

## 概要设计说明书

### 1 引言

#### 1.1 编写目的

#### 1.2 背景

#### 1.3 定义

#### 1.4 参考资料

### 2 总体设计

#### 2.1 需求规定

#### 2.2 运行环境

#### 2.3 基本设计概念和处理流程

#### 2.4 结构

#### 2.5 功能需求与程序的关系

#### 2.6 人工处理过程

#### 2.7 尚未解决的问题

### 3 接口设计

#### 3.1 用户接口

#### 3.2 外部接口

#### 3.3 内部接口

### 4 系统数据结构设计

#### 4.1 逻辑结构设计要点

#### 4.2 物理结构设计要点

#### 4.3 数据结构与程序的关系

### 5 系统出错处理设计

#### 5.1 出错信息

#### 5.2 补救措施

#### 5.3 系统维护设计

## 概要设计说明书

### 1 引言

#### 1.1 编写目的

此份概要设计说明书是为了阐述学生管理设计原理与思路，为源码阅读者解释程序设计思路

## 1.2 背景

学生管理系统是科锐软件安全培训第一阶段C语言项目，实现管理学生信息功能。

## 1.3 定义

结构体：自定义的学生信息结构体

文件：存储学生信息的数据文件

## 1.4 参考资料

# 2 总体设计

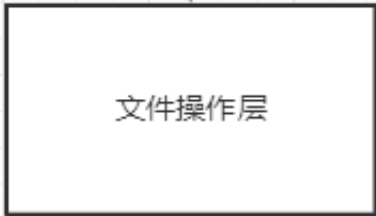
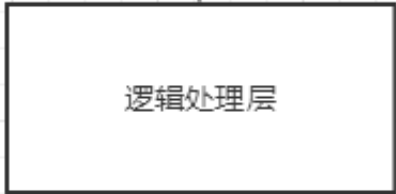
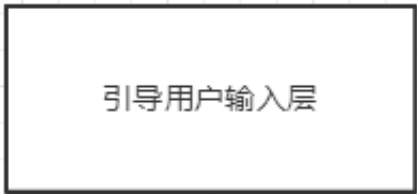
## 2.1 需求规定

1. 增加: 完成对学生信息的增加操作
2. 删除：完成对学生信息的删除操作
3. 修改：可修改指定的学生信息，如果空间长度不够则存储在其他地方，原学生信息视为被删除。
4. 查询：可按学生信息内容、地址或者编号等不同方式查找学生信息的基本信息，要求支持模糊查找，不显示生日而显示年龄。
5. 统计：可统计所有学生各科成绩的最高分、最低分、平均分、总分。
6. 存储信息：显示存储的资源分布，如windows 提供的磁盘碎片整理图，按顺序显示各种已分配、未分配资源。碎片整理

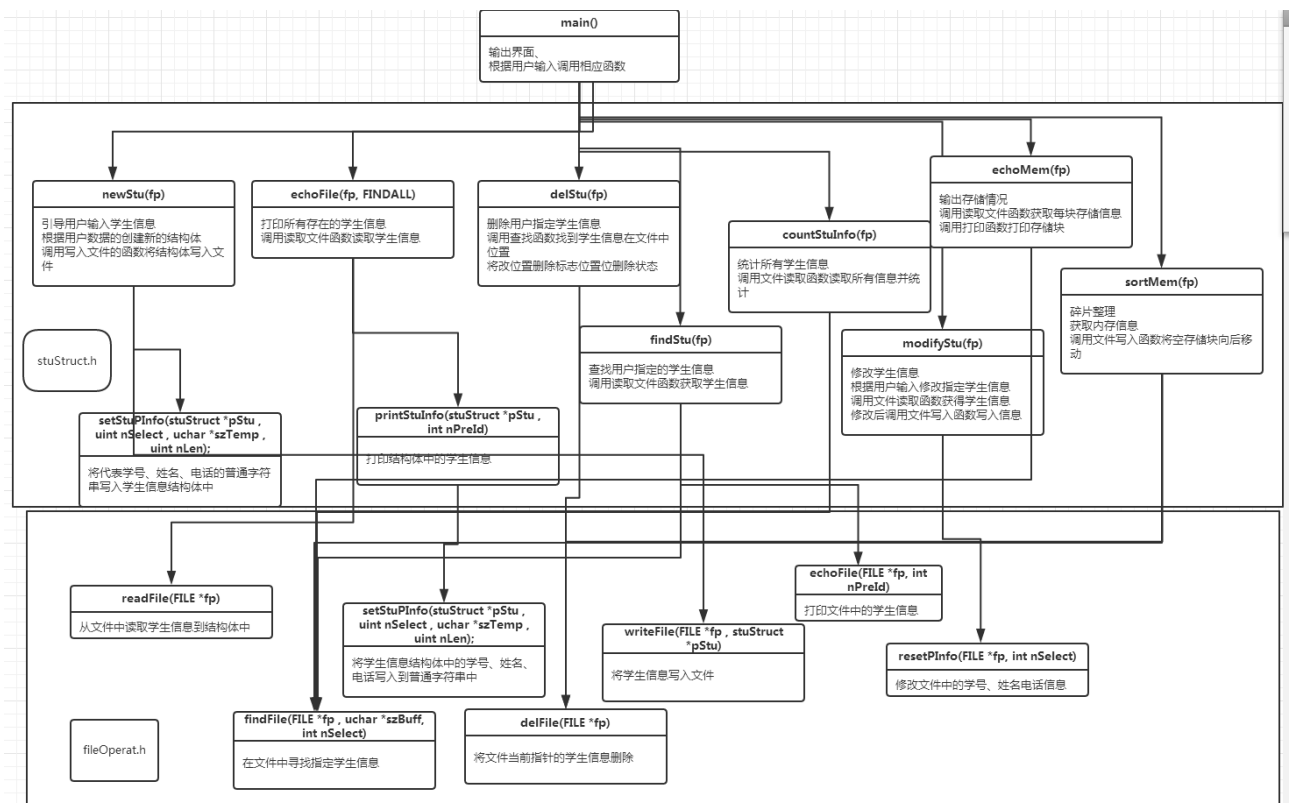
## 2.2 运行环境

支持32位程序运行的windows xp 及以上版本系统

## 2.3 基本设计概念和处理流程



## 2.4 结构



## **2.5 功能需求与程序的关系**

所有功能都在一个程序中完成

## **2.6 人工处理过程**

需要人工输入数据

## **2.7 尚未问决的问题**

无

# **3 接口设计**

## **3.1 用户接口**

```
1.
2.  /*
3.  函数功能：
4.      打印初始选项
5.  参数：
6.      无
7.  返回值：
8.      无
9.  */
10. void printStart();
11.
12. /*
13. 函数功能：
14.     重复打印同一个字符
15. 参数：
16.     cTemp  : 要打印的字符
17.     nNum   : 需要打印的数量
18. 返回值：
19.     无
20. */
21. void printC(char cTemp , int nNum);
22.
23.
24. /*
25. 函数功能：
26.     打开数据库文件
27. 参数：
28.     *fp : 指向数据文件的指针
29. 返回值：
30.     0 正常打开，非0 打开失败
31. */
32. int openStuFile(FILE **fp);
33.
34. /*
35. 函数功能：
36.     关闭重定向文件
37. 参数：
38.     无
39. 返回值：
40.     无
41. */
42. void exitStu();
```

## 3.2 外部接口

```

1.  /*
2.  函数功能：
3.      计算学生信息结构体所存信息在文件中存储需要的大小
4.      并将结构赋值给结构体中的 nStuLen
5.  参数：
6.      *pStu : 指向学生信息结构体的指针
7.      nChLen: 扩展长度
8.  返回值：
9.      无
10. */
11. void countStuLen(stuStruct *stuTemp, ushort nChLen);
12.
13. /*
14. 函数功能：
15.      保存学生信息结构体中原先的学号、姓名、电话信息
16. 参数：
17.      *pStu      : 指向学生信息结构体的指针
18.      *nLenArr   : 保存原先数据长度的一维数组指针
19.      *szSave[]  : 指向需要存放信息的二维数组指针
20. 返回值：
21.      无
22. */
23. void saveStuPInfo(stuStruct *pStu ,
24.                  ushort *nLenArr,
25.                  uchar szSave[][MAXINFOLEN]);
26.
27. /*
28. 函数功能：
29.      通过传入的选项设置学号、姓名、电话
30.      在被调用之前调用函数应分配好足够的结构体空间
31. 参数：
32.      *pStu      : 指向学生信息的结构体
33.      nSelect    : 操作选项 (0、学号 1、姓名 2、电话)
34.      *szTemp    : 设置内容字符串
35.      nLen       : 字符串长度
36. 返回值：
37.      无
38. */
39. void setStuPInfo(stuStruct *pStu ,
40.                 uint nSelect ,
41.                 uchar *szTemp ,
42.                 uint nLen);
43.
44. /*
45. 函数功能：
46.      在堆中创建一个学生信息结构体 并要求输入赋值
47.      最终将结构体信息写入文件
48. 参数：
49.      *fp 数据文件指针
50. 返回值：
51.      指向结构体的指针
52. */
53. void newStu(FILE *fp);

```

```

54.
55.  /*
56.  函数功能：
57.      从文件中读取指针所在位置的数据到结构体中
58.  参数：
59.      *fp : 数据文件指针
60.  返回值：
61.      指向在堆中的学生信息结构体的地址
62.  */
63.  stuStruct *readFile(FILE *fp);
64.
65.  /*
66.  函数功能：
67.      打印学生结构体信息
68.  参数：
69.      *pStu : 指向学生信息结构体的指针
70.      nPreId : 该结构体序号
71.  返回值：
72.      无
73.  */
74.  void printStuInfo(stuStruct *pStu , int nPreId);
75.
76.  /*
77.  函数功能：
78.      通过用户的输入进行查找操作
79.  参数：
80.      *fp : 指向数据文件的指针
81.  返回值：
82.      无
83.  */
84.  void findStu(FILE *fp);
85.
86.  /*
87.  函数功能：
88.      根据用户输入修改学生信息
89.  参数：
90.      *fp : 数据文件指针
91.  返回值：
92.      无
93.  */
94.  void modifyStu(FILE *fp);
95.
96.  /*
97.  函数功能：
98.      重新设置学号、姓名、电话
99.  参数：
100.     *fp : 数据文件指针
101.     nSelect: 修改选项
102.  返回值：
103.     无
104.  */
105.  void resetPInfo(FILE *fp, int nSelect);
106.
107.  /*

```

```
108.  函数功能：
109.      根据用户输入删除指定学生信息
110.  参数：
111.      *fp : 数据文件指针
112.  返回值：
113.      无
114.  */
115.  void delStu(FILE *fp);
```

### 3.3 内部接口



```
1.  /*
2.  函数功能：
3.      将结构体信息写入到文件中
4.  参数：
5.      *fp    : 数据文件指针
6.      *pStu  : 学生结构体指针
7.  返回值：
8.      无
9.  */
10. void writeFile(FILE *fp , stuStruct *pStu);
11.
12. /*
13. 函数功能：
14.      从文件中读取指针所在位置的数据到结构体中
15. 参数：
16.      *fp : 数据文件指针
17. 返回值：
18.      指向在堆中的学生信息结构体的地址
19.  */
20. stuStruct *readFile(FILE *fp);
21.
22. /*
23. 函数功能：
24.      删除文件中文件指针所在位置的学生结构体信息
25. 参数：
26.      *fp : 数据文件指针
27. 返回值：
28.      无
29.  */
30. void delFile(FILE *fp);
31.
32. /*
33. 函数功能：
34.      打印所有未被删除的学生信息
35.      如打印所有则在打印完后将指针指向文件头部
36.      如打印指定则打印完后将指针指向该结构体在文件中的起始位置
37. 参数：
38.      *fp      : 数据文件指针
39.      nPreId   : 如赋值为0则打印所有学生信息
40.                  否则只打印第nPreId个学生信息
41. 返回值：
42.      0 没有查找到；非0 查找到
43.  */
44. int echoFile(FILE *fp, int nPreId);
45.
46. /*
47. 函数功能：
48.      在文件中查找结构体，
49.      若指定只查找匹配到的第一个学生信息，找到后将文件指针置向
50.      查找到的结构体在文件中的起始位置；
51.      否则查找结束后将文件指针指向文件起始位置
52. 参数：
53.      *fp      : 数据文件指针
```

```

54.     *szBuff : 待查找的字符串
55.     nSelect : 查找模式 FINDALL FINDONE
56. 返回值:
57.     0 没有查找到; 非0 查找到
58. */
59. int findFile(FILE *fp , uchar *szBuff, int nSelect);
60.
61. /*
62. 函数功能:
63.     统计所有学生人数, C语言成绩最高分、最低分、平均分、总分
64. 参数:
65.     *fp : 指向数据文件的指针
66. 返回值:
67.     无
68. */
69. void countStuInfo(FILE *fp);
70.
71. /*
72. 函数功能:
73.     打印文件存储情况
74. 参数:
75.     *fp : 数据文件指针
76. 返回值:
77.     无
78. */
79. void echoMem(FILE *fp);
80.
81. /*
82. 函数功能:
83.     打印内存块
84. 参数:
85.     nMem : 打印数量
86.     isEmp : 打印类型 非0 空内存 0 已使用内存
87.     nNum : 计算已打印的内存的数量, 控制输出
88. 返回值:
89.     无
90. */
91. void printMem(ushort nMem , int isEmp, long long *nNum);
92.
93. /*
94. 函数功能:
95.     碎片整理
96. 参数:
97.     *fp : 数据文件指针
98. 返回值:
99.     无
100. */
101. void sortMem(FILE *fp);

```

## 4 系统数据结构设计

### 4.1 逻辑结构设计要点

```

1.  /*
2.  学生信息结构体:
3.      nStuLen    : 学生信息长度
4.      isDel      : 是否被删除
5.      nIdLen     : 学号长度
6.      nNameLen   : 姓名长度
7.      nTelLen    : 电话号码长度
8.      nYear      : 生日年份
9.      nMonth     : 生日月份
10.     nDay       : 生日日期
11.     fScore     : C语言成绩, 精确到小数点后1位
12.     szInfoArr  : 用于存放学号、姓名、电话信息
13.                  最终通过分配空间实现可变长数组
14.     通过上面的顺序结构体默认大小为 20 + ALIGNED 字节
15.  */
16.  typedef struct _stuStruct
17.  {
18.      ushort nStuLen;
19.      ushort isDel;
20.      ushort nIdLen;
21.      ushort nNameLen;
22.      ushort nTelLen;
23.      ushort nYear;
24.      ushort nMonth;
25.      ushort nDay;
26.      float fScore;
27.      uchar szInfoArr[ ALIGNED ];
28.  } stuStruct;

```

## 4.2 物理结构设计要点

Offset	数据长度	删除标志位	5	6	7	数据内容	A	B	C	D	E	F
00000000	20 00	00 00	03 00	02 00	03 00	CB 07	06 00	1B 00				
00000010	33 33	6F 42	31 32	33 C0	EE 31	31 30	CD CD	CD CD				
00000020	2C 00	00 00	05 00	04 00	0C 00	CB 07	03 00	0B 00				
00000030	9A 99	B8 42	35 34	38 36	34 C0	CF BD	AF 30	32 39				
00000040	2D 38	35 31	33 39	38 35	36 30	35 CD	2C 00	00 00				
00000050	06 00	04 00	0C 00	CE 07	04 00	1E 00	E1 FA	B3 42				
00000060	37 34	39 38	32 36	CB BE	BB FA	30 32	39 2B	38 35				
00000070	31 33	39 38	35 36	CD CD	5C 00	FF FF	0A 00	06 00				
00000080	0B 00	CB 07	05 00	1E 00	9A 99	9F 42	31 33	32 30				
00000090	30 34	34 30	35 34	CB BE	D3 FD	B2 A9	31 33	31 31				
000000A0	30 33	38 38	39 32	32 CD	2C 00	FF FF	35 36	CD CD				
000000B0	24 00	FF FF	05 00	04 00	03 00	CB 07	03 00	0B 00				
000000C0	9A 99	B8 42	35 34	38 36	34 C0	CF BD	AF 2B	38 36				
000000D0	CD CD	CD CD										

## 4.3 数据结构与程序的关系

程序的所有数据处理都是按照以上数据结构的定义进行的

## **5 系统出错处理设计**

### **5.1 出错信息**

1. 数据文件打开失败
2. 数据输入有误

### **5.2 补救措施**

数据输入有误则提示重新输入

### **5.3 系统维护设计**

在需要增加学生信息字段时直接在结构体中增加一个变量就行