

复 习

题型及分值：(仅供参考，很可能会有调整)

选择	约 20 分
程序综合(找错、写运行结果、填空)	约 40 分
程序设计	约 40 分

课程内容：

面向对象程序设计：类和对象，继承和派生，虚函数和运算符重载
类的静态成员，友元，`const`
C++程序设计：函数、引用、指针、数组、字符串
泛型程序设计：模板、STL

复习提示：

OOP 是核心，课件 PPT 是基础，例程、作业很重要

(一)课程主要内容回顾：

绪论

面向对象程序设计
面向过程程序设计
面向对象程序设计的发展史
C++发展史
C++与 C 的关系
面向对象程序设计的特征
类、对象、封装、继承、多态
面向对象程序设计的优点
模块的独立性高（类的封装）
代码的可重用性高（继承和派生、虚函数、模板、STL）

C++程序设计

函数：
传参的 3 种方式：传值、传引用、传指针(地址)
函数形参的默认值
函数重载
内联函数 `inline`
数组
指针
动态内存分配 `new`
动态内存回收 `delete`
引用
字符串
标准字符数组型字符串 `char name[20];`

string 型字符串 string name;

类和对象

模块的独立性

类定义

private/protected/public

作用域限定运算符::

成员数据、成员函数

this 指针

类使用

对象定义

前向引用声明

作用（类之间的相互使用）

如何定义前向引用类

构造、拷贝构造和析构

构造函数的格式、用法、功能（无返回值、形参、默认值、默认构造函数）

拷贝构造函数的格式、用法、功能

析构函数的格式、用法、功能

默认构造函数、默认拷贝构造函数、默认析构函数

构造函数的调用时机

拷贝构造函数的调用时机

析构函数的调用时机

何时一定要重载构造函数？

浅拷贝与深拷贝

类的组合

类的组合的含义

组合类的构造函数

如何访问子对象、修改子对象

作用域、生命期和可见性

作用域、生命期和可见性

变量的存储类型：动态临时变量、static、register、extern

函数中的 static 函数中的数据共享

静态成员 static

类中的数据共享

功能、格式、初始化

静态成员数据、静态成员函数

静态成员的使用

静态函数中对普通对象的访问

友元 friend

类和类外的数据共享

友元函数的功能、格式

友元类的功能、格式

友元关系的特征

const、常成员、常对象

共享数据的保护

函数传参：常引用 `const` 类型&
常成员数据的作用、如何定义、初始化
常成员函数的作用、如何定义；常成员函数可以用于函数重载
常对象如何定义、初始化？
常对象只能使用常函数

继承和派生

代码重用
功能、格式、派生类的内存结构图
赋值兼容（公有派生对象给基类对象、引用、指针赋值或初始化）
派生类的构造函数、含内嵌对象时的构造函数
派生类的析构函数
派生类构造函数和基类构造函数的关系
派生类析构函数和基类析构函数的关系
接口的重载和重写
三种继承方式：`private/protected/public` 继承
三种继承方式下，基类成员在派生类中访问控制权限的变化
虚基类（含义、格式、作用）

多态性

多态性的含义、功能，分类（重载多态、强制多态、参数多态、包含多态）
绑定（联编）、静态联编、动态联编
虚函数是动态联编的
其他的多态（重载多态、强制多态、参数多态）是静态联编的

虚函数

接口重用
虚函数功能、格式
使用虚函数时，指针、引用的用法
没有虚构造函数
虚函数家族中的析构函数是虚析构函数
纯虚函数、抽象类

运算符重载

运算符重载的功能、格式、编译器的理解方式
运算符重载的规则（哪些能重载/哪些不能，语义的要求，优先级、结合性）
重载成成员函数（哪些符号只能重载成成员函数）
重载成非成员函数（友元）（哪些符号只能重载成非成员函数）
形参何时为引用，返回值何时为引用？（形参或返回值是否引用有什么不同？）
单目运算符的重载、双目运算符的重载
`+, -, *, /, %`, 前`++`, 后`++`, 前`--`, 后`--`, `<<`, `>>`, `[]`, `=`, `>`, `<`, `==`, `+=`, `-=`, `.....`, 类型转换运算符, `.....` 的重载
何时要一定重载拷贝构造函数、`=`？ 如何重载？
深拷贝和浅拷贝

模板

`template`
泛型程序设计
模板的含义、分类、用法
函数模板

模板的实现过程（模板是否能转换成二进制码？为什么？）、模板实例化、模板函数
模板形参表
模板特化
函数模板与函数重载
类模板

STL

泛型程序设计的一个范例

Standard Template Library 标准模板库

容器、迭代器、算法、函数对象

容器分为顺序容器、关联容器、容器适配器

常用的容器：

顺序容器：向量 `vector`、链表 `list`、双端队列 `deque`

关联容器：集合 `set`、多重集合 `multiset`、映射 `map`、多重映射 `multimap`

容器适配器：堆栈 `stack`、队列 `queue`、优先队列 `priority_queue`

迭代器：

`iterator`

面向对象版的指针

常用的算法：

`find`、`count`、`search`、`merge`、`sort`

（二）重点程序：

在程序的复习中，重点是 OOP 程序设计方法的使用：

类和对象

work1 日期 `Date` 类

work2 运动员 `Sporter` 类 构造函数

学校 `School` 类 重写拷贝构造函数实现深拷贝

work3 类的组合 三角形 `Triangle`-点 `Point` 类

静态成员

例：具有静态成员数据 `_count` 的 `Point` 类

静态成员的声明和实现、静态成员函数的使用

work4 养鱼 `Fish` 类 静态成员 鱼的总数

友元

友元函数和友元类

work5 电量统计 `FamElec` 类 友元函数 `TotalElec`、`AveElec`、`Cost`

work6 教师-学生类

教师类是学生的友元类

教师修改学生的成绩、计算学生排名、奖学金等级

继承和派生

work7 世界时钟 `WorldClock`-时钟 `Clock` 类

work8 `Person`、`Student`、`Employee`、`StuEmployee` 类 虚基类设计

虚函数

work9 几何形状类族：形状 `Shape`、点 `Point`、圆 `Circle`、矩形 `Rectangle`、三角形 `Triangle` 类

抽象类 Shape、

(纯) 虚函数 void Show()、double Area()、double Perimeter()

运算符重载

例程：运算符重载 fraction 类 (含有 fraction、STRING、intArray 类)

特别重载了以下符号：取负、>、long()、+=、=、[]

重载=、重写拷贝构造函数解决浅拷贝问题

work10 人民币类 重载+、-、前++、后++、前--、后--、输出符<<、输入符>>

模板

ppt 中的例程

work11 冒泡排序函数模板

STL

ppt 中的例程

work12 约瑟夫问题

(三) 程序格式和规范:

编译预处理命令:

#include

#ifndef

#define

#endif

多文件组织 .h .cpp

头文件:

iostream

fstream

iomanip

cmath

string

vector

list

stack

set

map

algorithm

使用标准命名空间:

using namespace std;