**实验十一　枚举/结构体/联合体**

|  |
| --- |
| 实验目标 |
| 1. 掌握枚举/结构体/联合体的定义及初始化 2. 掌握三者的访问方法 3. 能在较为复杂的工程项目中恰当使用 |

|  |
| --- |
| E-easy必做 N-normal必做 H-hard选做 |

1. **结构体和枚举的定义 E**

请为保存学生记录定义一个结构体，包括：学号（1000~1999整型），姓名（20个字符以内），生日（单独定义一个结构体，并且月份请用枚举），性别（字符），成绩（整型数组scores[3]）。

1. **结构体的访问 E**

继续问题0，用该结构体定义一个变量stu1，并在主函数初始化该变量，并输出。

1. **结构体作为参数传递 E**

继续问题1，将初始化工作和输出工作通过定义两个函数实现，并在主函数中调用这2个函数。学号请用随机整数进行初始化；姓名请用1个大写字母+3个小写字母构成，字母均随机产生；生日为随机数据；性别为随机数据；成绩请随机产生。

* 1. 问题分析：姓名构成:name[0]=’A’+rand（）%26，name[1]=’a’ +rand（）%26…

性别构成：字符数组char gender[2]={‘F’,’M’};

gender[rand()%2]

1. **学生成绩管理软件 H**

作为一名大学生，不论是报到和缴费，还是选课没有哪一样能离开管理学生信息的软件。虽然这些软件的具体功能不同，但都有一些基本的功能供管理者使用，比如增删学生记录，修改学生记录，查找记录和排序。下面让我们用学过的知识来模拟部分功能，看看才上了区区若干节编程课的同学，能不能做出个麻雀虽小，五脏俱全的“产品”来！

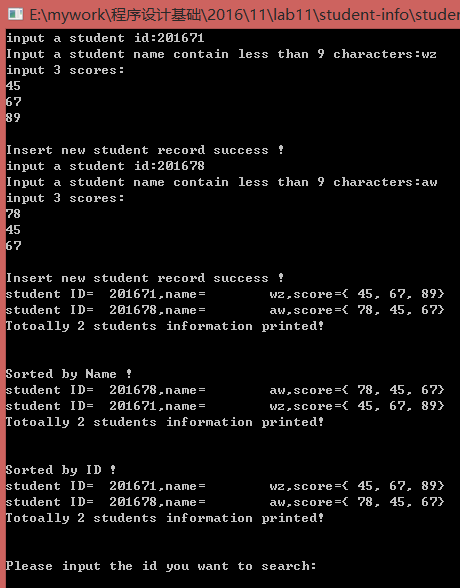
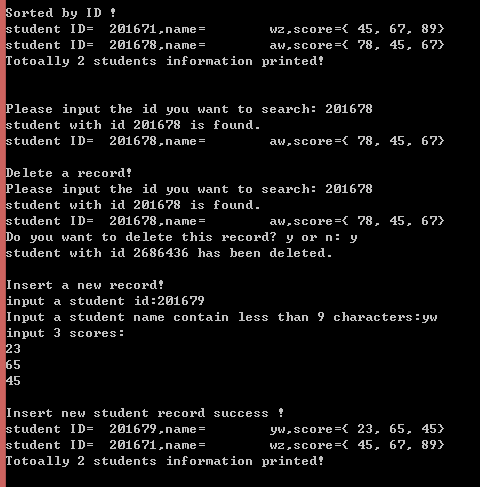
在练习题2的基础上，用结构体定义出结构体数组（例如含5个元素），作为学生成绩记录，随后分模块实现以下功能，并于主函数中依次调用（也可以创建菜单调用）：

（1）插入新学生记录

（2）按学号查找学生记录：注意查找暗含排序

（3）按学号删除学生记录

部分程序示例如下：



* 1. 问题分析：
     1. 如果说要设计成menu方式，那么需要参考ATM机的具体设计方案。
     2. 如果要想做的简单，不调用菜单，则只需要在main中依次调用3个功能模块，表明程序所有功能都可以正常执行。
  2. 开发模式：由于程序本身涉及的功能模块比较多，建议采用增量式开发，增量的顺序例举如下：

（0）菜单（可选）；（1）录入和显示；（2）排序；（3）查找；（4）删除

* 1. 详细设计步骤（无菜单模式）：
     1. 增量1：录入和显示

1. 定义结构体
2. 主函数中定义结构体数组：考虑数组的定义方式：长度是否确定，动态还是静态；为了简便，此处用了静态，即定义一个较大的数组；为了便于测试，最初可以将数组长度设置为2
3. 主函数中定义一个变量记录实际录入的学生数目
4. 函数：录入一条新记录。读入一个数据，存放于数组中，暂不考虑排序问题
5. 函数：输出显示。**设计小窍门：代码复用**

复用性：http://baike.baidu.com/link?url=CKHkALqgdFXWnPYL\_YJBmlOE2aq9w-Al521nP\_YhpFkyqYFGK-aBKYURVrM3ZI5QySODk8NrctZCbIUbXIeRda

软件复用：

http://baike.baidu.com/link?url=iA-Nmti4AtYFk86\_mjttj-nK9Gjcy\_fbe92TM28S5QQm\_Z8rBSyywabpxXz6gsmM

为了改善程序的复用性，在输出显示部分，我们用2个函数实现，第一个函数输出单条记录，即第i条记录；第二个函数输出所有记录，即通过循环调用第一个函数。这样做的好处在于，当我们调用查询和删除功能时，需要显示单条记录时，可以复用第一个函数。

* + 1. 增量2：排序

1. 排序显然是单独的一个函数：按学号排序，可以参考以前的任何一个排序算法，将其中被比较的对象换为stu[i].ID
2. 需要特别注意的是，数组中存在空白数据，如何区别有效记录和空白或无效记录，需要在程序中予以约定。比如空白记录的学号为0，这一约定将需要你在数组定义后立刻初始化所有元素，并且每删除一条记录，就需要将其学号设置为0。
3. 排序函数在什么时候调用呢？这时候需要考虑程序的执行速度。软件性能<http://baike.baidu.com/view/1812806.htm>，

排序可以被调用的2个地方：

* 为了提高时间性能，排序应该只在必要的时候调用，比如查询前或者输出全部数据时；**优点：**此方法中排序调用的次数小于第二种方法。具体执行过程中，需要设置一个“是否需要更新”的标志变量，每次调用“录入新记录”函数后，更新标准被设置为有效，比如1或者true，在输出函数或者查找函数被调用时，先判断是否需要更新，如果是，调用排序函数，否则不排序；排序后将更新标志置为0或无效。**缺点：**以后程序新增加其它功能时，需要考虑排序函数的调用问题，容易导致导致排序的疏漏。
* 为了统一程序的设计风格，排序应该只在某个位置固定调用，比如每次读入新数据或删除数据时，排序作为读入新数据函数的最后一条指令；**缺点：**显然当连续录入多条记录时，此方法的效率不高，因为排序会反复执行；**优点：**此方法最简单易行，不会导致排序的疏漏。
  + 1. 增量3：按学号查找

此功能很简单，读入学号；对已经排序的数组，用最简单的顺序查找算法或者效率更高的二分查找方法；一旦找到，调用打印单条信息函数。

* + 1. 增量4：按学号删除

此功能较为复杂，包含5个步骤：读入学号；调用查找函数；询问是否删除；如果确认删除，将该记录赋值为0；考虑是否重新排序问题（具体方法需要结合增量2中，你选择的具体排序方案）。

* + 1. 增量5:按姓名排序，在排序函数中仅仅修改比较指令，由

if (stu[walker].nID < stu[walker - 1].nID) 改为

if (strcmp(stu[walker].name,stu[walker - 1].name)<0)

* + 1. 增量6:按姓名查找，对比按学号查找，修改比较多，依次为：读入数据指令，if比较指令，输出指令
    2. 增量7:按姓名删除，此功能修改最为简单，将原有的查询函数调用指令修改，并且修改输出删除是否成功的2条printf指令，其一如下。

printf("student with id %d has been deleted.\n",id);

