的字符串

TEMPSTR DB 10 DUP(0)
NOTICE DB 'Please input the students name:',0DH,0AH; 0DH 代表回车, 0AH 代表换行
        DB '(enter q to quit, enter a to calculate the avarage grade)$'
GRANOT DB 'The grade of the student is:$'
CONTINUE DB 'Please enter any key to continue...',0DH,0AH,'$'
NOTEXIST DB 'Sorry, the student does not exist.',0DH,0AH,'$'
CALSUCES DB 'Calculate all the avarage grades successfully!',0DH,0AH,'$'
ENTRE DB 0DH,0AH,'$'
POIN DW 0
DATA ENDS

STACK SEGMENT USE16 STACK
DB 200 DUP(0)
STACK ENDS ; 定义堆栈段

CODE SEGMENT USE16

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

START: MOV AX,DATA
        MOV DS,AX
        MOV ES,AX

HINT: LEA DX,NOTICE; 将 NOTICE 的偏移地址放入 DX
```

```

MOV AH,09H; 9号DOS功能调用 输出字符串
INT 21H ;弹出字符串提示输入
LEA DX,ENTRE;回车换行
MOV AH,09H
INT 21H

LEA DX,IN_NAME;将IN_NAME的偏移地址放入DX,
MOV AH,0AH;10号DOS功能调用
INT 21H;读入学生姓名
MOV CX,N;读入学生个数
MOV BX,OFFSET IN_NAME;将IN_NAME的偏移地址存入BX
ADD BX,2;BX=BX+2 跳过两个字符串长度,从名字开始

CMP IN_NAME+2,0DH;如果只输入回车
JE HINT;返回开始输入
CMP IN_NAME+2,'q';如果输入q
JE QUIT;则退出
CMP IN_NAME+2,'a';如果输入a
JE GRALL;则计算平均成绩

LEA DX,ENTRE;回车换行
MOV AH,09H
INT 21H
INC CX;学生数加1
LOOPA: DEC CX
        JE NOMAT ;如果查找了N个人的名字也没查找到 输出不存在该学生
        MOV BP,N
        SUB BP,CX
        IMUL BP,SDATA;找到名字的首地址,即14*(N-CX)
        MOV DI,NLENG;DI中存入名字长度,每比一个就减一
        MOV SI,0;已查找的字符数
LOOPB: MOV DH,DS:[BP+SI];内存中名字的字符
        MOV DL,DS:[BX+SI];输入的名字的字符
        CMP DH,0;若名字后面有'0',比到0就说明比完了,开始计算平均成绩
        JE GRADE;
        CMP DH,DL;若有一个字符不符说明不是该学生
        JNE LOOPA;进入LOOPA对比下一个学生
        INC SI;若该字符一样,已查找的字符数加一
        DEC DI;DI控制进入GRADE还是LOOPB
        JE GRADE;若比完了,开始计算平均成绩
        JNE LOOPB;若没比完,再进入LOOPB进行下一个字符的比较
GRADE: MOV BP,N
        SUB BP,CX
        IMUL BP,SDATA;找到该同学名字的首地址,即14*(N-CX)
        ADD BP,NLENG;BP+=NLENG,跳过名字,取得成绩的首地址

```

```
MOV POIN,BP;将成绩的偏移地址存入 POIN
MOV DX,0
MOV AX,0;清零 AX,DX
MOV AL,DS:[BP];将语文成绩存入 AL
SAL AX,1;AX 左移一位,实现乘 2
MOV DL,DS:[BP+1];将数学成绩存入 DL
ADD AX,DX;语文成绩乘 2 加数学成绩乘 1 存入 AX
MOV DL,DS:[BP+2];将英语成绩存入 DL
SAR DL,1;DL 右移一位,实现除以 2
ADD AX,DX;语文成绩乘 2 加数学成绩加英语成绩除以 2 乘 1 存入 AX
SAL AX,1;AX 左移一位,实现乘 2
MOV DL,7;再除以 7
IDIV DL;整体是除以 3.5
MOV DS:[BP+3],AL;将计算结果存入平均成绩位
LEA DX,GRANOT;输出学生的平均成绩等级
MOV AH,09H
INT 21H
CMP AL,90
JGE GRADE_A
CMP AL,80
JGE GRADE_B
CMP AL,70
JGE GRADE_C
CMP AL,60
JGE GRADE_D
JL GRADE_F
```

```
NOMAT: LEA DX,NOTEXIST;输出不存在该学生
MOV AH,09H
INT 21H
JMP CONTI
```

```
CONTI: LEA DX,ENTRE ;回车换行
MOV AH,09H
INT 21H
LEA DX,CONTINUE;输出继续字符串
MOV AH,09H
INT 21H
```

```
NEXT: MOV AH,01H
INT 21H ;字符输入,继续下一个操作
JMP HINT
```

```
GRADE_A:MOV DL,41H
MOV AH,2H
```

```

        INT 21H
        LEA DX,ENTRE
        MOV AH,9
        INT 21H
        JMP CONTI

GRADE_B:MOV DL,42H
        MOV AH,2H
        INT 21H
        LEA DX,ENTRE
        MOV AH,9
        INT 21H
        JMP CONTI

GRADE_C:MOV DL,43H
        MOV AH,2H
        INT 21H
        LEA DX,ENTRE
        INT 21H
        JMP CONTI

GRADE_D:MOV DL,44H
        MOV AH,2H
        INT 21H
        LEA DX,ENTRE
        MOV AH,9
        INT 21H
        JMP CONTI

GRADE_F:MOV DL,46H
        MOV AH,2H
        INT 21H
        LEA DX,ENTRE
        MOV AH,9
        INT 21H
        JMP CONTI

GRALL: MOV CX,N
        MOV BP,N
GRALP: SUB BP, CX; 每计算完一个,CX 就-1
        IMUL BP, SDATA
        ADD BP, NLENG; 使得BP 是当前学生的名字的首地址
        MOV DX,0
        MOV AX,0
        MOV AL,DS:[BP]; 开始计算平均成绩,方法同GRADE 段
        SAL AX,1

```

```

MOV DL, DS:[BP +1]
ADD AX, DX
MOV DL, DS:[BP +2]
SAR DL, 1
ADD AX, DX;
SAL AX, 1
MOV DL, 7
IDIV DL
MOV DS:[BP+3], AL
MOV BP,N; 每计算完将BP 重置为N 来定位下一个学生名字的首地址
DEC CX; 每计算完一个学生 CX 就-1
JNE GRALP; 只要还有学生 就进入 GRALP 继续算该学生的平均成绩
LEA DX, CALSUCES
MOV AH, 9; 显示计算完毕
INT 21H
JE CONTI
QUIT: MOV AH,4CH
INT 21H
CODE ENDS
END START

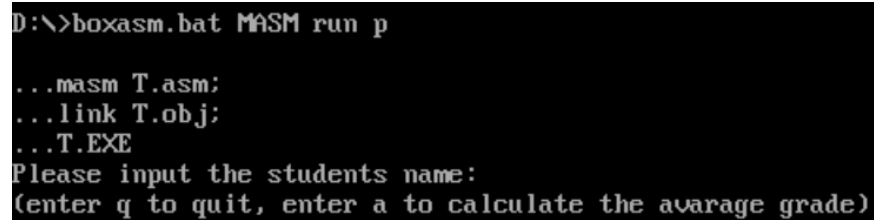
```

五. 实验步骤

1. 在 VSCODE 中编写 .asm 程序。
2. MASM 编译通过后运行。
3. 分别输入 “ZhangSan”、“q”、回车验证功能一是否成功实现。
4. 分别输入 “ZhangSan”、“LiSi”、“XuZiChuan”、“WangWu” 验证功能二是否成功实现。
5. 输入 “a” 验证功能三是否实现。
6. 观察第 4 步的结果，验证功能四是否成功实现。

六. 实验结果

1. 编写的.asm 程序已在源程序部分列出。
2. 编译通过后运行结果如图 2-1

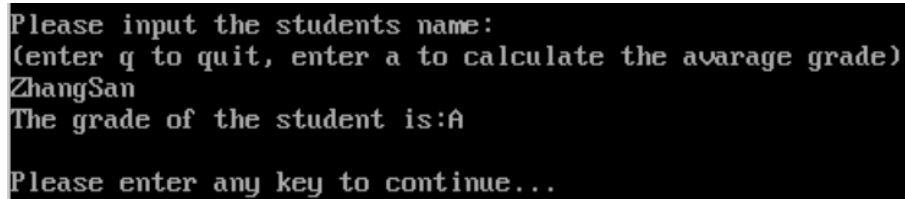


```
D:\>boxasm.bat MASM run p

...masm T.asm;
...link T.obj;
...T.EXE
Please input the students name:
(enter q to quit, enter a to calculate the avarage grade)
```

图 2-1

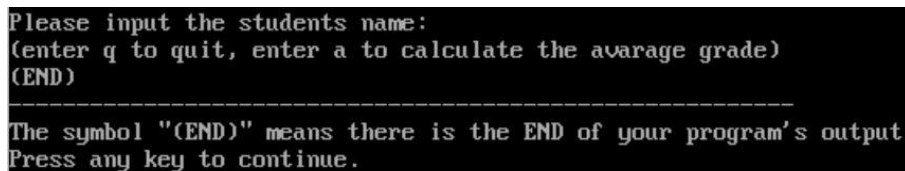
3. 输入“ZhangSan”后的结果如图 3-1



```
Please input the students name:
(enter q to quit, enter a to calculate the avarage grade)
ZhangSan
The grade of the student is:A
Please enter any key to continue...
```

图 3-1

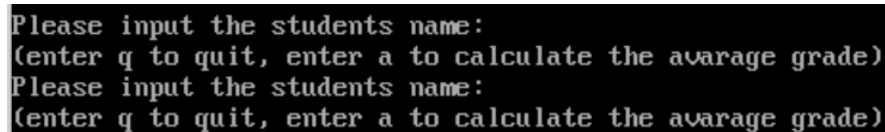
- 输入“q”后的结果如图 3-2



```
Please input the students name:
(enter q to quit, enter a to calculate the avarage grade)
(END)
-----
The symbol "(END)" means there is the END of your program's output
Press any key to continue.
```

图 3-2

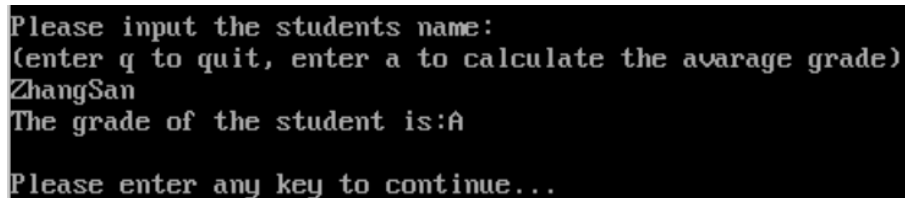
- 输入回车后的结果如图 3-3



```
Please input the students name:
(enter q to quit, enter a to calculate the avarage grade)
Please input the students name:
(enter q to quit, enter a to calculate the avarage grade)
```

图 3-3

4. 输入“ZhangSan”后的结果如图 4-1



```
Please input the students name:
(enter q to quit, enter a to calculate the avarage grade)
ZhangSan
The grade of the student is:A
Please enter any key to continue...
```

图 4-1

- 输入“LiSi”后的结果如图 4-2

```
Please input the students name:
(enter q to quit, enter a to calculate the avarage grade)
LiSi
The grade of the student is:B
Please enter any key to continue...
```

图 4-2

输入“XuZiChuan”后的结果如图 4-3

```
Please input the students name:
(enter q to quit, enter a to calculate the avarage grade)
XuZiChuan
The grade of the student is:B
Please enter any key to continue...
```

图 4-3

输入“WangWu”后的结果如图 4-4

```
Please input the students name:
(enter q to quit, enter a to calculate the avarage grade)
WangWu
Sorry, the student does not exist.
Please enter any key to continue...
```

图 4-4

5. 输入“a”后的结果如图 5-1

```
Please input the students name:
(enter q to quit, enter a to calculate the avarage grade)
Calculate all the avarage grades successfully!
Please enter any key to continue...
```

图 5-1

6. 观察第 4 步结果，分别对 ZhangSan, LiSi, XuZiChuan 的平均成绩均进行了评级，则功能四成功实现。

4. 总结与体会

本次实验利用分支和循环结构设计学生成绩查询系统，感觉难度较大，通过查阅书籍，资料，以及询问同学，完成了本次实验。

通过本次实验，我熟悉了分支、循环程序的结构及控制方法，简单掌握了分支、循环程序的调试方法。加深了转移指令及一些常用的汇编指令的理解。加深了对常用 DOS 功能调用指令的理解，学会了常用的输入输出调用，进一步熟悉了 TD 的使用。

希望能通过更多的实验熟悉汇编程序流程，掌握更多的汇编指令，深入理解汇编程序的框架。