**实验十二 模块化设计**

1. **实验目的和要求**

设 计 一 个 学 生 成 绩 管 理 程 序 ， 实 现 对 n 个 学 生 的 m 门 课 程 的 成 绩 的 记 录 与 统 计 工 作 。

学 生 信 息 包 括 ： 学 号 、 姓 名 、 数 学 成 绩 、 英 语 成 绩 、 物 理 成 绩 。

程 序 基 本 功 能 要 求 如 下 ：

a. 能 够 新 增 学 生 信 息 ， 并 计 算 总 分 和 平 均 分 。

b. 能 够 分 别 根 据 学 号 和 姓 名 查 询 该 学 生 的 基 本 信 息。

c. 能 够 根 据 学 号 修 改 某 个 学 生 的 信 息

d. 能 够 显 示 所 有 学 生 的 成 绩 信 息

e. 能 够 分 别 按 照 学 号 和 总 分 进 行 排 序 。

程 序 运 行 时 ， 出 现 菜 单 ， 形 式 如 下 ：

欢 迎 使 用 成 绩 管 理 系 统 ！

1 一 添 加 学 生 信 息

2一 修 改 学 生 信 息

3 一 显 示 全 部 学 生 信 息

4 一 按 学 号 查 询 学 生 信 息

5 一 按 姓 名 查 询 学 生 信 息

6 一 按 学 号 升 序 排 序

7 一 按 总 分 降 序 排 序

0 一 退出

**二、实验内容**

1. 实验准备
   1. 理论知识介绍
      1. 数据的输入：

要从用户依次读入int类型的变量a,n，可以使用如下语句:

cin >> a>>n;

* + 1. 数据的输出

要将变量的内容显示在显示器上，可以使用cout和流插入运算符。Cout还可以输出表达式的执行结果，比如:

cout << a << ‘+’ << b << ‘=’ << a+b << endl;

* + 1. 算术表达式

C++中，算术运算里有+（加法），-（减法），/（除法），%（取模），但是没有取平方算数运算。

* + 1. 数据类型 （整型）：

C++中一个整型变量可以储存一个整数，可以直接通过cin和cout输入和输出。

* + 1. 关系表达式

关系表达式用于比较两个值的大小。C++提供了6个关系运算符: <, <=, >, >=,

==, !=。关系运算符左结合。

* + 1. for语句

**for (i = 0; i < n; ++i)**

**需要重复执行的语句**

* + 1. while语句

**while (表达式) {**

**需要重复执行的语句**

**}**

* + 1. switch 语句

**switch (控制表达式) {**

**case 常量表达式1: 语句1； break;**

**case 常量表达式2: 语句2； break;**

**…**

**case 常量表达式n: 语句n； break;**

**default: 语句n+1;**

**}**

* + 1. 顺序查找

从数组的第一个元素开始，依次往下比较，直到找到要找的元素，输出元素的储存位置。

* + 1. 冒泡排序法

冒泡排序法是一种常用的排序算法，它通过调整违反次序的相邻元素的位置达到排序的目的。

* + 1. 字符串

由一系列字符组成的一个处理单元称为字符串，要会使用字符串需要知道字符串的储存及初始化，字符串的输入和输出以及字符串处理函数。

* + 1. 函数

编写一个实现某个功能的函数称为函数定义。一旦定义了一个函数，在程序中就可以反复调用这个函数。函数的定义形式如下：

**类型名 函数名 (形式参数表)**

**{**

**变量定义部分**

**语句部分**

**}**

* + 1. 结构体

我们需要能够将一组无序的、异质的数据看作一个整体进行操作。在程序设计语言中，这样的一组数据被称为记录，在C++中，记录被称为结构体。

* + 1. 模块化开发

随着要解决的问题越来越大，要解决的问题越来越复杂，程序设计开始了结构化的思想，集体而言，结构化程序设计的基本思想是在设计阶段采用“自顶向下，逐步求精”的方法，将大问题划分成一系列凶小问题，把小问题再进一步划分，知道问题小道可以直接写出解决办法。

1. 实验项目
   1. 分析
      1. 主程序刚开始应该输出使用说明，并进入一个while循环
      2. 这个成绩管理系统有多个功能，所以可以将每个功能设计成一个函数
      3. 每个学生应该作为一个结构体，然后用数组来将所有学生包含起来
   2. 方案
      1. 设计一个学生类型的结构体，其中的信息包括学生的学号，姓名，数学成绩，英语成绩，物理成绩，总成绩。
      2. 初始化一个包含多个学生的学生数组。
      3. main函数里开始时用cout输出使用说明，进入一个while循环，根据用户的不同的输入来执行不同的函数
      4. 一共提供6个函数
      5. 查找采用顺序查找
      6. 排序采用冒泡排序
   3. 测试数据、运行结果

欢 迎 使 用 成 绩 管 理 系 统 ！

1--添 加 学 生 信 息

2--修 改 学 生 信 息

3--显 示 全 部 学 生 信 息

4--按 学 号 查 询 学 生 信 息

5--按 姓 名 查 询 学 生 信 息

6--按 学 号 升 序 排 序

7--按 总 分 降 序 排 序

0--退 出

1

依 次 输 入 学 生 的 学 号 、 姓 名 、 数 学 成 绩 、 英 语 成 绩 、 物 理 成 绩

002 李四 60 60 60

欢 迎 使 用 成 绩 管 理 系 统 ！

1--添 加 学 生 信 息

2--修 改 学 生 信 息

3--显 示 全 部 学 生 信 息

4--按 学 号 查 询 学 生 信 息

5--按 姓 名 查 询 学 生 信 息

6--按 学 号 升 序 排 序

7--按 总 分 降 序 排 序

0--退 出

1

依 次 输 入 学 生 的 学 号 、 姓 名 、 数 学 成 绩 、 英 语 成 绩 、 物 理 成 绩

001 张三 70 70 70

欢 迎 使 用 成 绩 管 理 系 统 ！

1--添 加 学 生 信 息

2--修 改 学 生 信 息

3--显 示 全 部 学 生 信 息

4--按 学 号 查 询 学 生 信 息

5--按 姓 名 查 询 学 生 信 息

6--按 学 号 升 序 排 序

7--按 总 分 降 序 排 序

0--退 出

1

依 次 输 入 学 生 的 学 号 、 姓 名 、 数 学 成 绩 、 英 语 成 绩 、 物 理 成 绩

003 王五 80 80 80

欢 迎 使 用 成 绩 管 理 系 统 ！

1--添 加 学 生 信 息

2--修 改 学 生 信 息

3--显 示 全 部 学 生 信 息

4--按 学 号 查 询 学 生 信 息

5--按 姓 名 查 询 学 生 信 息

6--按 学 号 升 序 排 序

7--按 总 分 降 序 排 序

0--退 出

3

002 李 四 60 60 60

001 张 三 70 70 70

003 王 五 80 80 80

欢 迎 使 用 成 绩 管 理 系 统 ！

1--添 加 学 生 信 息

2--修 改 学 生 信 息

3--显 示 全 部 学 生 信 息

4--按 学 号 查 询 学 生 信 息

5--按 姓 名 查 询 学 生 信 息

6--按 学 号 升 序 排 序

7--按 总 分 降 序 排 序

0--退 出

2

1--按 学 号 修 改 学 生 信 息

2--按 姓 名 修 改 学 生 信 息

2

请 输 入 要 修 改 的 学 生 的 姓 名

张三

要 修 改 哪 些 内 容 ？

1--学 号

2--姓 名

3--数 学 成 绩

4--英 语 成 绩

5--物 理 成 绩

0--退 出

5

输 入 新 的 物 理 成 绩

99

要 修 改 哪 些 内 容 ？

1--学 号

2--姓 名

3--数 学 成 绩

4--英 语 成 绩

5--物 理 成 绩

0--退 出

0

欢 迎 使 用 成 绩 管 理 系 统 ！

1--添 加 学 生 信 息

2--修 改 学 生 信 息

3--显 示 全 部 学 生 信 息

4--按 学 号 查 询 学 生 信 息

5--按 姓 名 查 询 学 生 信 息

6--按 学 号 升 序 排 序

7--按 总 分 降 序 排 序

0--退 出

3

002 李 四 60 60 60

001 张 三 70 70 99

003 王 五 80 80 80

欢 迎 使 用 成 绩 管 理 系 统 ！

1--添 加 学 生 信 息

2--修 改 学 生 信 息

3--显 示 全 部 学 生 信 息

4--按 学 号 查 询 学 生 信 息

5--按 姓 名 查 询 学 生 信 息

6--按 学 号 升 序 排 序

7--按 总 分 降 序 排 序

0--退 出

4

请 输 入 要 查 询 的 学 号

03

欢 迎 使 用 成 绩 管 理 系 统 ！

1--添 加 学 生 信 息

2--修 改 学 生 信 息

3--显 示 全 部 学 生 信 息

4--按 学 号 查 询 学 生 信 息

5--按 姓 名 查 询 学 生 信 息

6--按 学 号 升 序 排 序

7--按 总 分 降 序 排 序

0--退 出

4

请 输 入 要 查 询 的 学 号

003

003 王 五 80 80 80

欢 迎 使 用 成 绩 管 理 系 统 ！

1--添 加 学 生 信 息

2--修 改 学 生 信 息

3--显 示 全 部 学 生 信 息

4--按 学 号 查 询 学 生 信 息

5--按 姓 名 查 询 学 生 信 息

6--按 学 号 升 序 排 序

7--按 总 分 降 序 排 序

0--退 出

6

欢 迎 使 用 成 绩 管 理 系 统 ！

1--添 加 学 生 信 息

2--修 改 学 生 信 息

3--显 示 全 部 学 生 信 息

4--按 学 号 查 询 学 生 信 息

5--按 姓 名 查 询 学 生 信 息

6--按 学 号 升 序 排 序

7--按 总 分 降 序 排 序

0--退 出

3

001 张 三 70 70 99

002 李 四 60 60 60

003 王 五 80 80 80

欢 迎 使 用 成 绩 管 理 系 统 ！

1--添 加 学 生 信 息

2--修 改 学 生 信 息

3--显 示 全 部 学 生 信 息

4--按 学 号 查 询 学 生 信 息

5--按 姓 名 查 询 学 生 信 息

6--按 学 号 升 序 排 序

7--按 总 分 降 序 排 序

0--退 出

7

欢 迎 使 用 成 绩 管 理 系 统 ！

1--添 加 学 生 信 息

2--修 改 学 生 信 息

3--显 示 全 部 学 生 信 息

4--按 学 号 查 询 学 生 信 息

5--按 姓 名 查 询 学 生 信 息

6--按 学 号 升 序 排 序

7--按 总 分 降 序 排 序

0--退 出

3

003 王 五 80 80 80

001 张 三 70 70 99

002 李 四 60 60 60

欢 迎 使 用 成 绩 管 理 系 统 ！

1--添 加 学 生 信 息

2--修 改 学 生 信 息

3--显 示 全 部 学 生 信 息

4--按 学 号 查 询 学 生 信 息

5--按 姓 名 查 询 学 生 信 息

6--按 学 号 升 序 排 序

7--按 总 分 降 序 排 序

0--退 出

2

1--按 学 号 修 改 学 生 信 息

2--按 姓 名 修 改 学 生 信 息

001

请 输 入 要 修 改 的 学 生 的 学 号

001

要 修 改 哪 些 内 容 ？

1--学 号

2--姓 名

3--数 学 成 绩

4--英 语 成 绩

5--物 理 成 绩

0--退 出

3

输 入 新 的 数 学 成 绩

99

要 修 改 哪 些 内 容 ？

1--学 号

2--姓 名

3--数 学 成 绩

4--英 语 成 绩

5--物 理 成 绩

0--退 出

0

欢 迎 使 用 成 绩 管 理 系 统 ！

1--添 加 学 生 信 息

2--修 改 学 生 信 息

3--显 示 全 部 学 生 信 息

4--按 学 号 查 询 学 生 信 息

5--按 姓 名 查 询 学 生 信 息

6--按 学 号 升 序 排 序

7--按 总 分 降 序 排 序

0--退 出

7

欢 迎 使 用 成 绩 管 理 系 统 ！

1--添 加 学 生 信 息

2--修 改 学 生 信 息

3--显 示 全 部 学 生 信 息

4--按 学 号 查 询 学 生 信 息

5--按 姓 名 查 询 学 生 信 息

6--按 学 号 升 序 排 序

7--按 总 分 降 序 排 序

0--退 出

3

001 张 三 99 70 99

003 王 五 80 80 80

002 李 四 60 60 60

欢 迎 使 用 成 绩 管 理 系 统 ！

1--添 加 学 生 信 息

2--修 改 学 生 信 息

3--显 示 全 部 学 生 信 息

4--按 学 号 查 询 学 生 信 息

5--按 姓 名 查 询 学 生 信 息

6--按 学 号 升 序 排 序

7--按 总 分 降 序 排 序

0--退 出

* 1. 异常现象
     1. 由于在主程序中，储存用户选项的是一个整型opt，所以如果输入其他字符的话，会出现难以预料的异常，其中一种异常就是会一直输出使用说明而不能执行任何命令。
     2. 开始的时候程序修改学生的信息，查询信息，排序都会出现异常，会无法找到学生的信息等等。
  2. 分析
     1. 为了解决第一个问题，我们需要将opt这个变量修改为一个字符类型或者字符串的类型。
     2. 第二个问题我们最终发现原来是因为字符串的比较的问题。因为刚开始字符串的比较是通过s1 == s2, s1 > s2 等等来进行比较的，实际上这样是不对的，要通过字符串的专门的处理函数才能进行有效的比较。
  3. 总结

用字符串类型接受用户输入相对整型类型来说更灵活。字符串的操作要使用专门的操作的函数。

**三、实验小结**

1. 完成情况

已经完成了所有要求的功能，但是程序的健壮性还不够好。

1. 重点

整个程序的结构化的设计过程和思想。

1. 难点

采用结构化设计的时候各种变量的作用范围难以确定，难以确定某个变量在一个作用范围内的值到底是什么。

1. 解决方法

在特定的作用范围内给予变量新的有意义的名称而不是一直沿用全局变量。

1. 有待改进之处

将opt这个变量设计成一个字符串或者字符，这样的话程序的健壮性就会增加，整个程序不容易出错。

1. 收获、体会

程序的结构化的设计的思想非常重要，我们不能在刚开始的时候就拘泥于程序中的实现的细节，这样的话很可能最后会产生程序根本无法运行的结果，可能会出现大量的坑要我们去填。而采用结构化的程序设计能让我们的程序的结构更加清晰，具体实现的细节可以之后再做。