

温州大学计算机与人工智能学院

程序设计基础(2023-2024-1) 实验报告

实验名称	查找学生信息				
班 级	23大数据1	姓 名	徐王晶	学 号	23211870102
实验地点	南5B105	实验时间	2023-10-21,21:35:26	指导老师	虞铭财

一、问题编号：

1998

地址: <http://10.132.254.54/problem/1998/>

二、问题描述：

请设计一个简单的学生成绩管理系统，要求系统实现以下功能：

插入学生信息： Insert id name sex year month day x y z, 其中的参数分别为学号、姓名、性别、出生日期（年、月、日）、三门课的成绩，成绩为浮点数。

按输入顺序输出学生信息 List

查找学生信息： Find id 查找学号为id的学生信息。

退出程序： Quit或者Exit

要求：（1）编写查找学生信息函数

请注意：姓名的长度不超过20。

三、输入说明：

输入有多行，每行一条指令，指令格式如下：

Insert id name sex year month day x y z 其中的参数分别为学号、姓名、性别、出生日期（年、月、日）三门课的成绩，成绩为浮点数。

List 按输入顺序输出所有学信息。

Find id 查找学号为 id 的学生信息。

Quit或者Exit 输出"Good bye!"后结束程序。

请注意：姓名的长度不超过20。

四、输出说明：

输出有多行，对应命令的输出如下：

Insert id name sex year month day x y z 插入后在单独的一行中输出"Insert:",
如果链表中不存在相同学号的学生信息，在第二行中显示学生信息，
格式: id name sex year month day x y z ave sum, 分别为学号、姓名、性别、出生日期（年、月、日）和三门课（语文、数学、英语）的成绩，平均成绩和总成绩，
数据之间用一个空格分开，成绩保留1位小数。
如果已经有相同学号的学生信息，插入不成功，输出"Failed"

List 输出"List:"后，按输入顺序输出所有学生的信息，格式与插入学生信息后输出的格式相同。

Find id 第一行显示"Find:", 第二行显示格式如下: 如果找到学号为id的学生，则在单独一行中显示学生信息，格式如List。否则在单独一行显示"Failed"。

Quit或者Exit 在单独一行中输出"Good bye!"后结束程序。

五、输入样例：

```
Insert 0911001 zhangsan F 1992 3 24 87 78 65
Insert 0911003 Lisi F 1992 5 3 77 72 55
List
Find 0911002
Find 0911003
Insert 0911001 zhangou M 1992 3 24 98 78 65
Insert 0911002 zhaoliu F 1993 8 8 97 90 55
Insert 0911005 Wangrong F 1990 12 12 68 56 100
Quit
```

六、输出样例：

```
Insert:
0911001 zhangsan F 1992 3 24 87.0 78.0 65.0 76.7 230.0
Insert:
0911003 Lisi F 1992 5 3 77.0 72.0 55.0 68.0 204.0
List:
0911001 zhangsan F 1992 3 24 87.0 78.0 65.0 76.7 230.0
0911003 Lisi F 1992 5 3 77.0 72.0 55.0 68.0 204.0
Find:
Failed
Find:
0911003 Lisi F 1992 5 3 77.0 72.0 55.0 68.0 204.0
Insert:
Failed
Insert:
0911002 zhaoliu F 1993 8 8 97.0 90.0 55.0 80.7 242.0
Insert:
0911005 Wangrong F 1990 12 12 68.0 56.0 100.0 74.7 224.0
Good bye!
```

七、解答内容：

所用语言：

源代码：

```
001. #include <stdio.h>
002. #include <string.h>
003.
004. int main(void)
005. {
006.     char num[100][20];
007.     char name[100][20];
008.     char sex[100][5];
009.     int year[100];
010.     int month[100];
011.     int day[100];
012.     double x[100];
013.     double y[100];
014.     double z[100];
015.     double avg[100];
016.     double sum[100];
017.     int p = 0;
018.
019.     char insert[] = "Insert";
020.     char lst[] = "List";
021.     char quit[] = "Quit";
022.     char exit[] = "Exit";
023.     char find[] = "Find";
024.     char change[] = "Change";
025.     char del[] = "Delete";
026.     char str[100];
027.
028.     scanf("%s", str);
029.     while (strcmp(quit, str) != 0 && strcmp(exit, str) != 0)
030.     {
031.         if (strcmp(insert, str) == 0)
032.         {
033.             printf("Insert:\n");
034.             scanf("%s", num[p]);
035.             int flag = 1;
```

```

036.     for (int i = 0; i < p; i++)
037.     {
038.         if (strcmp(num[i], num[p]) == 0)
039.         {
040.             flag = 0;
041.             break;
042.         }
043.     }
044.     if (flag)
045.     {
046.         scanf(
047.             "%s %s %d %d %d %lf %lf %lf",
048.             name[p],
049.             sex[p],
050.             &year[p],
051.             &month[p],
052.             &day[p],
053.             &x[p],
054.             &y[p],
055.             &z[p]
056.         );
057.
058.         sum[p] = x[p] + y[p] + z[p];
059.         avg[p] = sum[p] / 3.0;
060.
061.         printf(
062.             "%s %s %s %d %d %d %.1lf %.1lf %.1lf %.1lf %.1lf\n",
063.             num[p],
064.             name[p],
065.             sex[p],
066.             year[p],
067.             month[p],
068.             day[p],
069.             x[p],
070.             y[p],
071.             z[p],
072.             avg[p],
073.             sum[p]
074.         );
075.         p++;
076.     }
077.     else
078.     {
079.         printf("Failed\n");
080.         scanf(
081.             "%s %s %d %d %d %lf %lf %lf",
082.             name[p],
083.             sex[p],
084.             &year[p],
085.             &month[p],
086.             &day[p],
087.             &x[p],
088.             &y[p],
089.             &z[p]
090.         );
091.     }
092.
093. }
094. else if (strcmp(lst, str) == 0)
095. {
096.     printf("List:\n");
097.     for (int i = 0; i < p; i++)
098.     {
099.         printf(
100.             "%s %s %s %d %d %d %.1lf %.1lf %.1lf %.1lf %.1lf\n",
101.             num[i],
102.             name[i],
103.             sex[i],
104.             year[i],
105.             month[i],
106.             day[i],
107.             x[i],
108.             y[i],
109.             z[i],
110.             avg[i],
111.             sum[i]
112.         );
113.     }
114. }
115. else if (strcmp(find, str) == 0)
116. {
117.     char id[100];
118.     scanf("%s", id);
119.     printf("Find:\n");
120.     int flag = 0;
121.     for (int i = 0; i < p; i++)
122.     {
123.         if (strcmp(num[i], id) == 0)
124.         {
125.             printf(
126.                 "%s %s %s %d %d %d %.1lf %.1lf %.1lf %.1lf %.1lf\n",

```

```

127.         num[i],
128.         name[i],
129.         sex[i],
130.         year[i],
131.         month[i],
132.         day[i],
133.         x[i],
134.         y[i],
135.         z[i],
136.         avg[i],
137.         sum[i]
138.     );
139.     flag = 1;
140.     break;
141. }
142. }
143. if (flag == 0)
144. {
145.     printf("Failed\n");
146. }
147. }
148. else if (strcmp(change, str) == 0)
149. {
150.     char id[100];
151.     scanf("%s", id);
152.     printf("Change:\n");
153.     int flag = 0;
154.     int j;
155.     for (int i = 0; i < p; i++)
156.     {
157.         if (strcmp(num[i], id) == 0)
158.         {
159.             j = i;
160.             flag = 1;
161.         }
162.     }
163.     if (flag)
164.     {
165.         scanf(
166.             "%s %s %d %d %d %lf %lf %lf",
167.             name[j],
168.             sex[j],
169.             &year[j],
170.             &month[j],
171.             &day[j],
172.             &x[j],
173.             &y[j],
174.             &z[j]
175.         );
176.
177.         sum[j] = x[j] + y[j] + z[j];
178.         avg[j] = sum[j] / 3.0;
179.
180.         printf(
181.             "%s %s %s %d %d %d %.1lf %.1lf %.1lf %.1lf %.1lf\n",
182.             num[j],
183.             name[j],
184.             sex[j],
185.             year[j],
186.             month[j],
187.             day[j],
188.             x[j],
189.             y[j],
190.             z[j],
191.             avg[j],
192.             sum[j]
193.         );
194.     }
195.     else
196.     {
197.         scanf(
198.             "%s %s %d %d %d %lf %lf %lf",
199.             name[p],
200.             sex[p],
201.             &year[p],
202.             &month[p],
203.             &day[p],
204.             &x[p],
205.             &y[p],
206.             &z[p]
207.         );
208.         printf("Failed\n");
209.     }
210. }
211. else if (strcmp(del, str) == 0)
212. {
213.     char id[100];
214.     scanf("%s", id);
215.     printf("Delete:\n");
216.     int flag = 0;
217.     int j;

```

```

218.         for (int i = 0; i < p; i++)
219.         {
220.             if (strcmp(num[i], id) == 0)
221.             {
222.                 j = i;
223.                 flag = 1;
224.             }
225.         }
226.         if (flag)
227.         {
228.             for (int i=j+1; i < p; i++)
229.             {
230.                 strcpy(num[i-1], num[i]);
231.                 strcpy(name[i-1], name[i]);
232.                 strcpy(sex[i-1], sex[i]);
233.                 year[i-1] = year[i];
234.                 month[i-1] = month[i];
235.                 day[i-1] = day[i];
236.                 x[i-1] = x[i];
237.                 y[i-1] = y[i];
238.                 z[i-1] = z[i];
239.                 sum[i-1] = sum[i];
240.                 avg[i-1] = avg[i];
241.             }
242.             p--;
243.             printf("Deleted\n");
244.         }
245.         else
246.         {
247.             printf("Failed\n");
248.         }
249.     }
250.     scanf("%s", str);
251. }
252. printf("Good bye!\n");
253. return 0;
254. }

```

八、判题结果

AC - 答案正确

判题结果补充说明：

test id:3818,result:AC, usedtime:0MS, usedmem:1036KB,score:50 test id:3819,result:AC, usedtime:0MS, usedmem:1036KB,score:50