**实验一 绪论**

1. 面向对象思想及其三大特点是什么？
2. 对比面向过程和面向对象程序设计思想的优缺点。

实验二 **面向对象程序设计基础（2，3，4，5题中任选3题）**

**1.**熟悉VC环境，练习自己的第一个程序使用Visual C++ 6.0集成环境来编辑，运行简单的数据输入和运算实验

 （1）输入下面的程序，然后运行并分析结果。

 #include<iostream.h>

void main()

{

int x=6, y；

cout<<" x= "<<x<<endl；

 y = ++x

 cout<<"x= "<<x<<"

y ="<<y<<endl

 x=6; y=x++;

 cout<<"x= "<<x<<"

 y ="<<y<<endl

 x=6;y = x--;

 cout<<"x="<<x<< "

 y = "<<y<<endl

 x=6;y = --x;

 cout<<"x="<<x<< "

 y = "<<y<<endl

}

(2) 输入下面的程序，然后运行并分析结果。

#include<iostream.h>

void main()

{

int a=20, b=6;

a+=a; cout<<a<<endl;

a=20;

a-=a;

cout<<a<<endl;

a=20;

a\*=a;

cout<<a<<endl;

a=20;

a/=b;

cout<<a<<endl;

a=20;

a%=b;

cout<<a<<endl;

a=20;

a=a-b;

cout<<a<<endl;

a=10;

b=4;

a+=a\*=a-=a/b; cout<<a<<endl;

}

a=10,b=4

a+=a\*=a-=a/b=?

（3）编写一个程序，要求：提示输入3个数；显示这3个数，求他们的平均值并输出。

**2.** 用 C++ 语言定义一个三维空间的坐标点 Tpoint；并描述三维空间的球 TBall ，实现其主要操作（如计算体积和表面积，输出空间坐标等）。

要求：代码分开编写在不同的文件中，以便掌握多文件联编。

**3.** 身高预测：每个做父母的都关心自己孩子成人后的身高，据有关生理卫生知识与数理统计分析表明，影响小孩成人后的身高的因素包括遗传、饮食习惯与体育锻炼等。小孩成人后的身高与其父母的身高和自身的性别密切相关。

设faHeight为其父身高，moHeight为其母身高，身高预测公式为

男性成人时身高=(faHeight + moHeight)×0.54cm

女性成人时身高=(faHeight×0.923 + moHeight)/2cm

此外，如果喜爱体育锻炼，那么可增加身高2%；如果有良好的卫生饮食习惯，那么可增加身高1.5%。编程从键盘输入用户的性别（用字符型变量sex存储，输入字符F表示女性，输入字符M表示男性）、父母身高（用实型变量存储，faHeight为其父身高，moHeight为其母身高）、是否喜爱体育锻炼（用字符型变量sports存储，输入字符Y表示喜爱，输入字符N表示不喜爱）、是否有良好的饮食习惯等条件（用字符型变量diet存储，输入字符Y表示良好，输入字符N表示不好），利用给定公式和身高预测方法对身高进行预测。

编程要求：有用户输入输出提示信息。

**4.** 学生成绩统计

从键盘输入一个班（全班最多不超过30人）学生某门课的成绩，当输入成绩为负值时，输入结束，分别实现下列功能：

（1）统计不及格人数并打印不及格学生名单；

（2）统计成绩在全班平均分及平均分之上的学生人数，并打印这些学生的名单；

（3）统计各分数段的学生人数及所占的百分比。

注：将成绩分为六个分数段，60分以下为第0段，60~69为第1段，70~79为第2段，80~89为第3段，90~99为第4段，100分为第5段。

编程要求：

(1).较好的用户输入输出提示信息

(2).使用子函数来实现上述各个功能

(3).最好不要使用全局变量

**5.**字符串合并：

输入两个已经按从小到大顺序排列好的字符串，编写一个合并两个字符串的函数，使合并后的字符串，仍然是从小到大排列。

如：输入1223aabcc 和233abbdkm

输出：1222333aaabbbccdkm

编程要求：

较好的用户输入输出提示

先用普通数组编程方式编写，测试好后改用指针方式逐个访问字符数组元素

实验三 类的基础部分(5题中任选1题)

1.设计两个酒店管理员客房管理的类：一个是Person类，要求有储存房号，客户姓名和身份证号的信息；另一个类是Client类，要求新增客户的订房、退房和消费金额等信息。并给出相关测试算法。程序需要满足：

1）具有订房功能：增添客户的信息到相应的房号。包括姓名、身份证号码、开房时间，预计订房的时间；

2）具有退房功能

结算客户的消费时间，以及其他的消费项目，如订餐服务，结算客户的消费金额；并删除相应房号内的客户信息。

输入有效性验证，假设酒店客房有一百间。

2．通过我们所熟悉手机社交软件——微信的朋友圈，其中朋友的信息都存储到相应的结构体中：

struct friends

{

char name[20];//存储朋友姓名

char sex;//f or m

int age;

char telephone[21];

char email[20];

char birthday[20];

int counte;//邮件的记录

char tmessage[500];//存储短信内容

int countp;//电话记录

char emessage[500];

};

要求：1）请设计一个Friend类，用来读信息到结构体数组中；

2）并设计一个Master类，能够满足以下四个功能：

的朋友圈信息管理。需要满足：

a.添加功能

作为主人登录时，添加一个朋友的基本信息，包括姓名，性别，年龄，电话，邮箱，生日等，并选择要添加的分组。

b.删除功能

作为主人登录时，按分组和姓名对一个朋友的信息进行删除。

c.修改功能

作为主人登录时，按分组和姓名对一个朋友的信息进行修改。

d.显示功能

作为主人登录时，按分组有条理显示某一组内所有朋友的基本信息，包括姓名，性别，年龄，电话，邮箱，生日等。

3.为了方便图书管理统计，有reader.txt文件用于存放学生信息（包括学号、姓名和所借阅图书编号），readers.txt文件用于存放教师信息（包括教师工号、姓名和所借阅图书编号），book.txt文件用于存放图书库信息（包括图书名、图书编号和图书数量）。请编写两个类：

1） Student类，主要用于存储学生学生个人信息；

2） Teacher类，其用于存储教师个人信息；

并完成对学生和老师借书情况，借书还书操作，以及显示个人已经借阅图书情况。

4.已知一个航空管理员操作类的数据成员如下：

Class FlightInfo

{

stringNo;//航班号

stringStartCity;//起飞地点

stringEndCity;//降落地点

stringStartDate;//日期

stringStartTime;//起飞时间

stringEndTime;//降落时间

intFirstTicket;//头等舱剩余票数

intFirstPrice;//头等舱价格

intSecondTicket;//经济舱剩余票数

intSecondPrice;//经济价格

public:

…

};

1. 请完成这个类的定义，要求:

a.具有两个参数的构造函数，参数分别为航班号和日期实现航空管理员可以对航班进行新建、修改、删除等操作；

b.设置航班起飞的日期和时间、头等舱/经济舱的价格和剩余票数、出发地和目的地和航班号；

2) 在此基础上，定义一个FlightManager类用来管理航班信息，用来增加、编辑、删除航班。

5.假设有一书店的前台收银销售类如下定义：

Class Sale : //前台销售类

{

private:

int buycount; //购买书的数量

double total; //总价

BookData book[1000],buy[100]; //book数组用于存储图书信息，buy数组用于存储客户购买的图书的信息

public:

Sale(); //构造函数

void addBook(); //向购物车中增加书籍

void getBuy(); //查看购物车

void getCash(); //结算

void getData(); //打开书库文件并将文件中的图书信息读取到内存中

void putData(); //在实验结束前，将图书信息写入书库文件中

};

请完成该类定义，实验并且要满足：

1）向购物车中添加书籍：用户先输入书名查找要购买的书籍，系统会查找相关书籍并询问购买数量，用户输入数量后即可添加至购物车

2）查看购物车:系统显示用户购物车中的书籍以及价格、数量等信息

3）结算：系统计算出用户需要支付的价格

实验四 类和对象

**实验目的**

1. 掌握类的定义，根据具体要求设计类。能够根据类来创建对象。
2. 掌握拷贝构造函数的实现方法。
3. 掌握静态数据成员和静态函数的使用方法。
4. 理解友元的概念和掌握友元的使用方法。
5. 掌握对象数组、对象指针的使用方法。

**实验内容**

1.

* 定义一个student类
* 包含成员变量姓名，年龄，学号
* 包含成员函数在屏幕上显示学生的姓名，年龄，学号
* 包含构造函数，对成员变量初始化
* 创建此类的对象，测试构造函数的调用
* 验证成员函数功能

2.

用多文件组织类，完成上述工作。

3.

* 在student类中重载构造函数，用多种方式对该类的对象初始化
* 测试构造函数的调用顺序

4.

* 向student类添加：
* 构造函数：实现动态内存分配
* 析构函数：释放已分配的内存
* 测试构造函数和析构函数的调用顺序

5.

* 定义一个课程类
* 包含课程名和该课程的学分
* 编写课程类的构造函数
* 编写课程类的拷贝构造函数
* 定义课程对象，测试构造函数和拷贝构造函数

6. （选作）设计一个用来表示直角坐标系的Location类，在主程序中创建类Location的两个对象A和B，要求A的坐标点在第3象限，B的坐标点在第2象限，分别采用成员函数和友元函数计算给定两个坐标点之间的距离，要求按如下格式输出结果：

A(x1,y1),B(x2,y2)

Distance1=d1

Distance2=d2

其中：x1、y1、x2、y2为指定的坐标值，d1和d2为两个坐标点之间的距离。

【提示】

类Location的参考框架如下：

class Location{

public:

Location(double,double); //构造函数

double Getx(); //成员函数，取x坐标的值

double Gety(); //成员函数，取y坐标的值

double distance(Location&); //成员函数，求给定两点之间的距离

friend double distance(Location &,Location &); //友元函数，求给定两点之间的距离

private:

double x,y;

};

7. 声明一个Student类，在该类中包括一个数据成员score(分数)、两个静态数据成员total\_score(总分)和count(学生人数)；还包括一个成员函数account()用于设置分数、累计学生的成绩之和、累计学生人数，一个静态成员函数sum()用于返回学生的成绩之和，另一个静态成员函数average()用于求全班成绩的平均值。在main函数中，输入某班同学的成绩，并调用上述函数求出全班学生的成绩之和和平均分。

8. （选作）建立一个学生类（包括学号、成绩），然后建立一个对象数组，内放5个学生的数据，用指针指向数组首元素，输出第1,3,5个学生的数据。

实验五 输入输出流和文件操作（1，2，3，4题目中任选2题，第5题必做）

**实验目的**

1. 理解C++的输入/输出流的概念。
2. 熟悉文件流的工作过程。
3. 掌握磁盘文件（文本文件、二进制文件)的输入输出方法。

**实验内容**

1. 编写一个程序，要求定义in为fstream的对象，与输入文件file1.txt建立关联，文件file1.txt的内容如下：

abcdef

ghijklmn

定义out为fstream的对象，与输出文件file2.txt建立关联。当文件打开成功后将file1.txt 文件的内容转换成大写字母，输出到file2.txt文件中。

1. 编写一个程序，要求定义in为fstream的对象，与输入文件file1.txt建立关联，文件file1.txt的内容如下：

aabbcc

定义out为fstream的对象，与输出文件file2.txt建立关联。当文件打开成功后将file1.txt 文件的内容附加到file2.txt文件的尾部，运行前file2.txt文件的内容如下：

ABCDEF

GHIJKLMN

运行后，再查看文件file2.txt的内容。

1. 写一个程序，用于统计某文本文件中单词is的个数。
2. 编写一个程序，将两个文本文件连接成一个文件，然后将此文件中所有小写字母转换成大写字母，并打印出来。
3. 编写一个程序，实现以下功能：
4. 输入一系列的学生数据（包括学生的学号、姓名和成绩等基本信息），将学生信息写入二进制文件student.dat中。
5. 从student.dat文件中读出这些数据并显示出来。
6. 在student.dat文件中按姓名进行查询，如输入“李”，则将所有姓“李”的学生的 数据显示出来。
7. 可对指定学生的数据进行修改。
8. 可以删除指定的学生数据。

实验六 类的高级部分（运算符重载，1，2，3，4题中任选2题）

练习1

已知类Complex定义如下：

class Complex{

double real, imag;

public **:**

// 其它成员函数，根据需要填写, 如构造函数、拷贝构造函数

};

**任务**：完成Complex类的+和- 运算符重载（两个Complex对象的real和imag分别相加和相减）

练习2

已知类Point定义如下：

class Point{

float x, y;

public **:**

// 其它成员函数，根据需要填写, 如构造函数、拷贝构造函数

};

**任务**： 完成Point类的①== 、②!=、③>、④<< 运算符重载， 给定两个Point对象p1和p2,

①p1 == p2，即p1.x == p2.x 并且p1.y == p2.y;

②p1 != p2，即 p1.x != p2.x 或者p1.y != p2.y

③p1 > p2，即 p1.x > p2.x或者 p1.x == p2.x 并且p1.y > p2.y

④可以用cout<<输出一个Point对象

练习3

已知类Date定义如下：

class Date{

int d, m, y; //分别表示天、月、年

public **:**

// 其它成员函数，根据需要填写,如构造函数、拷贝构造函数

};

**任务**：①完成Date类的++运算符重载，实现对天数的加1。注意月末加1需要修改天和月，年末加1需要修改天、月和年。②实现Date类的++前缀和后缀，在主函数中分别调用。

练习4

已知类Student定义如下：

class Student{

int id;

char name[16];

public **:**

// 其它成员函数，根据需要填写, 如构造函数、拷贝构造函数

};

**任务：**完成Student类 = 运算符重载

练习5

已知类Student定义如下：

class Student{

int id;

char\* name;

public **:**

// 其它成员函数，根据需要填写，如构造函数、拷贝构造函数

};

在Student类的构造函数中采用new动态分配内存。实现main函数，并创建2个Student对象。

练习6

给出程序输出结果然后用计算机运行监测自己答案的正确性并分析出错原因。

class counter{

private:

unsigned value;

public:

counter() { value = 0;}

counter(unsigned int a) {value = a;}

counter & operator++();

void display();

};

counter & counter::operator++() {

value++ ;

return \* this;

}

void counter :: display() {

cout<<"Total is "<<value<<endl;

}

int main(){

counter i(0), n(10);

i = ++n;

i.display();

n.display();

i=n++;

i.display();

n.display();

return 0;

}

练习7

给出程序输出结果然后用计算机运行监测自己答案的正确性并分析出错原因。

class Num{

int a;

public :

Num(int i):a(i){}

void Inc(int& in){ in + =a;}

Num& operator++()

{

a++;

return \*this;

}

int GetVal(){return a;}

} ;

void main( ) {

Num n(2);

int b = 5;

n.Inc(b);

cout<<b<<endl;

++n;

n.Inc(b);

cout<<b<<endl;

cout<<n.GetVal()<<endl;

}

实验七 继承

1． Date类如下

class Date

{

private:

int year,month,day;

public:

Date(int year=1990,int month=1,int day=1);

Date(Date &r);

void Print();

};

Date::Date(int year,int month,int day)

{

this->year=year;

this->month=month;

this->day=day;

}

Date::Date(Date &r)

{

year=r.year;

month=r.month;

day=r.day;

}

void Date::Print()

{

cout<<year<<'-'<<month<<'-'<<day<<endl;

}

在上面的工程中添加Person类，包括姓名name（字符数组，包括结束符最长20字符），性别gender（字符），生日birthday（Date类对象），人数统计变量count（静态），自行设计成员函数，保证满足以下测试要求(在main函数中调用Test1函数)：

void Test1()

{

Person p1,p2("zhang",'f',Date(1991,1,1)),p3(p2);

p3.Print();//输出: “姓名:zhang;性别: 女;生日:1991-1-1;

cout<<”收支为：”<<p3.GetCashFlow()<<”元”<<endl; //输出：0元

Person \*p=new Person("li",'m',Date(1990,2,2));//创建堆对象（调用有参构造）

p>Print();//输出堆对象基本信息

delete p;//释放堆对象

}

2． 从Person类以public方式派生Student类，添加学号stuId，就读时间（年）stuDuration每年学费tuition(固定为5000元)，自行设计成员函数，要求满足以下测试要求：

void Test2()

{

Student s1,s2("zhang",'m',Date(1992,1,1),1001);

s2.Print(); //输出如下:

//姓名:zhang; 性别: 男; 生日:1992-1-1;

//学号:1001; 就读时间:0年

cout<<endl;

s2.SetDuration(2);

s2.Print();

//姓名:zhang; 性别: 男; 生日:1992-1-1;

//学号:1001; 就读时间:2年

cout<<endl<<"学费支出"<<s2.GetCashFlow()<<"元"<<endl;

//学费支出：10000元

Student \*p=new Student; //堆对象创建（调用无参构造）

delete p;//释放堆对象

}

. main函数如下：

void main(){

Test1();//测试Person类

Test2();//测试Student类

}

实验八 多态（虚函数）

3．多态

(1)已知类Number定义如下：（选作）

class Number {

int value;

public **:**

Number( int i=0 ){ value=i; }

// 其它成员函数，根据需要填写

};

为此类添加如下2个重载操作符函数，并在主函数中说明如何使用这2个重载操作符。

1）+= 完成2个Number类对象相加；

2）++和-- 完成Number类对象的自加和自减（注意：++和—有前置和后置两种）

（2）已知矩形类定义如下：（必做）

class Rect {

protected:

float width , length, area;

public:

Rect( ) { width = length = area = 0.0f ; }

Rect ( float w, float l ) {

width = w; length = l;

area = width \* length;

}

virtual float getArea( ) { return area; }

float getLen( ) { return length; }

float getWidth( ){ return width; }

};要求:1）在Rect类基础之上公有派生出长方体类**Cube**，新增一个私有数据成员：高度height，并设计函数成员中getArea函数，覆盖Rect中的同名虚函数，用于计算长方体的**表**面积。

2）设计main函数，对以上的两个类进行测试，其中cube类中getArea函数要求使用基类Rect的指针通过赋值兼容规则进行调用。