# 实验报告10

实验十名称：结构体 练习

实验目的：1、理解结构体数据、结构体类型的概念；2、掌握结构体变量、结构体数组、结构体指针变量的使用方法。

填写日期： 2022年12月8日

本次实验成绩：

1. **实验任务：**

1、定义一个表示复数的结构体类型，利用该结构体类型的变量实现复数的除法运算，1）直接使用主函数实现该运算；2）定义结构体变量为形式参数的函数实现该运算。

2、有五个学生，采用结构体变量的方法保存学生数据，每个学生有3门课的成绩，从键盘输入相关数据（包括：学号，姓名和三门课程的成绩），计算输出每位学生的平均成绩。

3、采用结构体数组的方法保存学生数据，完成练习2。

4、定义一个结构体数组，存放12个月的信息，每个数组元素由3个成员构成：月份的数字表示、月份的英文单词及该月的天数。定义一个指向结构体数组的指针变量，使用该指针变量输出一年12个月的相关信息。

1. **实验方法及知识点应用：**

1、结构体类型的概念和定义。

2、结构体变量和结构体数组的使用方法。

3、结构体数据做函数参数的使用方法。

1. **实验程序代码及结果记录：（以表述清楚为目的）**

任务1：

源代码：

#include<stdio.h>

typedef struct complex

{

double re,im;

}cp;

int main()

{

cp a,b,c;

printf("请输入第一个复数：\n");

scanf("%lf%lfi",&a.re,&a.im);

printf("请输入第二个复数：\n");

scanf("%lf%lfi",&b.re,&b.im);

if(b.re==0&&b.im==0)

{

printf("Divided by zero!");

return 0;

}

c.re=(a.re\*b.re+a.im\*b.im)/(b.re\*b.re+b.im\*b.im);

c.im=(a.im\*b.re-a.re\*b.im)/(b.re\*b.re+b.im\*b.im);

if(c.im>0)

{

printf("结果为：\n%lg+%lgi",c.re,c.im);

}

else if(c.im==0&&c.re==0)

{

printf("结果为：\n%lg",c.re);

}

else if(c.im==0)

{

printf("结果为：\n%lg",c.re);

}

else if(c.re==0)

{

printf("结果为：\n%lgi",c.im);

}

else

{

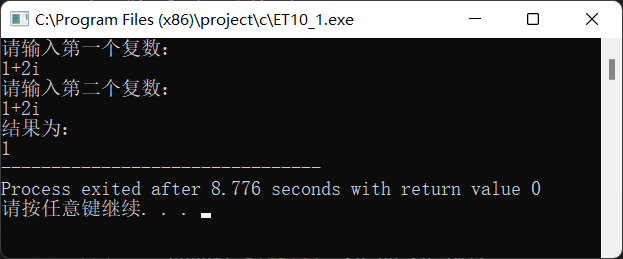
printf("结果为：\n%lg%lgi",c.re,c.im);

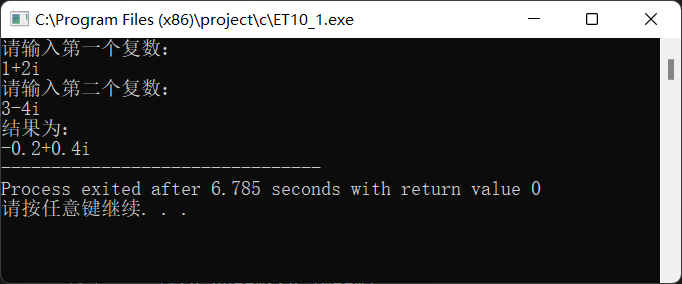
}

return 0;

}

测试：





任务2：

源代码：

#include<stdio.h>

typedef struct student

{

double id;

char name[100];

float score[3];

}STU;

int main(){

int i;

float avg[5];

STU s1;

float sum=0;

printf("请输入五位学生的学号姓名及三门课程成绩：\n");

scanf("%lf",&s1.id);

scanf("%s",s1.name);

for(i=0;i<3;i++)

{

scanf("%f",&s1.score[i]);

sum+=s1.score[i];

}

avg[0]= sum/3;

sum=0;

STU s2;

scanf("%lf",&s2.id);

scanf("%s",s2.name);

for(i=0;i<3;i++)

{

scanf("%f",&s2.score[i]);

sum+=s2.score[i];

}

avg[1]= sum/3;

sum=0;

STU s3;

scanf("%lf",&s3.id);

scanf("%s",s3.name);

for(i=0;i<3;i++)

{

scanf("%f",&s3.score[i]);

sum+=s3.score[i];

}

avg[2]= sum/3;

sum=0;

STU s4;

scanf("%lf",&s4.id);

scanf("%s",s4.name);

for(i=0;i<3;i++)

{

scanf("%f",&s4.score[i]);

sum+=s4.score[i];

}

avg[3]= sum/3;

sum=0;

STU s5;

scanf("%lf",&s5.id);

scanf("%s",s5.name);

for(i=0;i<3;i++)

{

scanf("%f",&s5.score[i]);

sum+=s5.score[i];

}

avg[4]= sum/3;

sum=0;

for(i=0;i<5;i++)

{

printf("第%d个学生平均成绩为：\n",i);

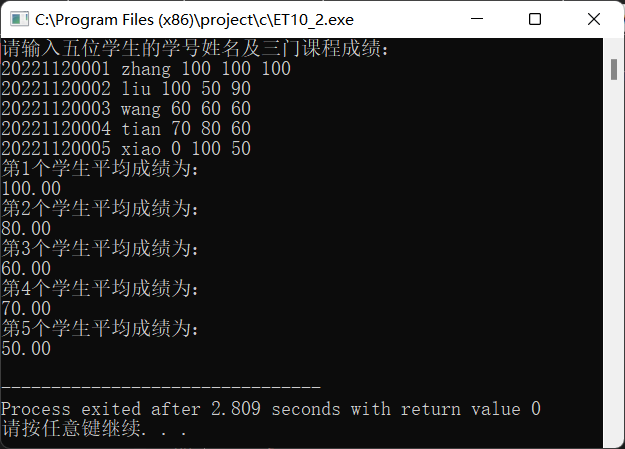
printf("%.2f\n",avg[i]);

}

return 0;

}

测试：



任务3：

源代码：

#include<stdio.h>

struct student

{

double id;

char name[100];

float score[3];

}stu[5];

int main()

{

int i;

float avg[5];

printf("请输入五位学生的学号姓名及三门课程成绩：\n");

for(i=0;i<5;i++)

{

float sum=0;

scanf("%lf",&stu[i].id);

scanf("%s",stu[i].name);

int j;

for(j=0;j<3;j++)

{

scanf("%f",&stu[i].score[j]);

sum+=stu[i].score[j];

}

avg[i]=sum/3;

}

for(i=0;i<5;i++)

{

printf("第%d个学生平均成绩为：\n",i+1);

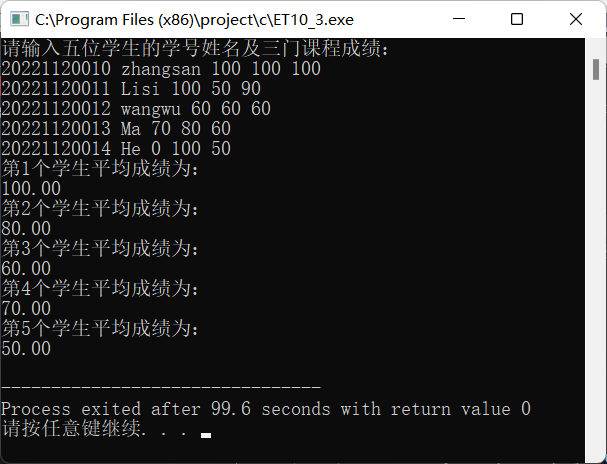
printf("%.2f\n",avg[i]);

}

return 0;

}

测试：



任务4：

源代码：

#include<stdio.h>

struct month

{

int num;

char\* word;

int days;

}mon[12]={{1,"January",31},{2,"February",28},

{3,"March",31},{4,"April",30},{5,"May",31},{6,"June",30},{7,"July",31},

{8,"August",31},{9,"September",30},{10,"October",31},{11,"November",30},{12,"December",31}};

int main()

{

char a;

printf("请输入是否是闰年(Y or N):\n");

a=getchar();

if(a=='Y')

{

mon[1].days=29;

}

else if(a!='N'&&a!='Y')

{

printf("Invalid character!");

return 0;

}

struct month\* p=mon;

int i;

for(i=0;i<12;i++)

{

printf("序号：%d 英文单词：%s 天数：%d\n",p->num,p->word,p->days);

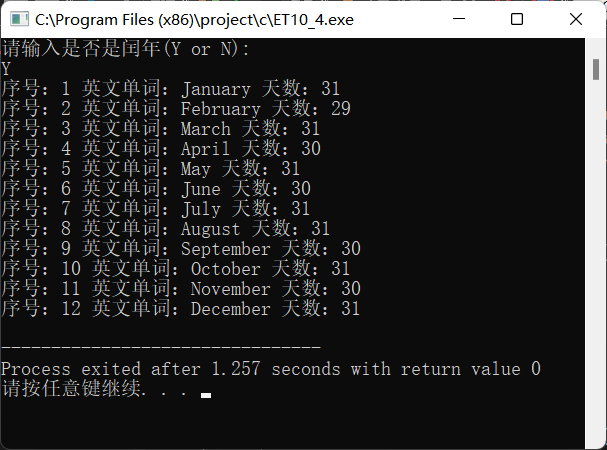
p++;

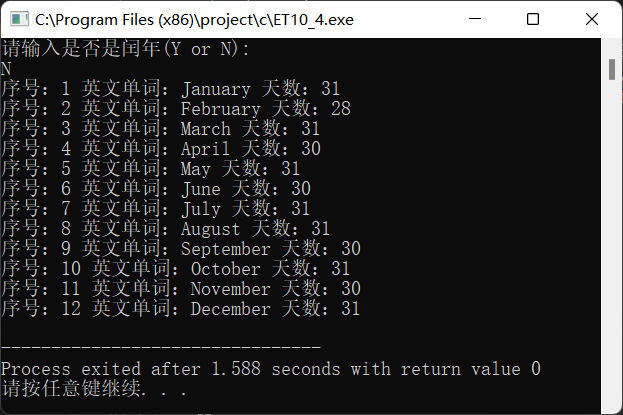
}

return 0;

}

测试：





1. **实验小结：**

先总结自己应用的知识点，再总结自己对知识点的理解和分析，根据实验结果和记录，最后总结实验中常出现的错误和问题。

应用知识点：

结构体数据结构类型的使用，成员变量的定义，结构体数组的定义与赋值，typedef关键字的使用，使用‘.’对成员变量进行调用，结构体指针的定义与使用指针操作结构体，%lg占位符的使用

理解：

1. 结构体是不同类型元素的集合，一个结构体内可以有不同数据类型的成员变量，而数组是相同数据类型元素的集合，不能存放其他类型的数据。
2. 在直接对成员变量或函数进行调用时使用“结构体变量名.成员变量（函数）”的形式进行调用，而用结构体指针进行调用时使用“指针名->成员变量（函数）”的方式进行调用。
3. typedef关键字用于对数据类型起别名，如typedef struct student{}STU;相当于将struct student别名为STU，在主函数中创建变量时可直接使用STU来进行创建，使代码更简洁。
4. 在结构体数组声明时可以使用类似{{val1,val2},{val1,val2}}的方式为其进行赋值。
5. %g/%lg占位符可以自动判断小数点后的位数，自动省略多余的小数位数上的0。

注意：

1. 结构体声明时注意{}后要有‘；’
2. 结构体中的字符数组成员变量在输入时仍不需要取址符&