

第三次上机练习，函数-数据类型

李宇豪 21305412

以下是记录学生信息的程序，请按要求扩充程序，增加功能。

```
#include <iostream>
#define N 5
using namespace std;
struct Student //定义结构体
{
    int no; //学号
    string name; //姓名
};

void main( )
{
    int i;
    Student ph_2021[N];
    //录入十个学生的信息
    i = 0;
    ph_2021[i].no = 202101;
    ph_2021[i].name = "熊大";
    i = i + 1;
    ph_2021[i].no = 202102;
    ph_2021[i].name = "熊二";
    i = i + 1;
    ph_2021[i].no = 202104;
    ph_2021[i].name = "张三";
    i = i + 1;
    ph_2021[i].no = 202106;
    ph_2021[i].name = "李四";
    i = i + 1;
    ph_2021[i].no = 202107;
    ph_2021[i].name = "王五";
    i = 1;
}
```

1. 枚举类型的定义与赋值

在结构体 ‘struct Student’ 添加成员描述学生的性别；录入学生的性别。
熊大，男；熊二，男；张三，女；李四，男；王五，女。

```
#include <iostream>
#define N 5
using namespace std;

struct Student //定义结构体
{
    int no;    //学号
    string name; //姓名
    string gender; //性别
};

int main( )
{
    int i;
    Student ph_2021[N];
    //录入十个学生的信息
    i = 0;
    ph_2021[i].no = 202101;
    ph_2021[i].name = "熊大";
    ph_2021[i].gender = "男";

    i = i + 1;
    ph_2021[i].no = 202102;
    ph_2021[i].name = "熊二";
    ph_2021[i].gender = "男";

    i = i + 1;
    ph_2021[i].no = 202104;
    ph_2021[i].name = "张三";
    ph_2021[i].gender = "女";

    i = i + 1;
    ph_2021[i].no = 202106;
    ph_2021[i].name = "李四";
    ph_2021[i].gender = "男";

    i = i + 1;
    ph_2021[i].no = 202107;
    ph_2021[i].name = "王五";
    ph_2021[i].gender = "女";

    i = 1;

    return 0;
}
```

2. 联合体的定义与赋值

在结构体中添加学生成绩，并录入如下学生成绩。

熊大，90；熊二，63；张三，86；李四，n；王五，74。（n 代表挂科）。

```
#include <iostream>
#define N 5
using namespace std;

union Grade //定义联合体
{
    int mark; //分数
    char fail; //挂科
};

struct Student //定义结构体
{
    int no; //学号
    string name; //姓名
    string gender; //性别
    Grade stu_grade; //成绩
};

int main( )
{
    int i;
    //录入十个学生的信息
    i = 0;
    ph_2021[i].no = 202101;
    ph_2021[i].name = "熊大";
    ph_2021[i].gender = "男";
    ph_2021[i].stu_grade.mark = 90;

    i = i + 1;
    ph_2021[i].no = 202102;
    ph_2021[i].name = "熊二";
    ph_2021[i].gender = "男";
    ph_2021[i].stu_grade.mark = 63;

    i = i + 1;
    ph_2021[i].no = 202104;
    ph_2021[i].name = "张三";
    ph_2021[i].gender = "女";
    ph_2021[i].stu_grade.mark = 86;

    i = i + 1;
    ph_2021[i].no = 202106;
    ph_2021[i].name = "李四";
    ph_2021[i].gender = "男";
    ph_2021[i].stu_grade.fail = 'n';

    i = i + 1;
```

```

    ph_2021[i].no = 202107;
    ph_2021[i].name = "王五";
    ph_2021[i].gender = "女";
    ph_2021[i].stu_grade.mark = 74;

    return 0;
}

```

3. 函数定义与调用

定义一个函数，用于输出一个学生的所有信息。键盘输入学号，调用该函数输出该学生的所有信息；如输入的学号不在数据库中，则输出‘该学生不存在！’。

//该函数定义中需要用到 <cmath> 头文件

//使用该函数定义时需要将 main 函数中 Student ph_2021[N]; 声明放在函数外改为全局变量声明

```
Student ph_2021[N];
```

```
int info_output() //定义信息输出函数
```

```

{
    int i, stu_number;
    int j = -1; //用于判断学生信息是否存在

    cout << "请输入学号: " << endl;
    cin >> stu_number;

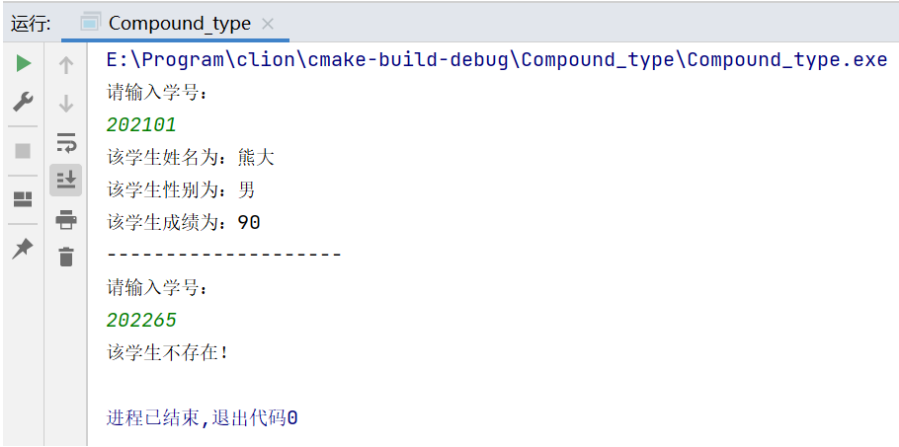
    for (i=0; i <= N; i++)
    {
        if (fabs(stu_number - ph_2021[i].no) < 1.0e-6)
        {
            cout << "该学生姓名为: " << ph_2021[i].name << endl;
            cout << "该学生性别为: " << ph_2021[i].gender << endl;
            j = 1; //表示该学生信息存在
            if (ph_2021[i].stu_grade.fail == 'n')
            {
                cout << "该学生成绩为: " << "挂科" << endl;
            }
            else
            {
                cout << "该学生成绩为: " << ph_2021[i].stu_grade.mark << endl;
            }
            break;
        }
    }

    if ( j < 0)
    {
        cout << "该学生不存在!" << endl;
    }

    return 0;
}

```

运行结果为



4. 联合体的更新

如果输入学号后，显示该学生不及格，则提示输入补考成绩；根据输入的补考成绩更新该学生的成绩。

//该函数定义中需要用到 <cmath> 头文件
//使用该函数定义时需要将 main 函数中 Student ph_2021[N]; 声明放在函数外改为全局变量声明

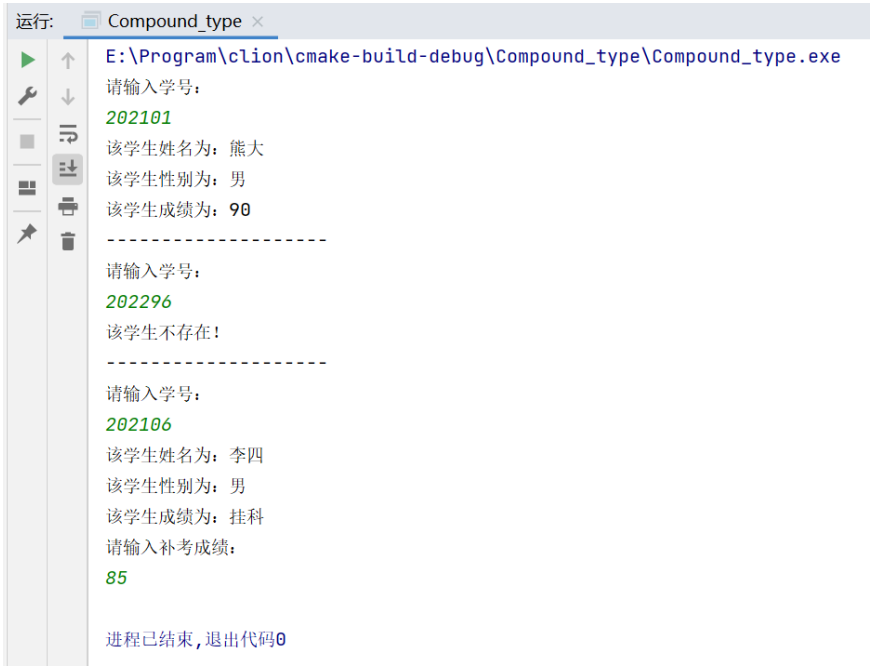
```
int info_output()
{
    int i, stu_number;
    int j = -1; //用于判断学生信息是否存在
    cout << "请输入学号: " << endl;
    cin >> stu_number;

    for (i = 0; i <= N; i++)
    {
        if (fabs(stu_number - ph_2021[i].no) < 1.0e-6)
        {
            cout << "该学生姓名为: " << ph_2021[i].name << endl;
            cout << "该学生性别为: " << ph_2021[i].gender << endl;
            j = 1; //表示该学生信息存在
            if (ph_2021[i].stu_grade.fail == 'n')
            {
                cout << "该学生成绩为: " << "挂科" << endl;
                // 学生补考成绩更新
                cout << "请输入补考成绩: " << endl;
                cin >> ph_2021[i].stu_grade.mark;
            }
            else
            {
                cout << "该学生成绩为: " << ph_2021[i].stu_grade.mark << endl;
            }
            break;
        }
    }

    if ( j < 0)
    {
        cout << "该学生不存在! " << endl;
    }

    return 0;
}
```

运行结果为



选做题：编写函数，计算不挂科同学成绩的平均分、方差；按照成绩从高到低输出学生成绩；

```
int calculate_grade()
{
    //统计所有不挂科学生的成绩
    int n = -1; //用于统计未挂科学生的数量
    float grades[5]; //用于存放未挂科学生的成绩

    for (int i = 0; i <= N-1; i++)
    {
        if (ph_2021[i].stu_grade.fail == 'n')
        {
            continue;
        }
        else
        {
            n++; //统计未挂科学生数量(n+1)
            grades[n] = ph_2021[i].stu_grade.mark;
        }
    }

    //计算平均值和方差
    float sum1 = 0; //各元素累加
    float sum2 = 0; //各元素的平方累加
    float aver, vari; //声明均值、方差

    for (int i = 0; i <= n; i++)
    {
        sum1 += grades[i];
        sum2 += (grades[i] * grades[i]);
    }
    aver = sum1 / float (n + 1);
    vari = sum2 / float (n + 1) - aver * aver;
    cout << "学生成绩的平均值为: " << aver << endl;
    cout << "学生成绩的方差为: " << vari << endl;
```

```

//冒泡排序法
float *p = grades;

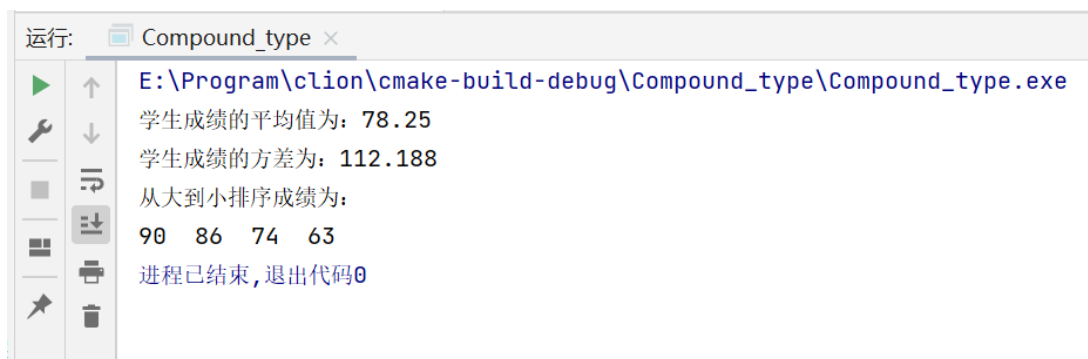
for(int i = n; i >= 0; i--)
{
    for(int j = 0; j < i; j++)
    {
        if (*(p + j) < *(p + j + 1))
        {
            float temp = *(p + j);
            *(p + j) = *(p + j + 1);
            *(p + j + 1) = temp;
        }
    }
}

//从大到小打印成绩
cout << "从大到小排序成绩为: " << endl;
for(int k = 0; k <= n; k++)
{
    cout << grades[k] << "\t";
}

return 0;
}

```

运行结果为：



```

运行: Compound_type x
E:\Program\clion\cmake-build-debug\Compound_type\Compound_type.exe
学生成绩的平均值为: 78.25
学生成绩的方差为: 112.188
从大到小排序成绩为:
90 86 74 63
进程已结束,退出代码0

```