第11次作业

**（两道题目写一个程序就可以了）**

4. 建立一个学生班的成绩登记表，包括的信息有班号(全班一个)、总人数(设为 10 个 人)、制表日期(整个表一个)和每个学生的信息。每个学生包含下列信息:学号、姓名、 四门课程的成绩。输入每个人的上述所有信息，统计每个学生四门课程的平均成绩，然后按 平均成绩从高到低的顺序输出每个学生的学号、姓名、平均成绩和名次一览表。要求平均成 绩保留两位小数;成绩相等者名次应相同。

5. 在题 4 的基础上，增加下列功能: (1)找出平均成绩的最高分，并输出最高分学生的学号、姓名、平均成绩: (2)统计平均成绩在 60 分以下学生的人数，输出这些学生的学号、姓名和平均成绩。

要求将功能 1 和 2 分别定义成函数。

#include <stdio.h>

#include <string.h>

#include <math.h>

#define N 10

struct student

{

char number[10];//学号

char name[10];//姓名

int grade[4];//四门课成绩

float ave\_grade;//平均成绩

int rank;//排名

};

struct dateInfor

{

//制表日期

int year;

int month;

int day;

};

struct classInfor

{

//制表日期

struct dateInfor data;

int class\_num;//班号

int student\_num;

struct student stu[N];

};

void input( struct student \*stu, int n);//信息输入

void sort(struct student \*stu, int n);//成绩排序

void output(struct classInfor cls);//打印表格

void findmax(struct student \*stu, int n);

void find\_nopass(struct student \*stu, int n) ;

int main()

{

struct classInfor cls;

int i=0;

char str[20];

cls.student\_num = N;

printf("please input class number\n");

scanf("%d",&(cls.class\_num));

getchar();

//最好用下面的

//get(str);

//cls.class\_num = atoi(str);

printf("please input date\n");

scanf("%4d%2d%2d",&(cls.data.year), &(cls.data.month), &(cls.data.day));

getchar();

//输入学生信息

input( cls.stu, N);

//排序

sort(cls.stu, N);

//生成表格

output(cls);

findmax(cls.stu, N);

find\_nopass(cls.stu, N);

return 0;

}

void input( struct student \*stu, int n)//信息输入

{

int i;

char str[20];

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

{

printf("please input student's name\n");

gets(stu->name);

printf("please input student's number\n");

gets(stu->number);

printf("please input student's grade\_one\n");

scanf("%d", stu->grade ); getchar();

//最好用下面的

//get(str);

//stu->grade[0] = atoi(str);

printf("please input student's grade\_two\n");

scanf("%d", stu->grade+1 );getchar();

//最好用下面的

//get(str);

//stu->grade[1] = atoi(str);

printf("please input student's grade\_three\n");

scanf("%d", stu->grade+2); getchar();

//最好用下面的

//get(str);

//stu->grade[2] = atoi(str);

printf("please input student's grade\_four\n");

scanf("%d", stu->grade+3); getchar();

//最好用下面的

//get(str);

//stu->grade[3] = atoi(str);

stu->ave\_grade = ( stu->grade[0] + stu->grade[1] + stu->grade[2] + stu->grade[3] )/4.0;

stu++;

}

}

void sort(struct student \*stu, int n)//成绩排序

{

int i,j,k;

struct student temp;

//以平均成绩从高到低排序

for(i=0;i<n-1;i++)

{

for(j=i+1;j<n;j++)

{

if( stu[i].ave\_grade > stu[j].ave\_grade )

{

temp=stu[i];

stu[i]=stu[j];

stu[j]=temp;

}

}

}

//确定排名

stu[0].rank=1;

k=0;

for(i=1;i<n;i++)

{

stu[i].rank=i+1;

if( fabs(stu[i].ave\_grade - stu[i-1].ave\_grade) < 1e-6 )

{//如果和前一位平均分相同，则排名往前k位

k++;

stu[i].rank=i+1-k;

}

else

{

k=0;

}

}

}

void output( struct classInfor cls)

{

int i=0;

printf("date: %d .%d .%d \n", cls.data.year, cls.data.month, cls.data.day);

printf("Class: %d \n Students: %d \n",cls.class\_num, cls.student\_num);

for(i=0;i<cls.student\_num;i++)

{

printf("%-10s%-10s %d %d %d %d %.2f %d \n",

cls.stu[i].number, cls.stu[i].name,cls.stu[i].grade[0], cls.stu[i].grade[1], cls.stu[i].grade[2], cls.stu[i].grade[3],

cls.stu[i].ave\_grade, cls.stu[i].rank);

}

}

void findmax(struct student \*stu, int n)

{

int i;

printf("Highet students:\n");

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

{

if( stu[i].rank == 1)

{

printf("%-10s%-10s%.2f\n",stu[i].number, stu[i].name, stu[i].ave\_grade);

}

}

}

void find\_nopass(struct student \*stu, int n)

{

int i;

printf("Nopass students:\n");

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

{

if( stu[i].ave\_grade < 60.0 )

{

printf("%-10s%-10s%.2f\n",stu[i].number, stu[i].name, stu[i].ave\_grade);

}

}

}