

C 习题与解析(友元-04) PDF转换可能丢失图片或格式，建议
阅读原文

https://www.100test.com/kao_ti2020/133/2021_2022_C___E4_B9_A0_E9_A2_98_E4_c97_133906.htm 用友元函数的方法重新设计

“引用”中的类Point，并求两个点之间的距离。解：将原来求两个点的距离的普通函数distance()改写为友元函数即可，可以看到采用友元函数方法使得代码更简洁。本题程序如下

```
#include #include class Point { int x,y. public: Point(int i,int j){x=i.y=j.} friend float distance(Point amp.p2). void disp() { cout }. float distance(Point amp.p2) // 友元函数的实现 { float d. d=sqrt((p1.x-p2.x)*(p1.x-p2.x) (p1.y-p2.y)*(p1.y-p2.y)). return d. } void main() { Point p1(2,2),p2(5,5). p1.disp(). coutcout} 本程序执行结果如下：(2,2)与(5,5)之间距离=4.24264
```

----- 题7.设计一个日期类Date，包括日期的年份、月份和日号，编写一个友元函数，求两个日期之间相差的天数。解：该类中设计有3个友元函数；count_day()函数，它有两个参数，第2个参数是一个标志，当其值等于1时，计算一年的开始到某日期的天数；否则计算某日期到年尾的天数。leap()函数用于判断指定的年份是否为闰年。subs()函数用于计算两个日期之间的天数。本题程序如下：

```
#include #include class Date { int year. int month. int day. public: Date(int y,int m,int d) { year=y.month=m.day=d. } void disp() { printf("%d.%d.%d",year,month,day). } friend int count_day(Date amp.d1,Date amp.d,int flag) { static int day_tab[2][12]={ {31,28,31,30,31,30,31,31,30,31,30,31}, {31,29,31,30,31,30,31,31,30,31,30,31} }. // 使用二维数组存放各月
```

天数，第一行对应非闰年，第二行对应闰年

```

int p,i,s;
if(leap(d.year)) p=1. else p=0. if(flag) { s=d.day. for(i=1.is
=day_tab[p][i-1]. } else { s=day_tab[p][d.month]-d.day.
for(i=d.month 1. is =day_tab[p][i-1]. } return s. } int leap(int year) {
if(year%4==0&&year%100!=0||year%400==0) // 是闰年 return 1. else //
不是闰年 return 0. } int subs(Date amp.d2) { int days,day1,day2,y.
if(d1.year{ days=count_day(d1,0). for(y=d1.year 1. yif(leap(y))
days =366L. else days =365L. days =count_day(d2,1). } else
if(d1.year==d2.year) { day1=count_day(d1,1).
day2=count_day(d2,1). days=day2-day1. } else days=-1. return
days. } void main() { Date d1(2000,1,1),d2(2002,10,1). int
ds=subs(d1,d2). printf("输出结果:\n "). if(ds>=0) { d1.disp().
printf("与"). d2.disp(). printf("之间有%d天\n\n",ds). } else printf("
时间错误!\n"). } 本程序的执行结果如下： 输出结果： 2000.1.1
与2002.10.1之间有1002天

```

----- 题8.编写一个程序，设计一个Point类，包括学号、姓名和成绩等私有数据成员，不含任何成员函数，只将main()设置为该类的友元函数。解：main()函数与其它的函数一样可以设置为类的友元函数，这样就可以在其中使用类对象的私有数据成员。本题的程序如下：

```

#include class Person { int no. char name[10]. int
deg. public: friend void main(). }. void main() { Person obj.
coutcin>>obj.no. coutcin>>obj.name. coutcin>>obj.deg.
coutcout} 本程序执行结果如下： 输入学号： 10 姓名：
Zhengming 成绩： 88 输出结果 学生Zhengming(学号10)成绩
为88

```

----- 题9.采

用友元类的方式重新编写“友元第04题”的程序。解：将原student类中的disp()成员函数和trans()友元函数作为友元类process的成员函数。其执行结果与第4题的结果完全相同。本题程序如下：

```
#include #include #include class student { char name[10]. int deg. char level[7]. friend class process. // 说明友元类 public: student(char na[],int d) { strcpy(name,na). deg=d. } }. class process { public: void trans(student &s) { if(s.deg>=90) strcpy(s.level,"优"). else if(s.deg>=80) strcpy(s.level,"良"). else if(s.deg>=70) strcpy(s.level,"中"). else if(s.deg>=60) 100Test 下载 频道开通，各类考试题目直接下载。详细请访问 www.100test.com
```