**学生信息管理系统——软件设计说明书**

**班级：软件学院xxxx**

**姓名：xxx**

**学号：xxx**

# 设计思想与设计原则

## 设计思想

1. 按工序将问题化简，将功能的实现与设计分开，便于分工协作，即采用结构化的分析与设计方法将逻辑实现与物理实现分开。将软件生命周期划分为制定计划、需求分析、软件设计、程序编写、软件测试和运行维护等六个基本活动，并且规定了它们自上而下、相互衔接的固定次序，如同瀑布流水，逐级下落。从系统需求分析开始直到产品发布和维护，每个阶段都会产生循环反馈，如果有信息未被覆盖或者发现了问题，那么最好 “返回”上一个阶段并进行适当的修改，开发进程从一个阶段“流动”到下一个阶段。
2. 先建造一个快速原型，实现客户或未来的用户与系统的交互，用户或客户对原型进行评价，进一步细化待开发软件的需求。通过逐步调整原型使其满足客户的要求，开发人员可以确定客户的真正需求是什么；第二步则在第一步的基础上开发客户满意的软件产品。
3. 软件设计的本质就是针对软件的需求，建立模型，通过将模型映射为软件，来解决实际问题。因此对于我们的软件而言，我们要求其具有以下特性：

1灵活性

2有效性

3可靠性

4可理解性

5维护性

6重用性

7适应性

8可移植性

9可追踪性

10互操作性

## （二）设计原则

A) 开闭原则：对扩展开放，对修改关闭。在程序需要进行拓展的时候，不能去修改原有的代码，实现一个热插拔的效果。简言之，是为了使程序的扩展性好，易于维护和升级。想要达到这样的效果，我们需要使用接口和抽象类。

B）接口隔离原则： 使用多个隔离的接口，而不使用单个接口。降低依赖，降低耦合。

C）迪米特原则：  一个实体应当尽量少地与其他实体之间发生相互作用，使得系统功能模块相对独立。

D）依赖倒转原则：针对接口编程，依赖于抽象而不依赖于具体。

# 设计规范

## 基本规范

采用C语言做为开发语言。

系统基于Windows操作系统平台，采用Code Blocks做为开发工具，采用 process on 做为流程图和界面图的设计工具。

管理员账号数据采用二进制保存，而学生信息数据采用文本模式保存，文件位置固定。

系统不驻留内存，单机独占作业。

## 界面规范

采用文本界面。通过定义不同的界面分区，取得相对一致的功能分区。

借助Windows系统功能，实现文本、色彩的控制功能。

## 代码规范

系统源代码要求书写规范，结构清晰，注释完善。

变量定义要求：

1. 模块内的全局变量必须加static限制；原则上不采用外部全局变量直接使用，而通过接口封装的方式提供使用,特殊情况除外；
2. 变量和函数的命名，都以大写英文开始，遵循“两个基本点”的原则，由于主要有两个子系统构成，故各函数变量均以 xxx1 和 xxx2 构成，即以数字递增结尾；
3. 模块内的函数必须加static限制，模块内提供给其他模块使用的函数，通过头文件方式提供；
4. 结构体类型定义采用大写及“\_”组合，同样遵循数字递增规律。

# 数据结构与算法设计

## 登录

采用心形算法，输出心形，优化界面。

登录使用伪登录，即使用检索本地二进制文件以核对账户密码完成登录，对于数据文件采用用户名加拼接的形式作为对应文件名。



## 学生基本信息管理系统

鉴于对学号的顺序排序，我们采用一颗AVL树以保存数据，能够相对的提高程序的性能。

鉴于AVL树的几个基本旋转以实现插入。

对于每一个删除操作，均需调整AVL树的平衡，由于删除操作开销较大，我们可以选取左右子树较高的树中的最右或最左树以替代要删除的数据，这样可以明显减少时间平均复杂度。

代码中多见递归操作。

## 学生成绩管理系统

鉴于需要对总成绩以及各科成绩查询，寻常的排序需要花费大量的时间，十分不值，并且代码及其冗长，因此我们采用多重链表对数据进行操作，使用多个表头对应每一种成绩。这样做的缺点是查找和插入效率可能不如AVL树，但对于多根节点AVL树而言，删除一个数据所需要付出的代价是昂贵的，所以我们选择链表实现。

## 更多细节算法

更多细节算法见大作业实验报告中。

# 模块设计

系统功能参见“学生信息管理系统软件需求说明书”。

## 写在前面

### 1.开发者的话

由于时间比较仓促、我所掌握的专业技能有限、社会经验不足、项目完成经验不足，本系统功能肯定有所缺陷，对此深表遗憾，很抱歉我们需要作出一些假定和约束。

### 假定和约束

**①：所有数据存档或读取均以文件为操作对象。**

**②：在此系统上仅支持键盘键入，并不支持鼠标点击。**

**③：所有信息的输入都有一个要求，您的输入不应该越过这个范围或违背要求，后续我们都会说明这个要求。**

**④：我们假定一个英文输入为1个长度，而中文输入被视为2个长度。**

**⑤：我们规定，所有相关文件均不能移动至别的路径。**

## 操作员管理功能

### 功能描述

管理员应当创建账号，或使用账号密码登陆系统，登陆之后系统应从账户中读取相应数据，并且在管理员保存数据之后，应将当前数据继续存入管理员账号当中，系统应支持修改密码与销毁账户。

### 数据来源

①：登陆数据来源于管理员键入的账号和密码。

②：操作数据来源于管理员个人账号中所保存的数据。

### 数据流向

①：登陆数据将流向一个专门的本地文件中，便于检索。

②：操作数据将流向管理员个人账号中

### 4.数据要求

在此我们仅说明管理员账号与密码的要求。

我们假定账号密码均只能使用英文字母与数字搭配的方式，为了更加安全，它们都应该在6~12位之间，对于不符合要求的输入，系统会对此做出提示。

### 5.功能实现

管理员键入帐号密码，系统会从本地文件检索账号密码，如果管理员选择创建账号，那么系统会将账号密码保存至本地文件中，以便下次登陆时检索，并且系统会以管理员账号名创建额外两个文件，分别用以存储学生基本信息和学生成绩信息。

登陆成功后，系统从本地文件中获取数据，管理员可以对这些数据进行操作，在管理员操作完毕选择保存数据之后，系统将数据继续存入相应的文件之中，以便下次操作。

管理员修改密码时，系统查阅本地文件，并修改数据，注销时，系统删除存储在本地文件中的数据。

### 6.业务处理

管理员可以登陆账号以管理学生信息，每个账号中都保存着相应的数据，管理员可以操控管理这些数据。管理员可以修改自己的密码，或者销毁自己的账户。

## 学生基本信息管理系统

### 基本信息维护

#### 功能描述

包括对学生学号、姓名、联系方式以及宿舍号的设置。四种设置均应实现添加、删除及修改操作。

#### 数据来源

**①：数据来源于管理员的输入。**

**②：数据来源于管理员账号中的读取。**

#### 数据流向

**①：数据会暂时保存于系统中，当管理员存档时，数据将会被保存至系统中供管理员查阅。**

**②：管理员存档时数据自动保存至管理员相应账户文件中。**

#### 数据要求

学生学号、姓名、联系方式以及宿舍号均要求在52长度之内，这不包括52本身，除此之外，学生学号不允许重复，对于不符合要求的输入，系统会给予提醒。

#### 功能实现

1. **添加：**管理员键入学生信息，这必须要符合数据要求，系统读取信息，并将其暂时存放在数据结构中。

②**删除：**管理员键入要删除学生的学号，系统自动查询，查询成功则删除，否则返回提示--“要删除的学生不存在”。

③**修改：**管理员键入要修改学生的学号，系统自动查询，查询成功则按提示修改数据，否则返回提示--“要修改的学生不存在”**。**

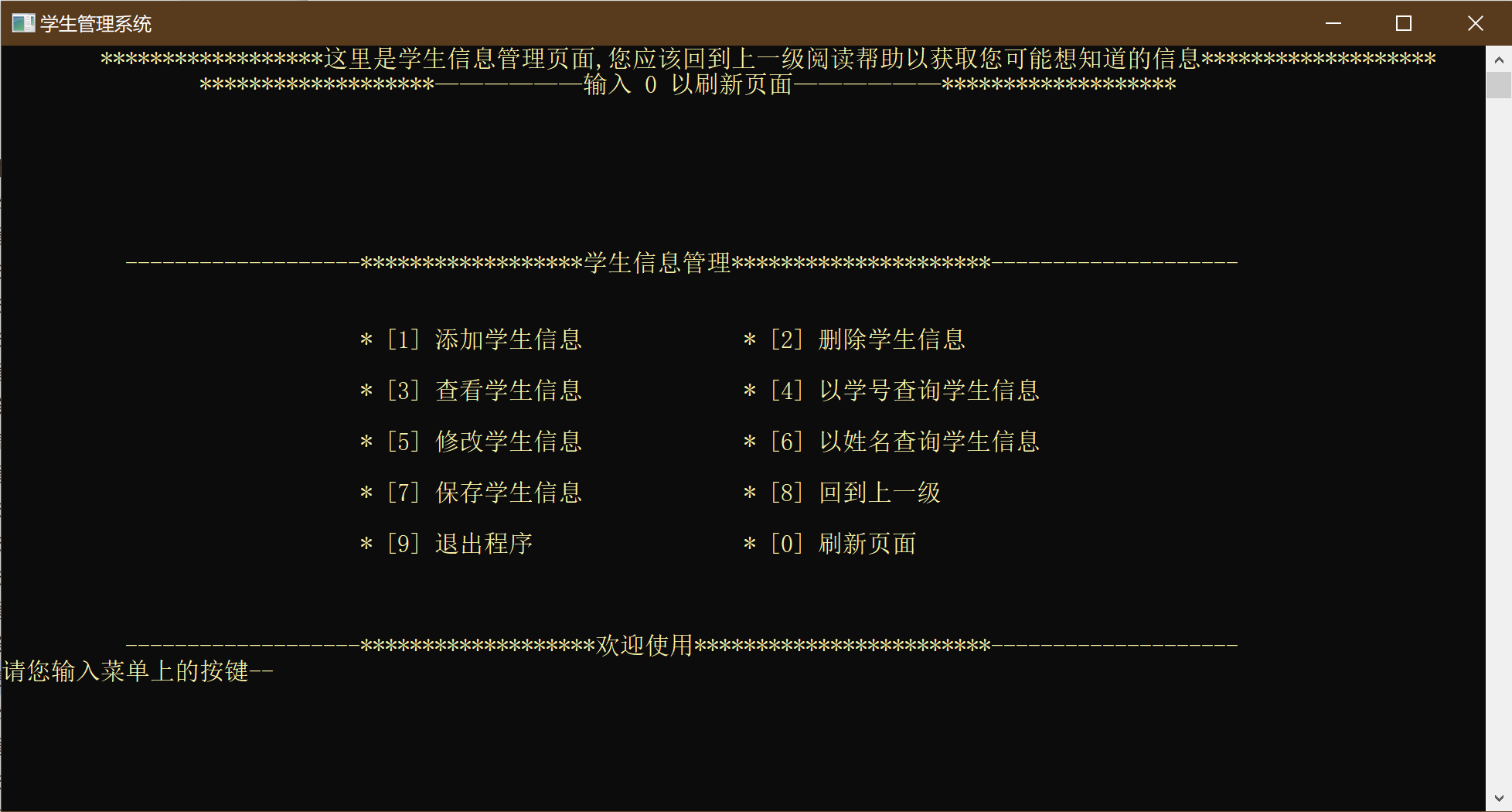
#### 业务处理

管理员有义务维护好学生的基本信息，管理员可以通过添加、删除和修改的基本操作对学生的基本信息进行维护。

### 学生基本信息管理

#### 功能描述

包括对学生信息的添加、修改、删除、查询、浏览和保存等操作，在管理的同时，管理员应做到对学生基本信息的维护。



#### （2）数据来源

**①：数据来源于管理员的输入。**

**②：数据来源于管理员账号中的读取。**

#### （3）数据流向

**①：数据会暂时保存于系统中，当管理员存档时，数据将会被保存至系统中供管理员查阅。**

**②：管理员存档时数据自动保存至管理员相应账户文件中。**

#### （4）数据要求

学生学号、姓名、联系方式以及宿舍号均要求在52长度之内，这不包括52本身，除此之外，学生学号不允许重复，对于不符合要求的输入，系统会给予提醒。

#### （5）功能实现

①**添加**：管理员键入学生信息，这必须要符合数据要求，系统读取信息，并将其暂时存放在数据结构中。

②**删除**：管理员键入要删除学生的学号，系统自动查询，查询成功则删除，否则返回提示--“要删除的学生不存在”。

③**修改**：管理员键入要修改学生的学号，系统自动查询，查询成功则按提示修改数据，否则返回提示--“要修改的学生不存在”。

④**查询**：管理员可以选择以学号查询或是以姓名查询，管理员键入要修改学生的学号或姓名，系统自动查询，查询成功则显示数据，否则返回提示--“要查询的学生不存在”。

⑤**浏览**：系统遍历所有的数据并且显示。

⑥**保存**：系统将数据保存至管理员账户中。

#### 业务处理

①**添加**：管理员按照学号、姓名、联系方式以及宿舍号的顺序依次录入，所录入的信息必须符合数据要求。

②**删除**：管理员输入要删除学生的学号，该学号必须存在，否则系统弹出提示，管理员可以选择一键删除。

③**修改**：管理员输入要修改学生的学号，然后管理员可以选择修改相应的信息，该学号必须存在，否则系统弹出提示。

④**查询**：管理员可以通过姓名查询，亦可以通过学号查询，查询成功则可以浏览该学生的信息，该学号或姓名必须存在，否则系统弹出提示。

⑤**浏览**：管理员可以浏览当前所存储的全部学生信息。

⑥**保存**：管理员在退出程序前或者退出登陆时，必须要保存数据，否则数据会丢失。

## （四）学生成绩管理系统

### 1.基本信息维护

#### （1）功能描述

包括对学生学号、姓名、高数成绩、英语成绩和C语言成绩的设置。六种设置均应实现添加、删除及修改操作。

#### （2）数据来源

**①：数据来源于管理员的输入。**

**②：数据来源于管理员账号中的读取。**

#### （3）数据流向

**①：数据会暂时保存于系统中，当管理员存档时，数据将会被保存至系统中供管理员查阅。**

**②：管理员存档时数据自动保存至管理员相应账户文件中。**

#### （4）数据要求

学生学号、姓名均要求在52长度之内，这不包括52本身，除此之外，学生学号不允许重复，对于不符合要求的输入，系统会给予提醒，对于成绩而言，仅支持十进制输入，支持小数输入，系统会自动屏蔽不符合规范的输入，如“q33!#”,其中q!#均会被自动屏蔽，对于其他不规范的输入，系统会给予提醒。

#### （5）功能实现

①**添加**：管理员键入学生成绩信息，这必须要符合数据要求，系统读取信息，并自动计算出3科平均成绩，并将其视为总成绩，这些数据会被暂时存放在数据结构中。

1. **删除**：管理员键入要删除学生的学号，系统自动查询，查询成功则删除，否则返回提示--“要删除的学生不存在”。
2. **修改**：管理员键入要修改学生的学号，系统自动查询，查询成功则按提示修改数据，否则返回提示--“要修改的学生不存在”。

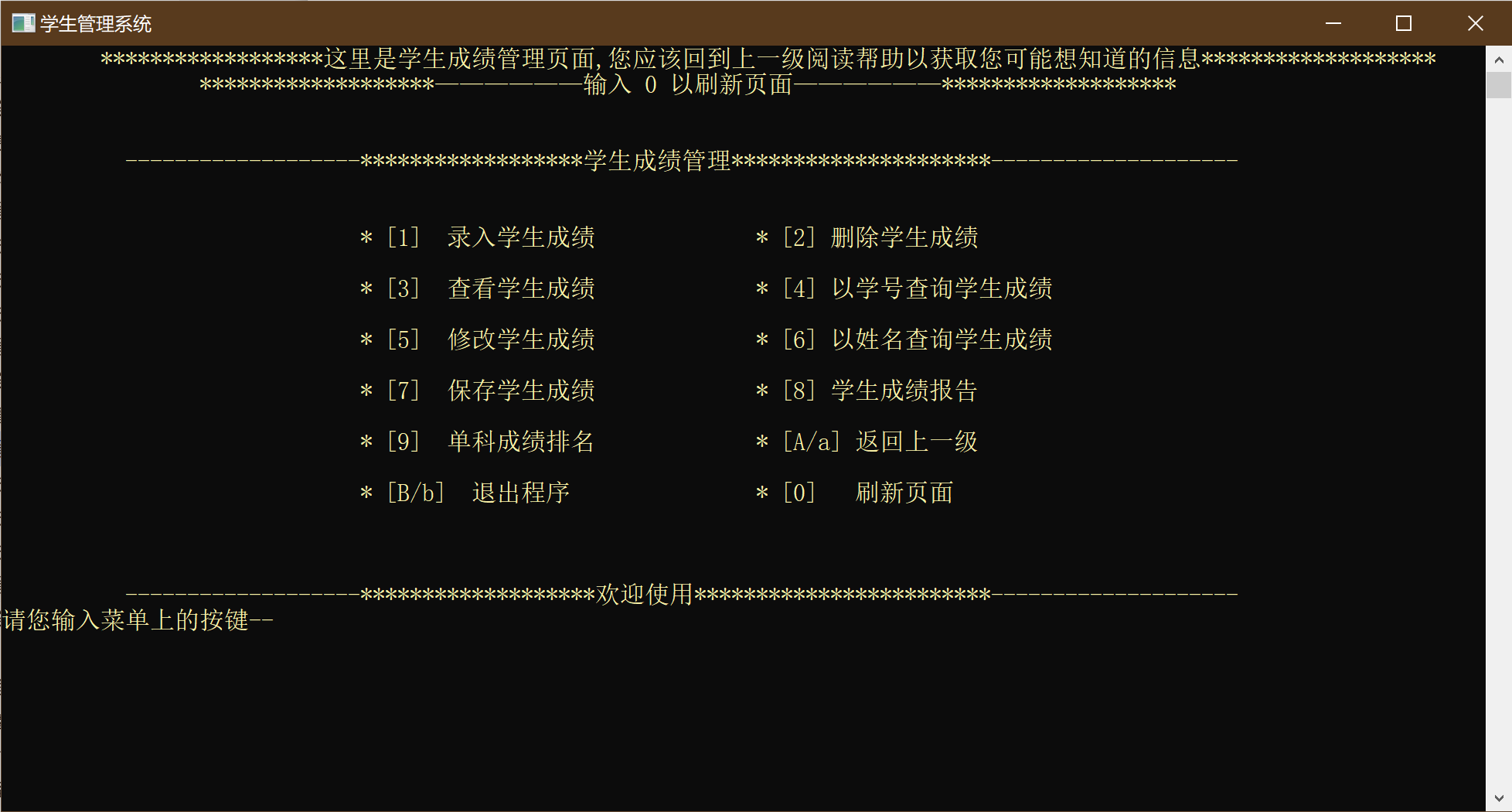
#### （6）业务处理

管理员有义务维护好学生的基本信息，管理员可以通过添加、删除和修改的基本操作对学生的成绩信息进行维护。

### 学生成绩管理

#### （1）功能描述

包括对学生成绩的添加、修改、删除、查询、浏览和保存等操作，在管理的同时，管理员应做到对学生基本信息的维护，此外管理员可以查看学生成绩的分析报告。



#### （2）数据来源

**①：数据来源于管理员的输入。**

**②：数据来源于管理员账号中的读取。**

#### （3）数据流向

**①：数据会暂时保存于系统中，当管理员存档时，数据将会被保存至系统中供管理员查阅。**

**②：管理员存档时数据自动保存至管理员相应账户文件中。**

#### （4）数据要求

学生学号、姓名均要求在52长度之内，这不包括52本身，除此之外，学生学号不允许重复，对于不符合要求的输入，系统会给予提醒，对于成绩而言，仅支持十进制输入，支持小数输入，系统会自动屏蔽不符合规范的输入，如“q33!#”,其中q!#均会被自动屏蔽，对于其他不规范的输入，系统会给予提醒

#### （5）功能实现

①**添加**：管理员键入学生成绩信息，这必须要符合数据要求，系统读取信息，并将其暂时存放在数据结构中。

②**删除**：管理员键入要删除学生的学号，系统自动查询，查询成功则删除，否则返回提示--“要删除的学生不存在”。

③**修改**：管理员键入要修改学生的学号，系统自动查询，查询成功则按提示修改数据，否则返回提示--“要修改的学生不存在”。

④**查询**：管理员可以选择以学号查询或是以姓名查询，管理员键入要修改学生的学号或姓名，系统自动查询，查询成功则显示数据，否则返回提示--“要查询的学生不存在”。

⑤**浏览**：管理员可以选择浏览以总成绩排名顺序分布的内容；可以浏览以单科成绩排名顺序分布的内容；亦可以浏览总体成绩报告分析。对于每一种需求，系统显示相应的数据。

⑥**保存**：系统将数据保存至管理员账户中。

#### 业务处理

①**添加**：管理员按照学号、姓名、高数成绩、大学英语成绩以及C语言成绩的顺序依次录入，所录入的信息必须符合数据要求。

②**删除**：管理员输入要删除学生的学号，该学号必须存在，否则系统弹出提示，管理员可以选择一键删除。

③**修改**：管理员输入要修改学生的学号，然后管理员可以选择修改相应的信息，该学号必须存在，否则系统弹出提示。

④**查询**：管理员可以通过姓名查询，亦可以通过学号查询，查询成功则可以浏览该学生的成绩信息，该学号或姓名必须存在，否则系统弹出提示。

⑤**浏览**：

1. **浏览总成绩**：管理员可以看到以总成绩从高到低的学生所有成绩信息。
2. **浏览单科成绩**：管理员可以选择具体的单科，并且可以看到以该单科成绩从高到低的学生单科成绩信息。
3. **浏览成绩报告分析**：管理员可以看到各科以及总成绩中60分一下人数占比，60分以上人数占比，以及90分以上人数占比，对于90分以上和60分以下的，管理员还可以看到他们的信息。

⑥**保存**：管理员在退出程序前或者退出登陆时，必须要保存数据，否则数据会丢失

# 四、运行需求

## （一）用户界面

碍于技术原因，我们无法做到拥有美丽的UI界面，我们只能尽可能的让其简单小巧，我们无法做到界面显示永远按界面大小比例显示，因此我们希望用户不要轻易改动运行界面的大小。

## （二）软硬件接口

暂无此需求。

## （三）运行性能

我们致力于让此系统更加流畅的运行，对此我们采纳了不同的数据结构的实现方式，尽管我们拼劲全力去维护性能，但其仍不可避免的会存在瑕疵。

## 运行设备

该系统仅支持PC端，暂不支持移动端，支持windows下极大多数系统。

## 设备故障

[如果在使用时遇到异常，可以将异常信息发送到邮件1637318597@qq.com中。](mailto:如果在使用时遇到异常，可以发送邮件到1637318597@qq.com)

我们会立刻进行修复。

# 六、其他需求与结束语

## 存储

介于技术原因，我们只能将数据保存至文件中，而不是数据库中，对于管理员的个人账户，我们将所有的账号密码信息均存储在当前路径下User.txt中，以更为安全的二进制方式存储，而对于学生基本信息内容数据，我们将保存在当前路径下对于的文件中，文件名将以 “用户账户” + “\_IF.txt”

为名，IF译为 information,由于我们假设用户名不存在\_，因此这个操作是可以实现的，并且无需担心重名问题，数据以文本格式存储，同理对于学生成绩信息，文件名将以 “用户账户” + “\_GD.txt”为名，GD译为grade。

## 仍需提高的问题

1. 应当支持鼠标点击
2. 应当将数据存储至数据库中
3. 页面可以继续优化
4. 操作仍需更加人性化
5. 更多需要完善的功能...

## 结束语

**此时做出来的系统仍上不得台面，距离前方还有很长一段路要走，路漫漫其修远兮，吾将上下而求索，继续努力！**