

C++程序设计实验考试大纲

一. 考核目标

该考试通过上机实验考试的方式，评价考生上机编程的能力。考生必须在规定的时间内根据要求完成程序的编写，并将程序调试运行得出结果。

二. 考试系统运行环境

- 硬件环境：IBM-PC 机及其兼容机
- 软件环境：
操作系统：Win9X，Win2000，WinNT 或 WinXP
C++编译器：
■ Microsoft Visual C++ 6.0

三. 考核方式

考试采用上机答题的形式。

共两道题目，满分 100 分。其中第一道题目 40 分，第二道题目 60 分。

题型如下（每套题目会从以下三种形式中任意选择两种）：

1. 程序改错题。给出一段程序和程序功能的说明，考生在指定的程序段中找出一至两个错误来，修改之后调试运行。
2. 完成程序题。给出部分程序和程序功能的说明，考生根据程序的功能，将程序完成并调试运行。
3. 编程题。根据程序功能说明，考生完成整个程序的编写，并调试运行。

四. 考核时间

考试时间 90 分钟。

五. 考核范围

- C++编译器的使用。包括：Microsoft Visual C++ 6.0 集成开发环境的使用；新建工程，编辑含有类的 C++ 源文件，存取文件，编译文件，调试程序和运行程序。
- 类的概念及其使用方法。包括：类和对象的概念、内联函数和 struct 类、c++的作用域、对象的构造和析构过程、初始化列表构造对象的方法、静态成员、类的成员函数的重载、类的默认构造和复制构造、隐藏对象的产生过程、访问权限、继承和派生、友元、赋值兼容原则、两义性及其解决办法、虚基类、Const 和 volatile、转换函数、指向类成员的指针和 C++中可能的数组形式。
- 多态性和虚函数。包括：多态性的两种实现方法及其比较、虚函数和空的虚函数、纯虚函数和抽象类、虚函数和构造函数及析构函数的概念。
- 运算符重载和流类库。包括：各类运算符的重载、流类的基本类等级、预定义的流、文件流、格式控制和流的错误处理。
- 模板。包括：函数模板、类模板以及模板与继承。
- 程序设计的一般知识。包括：虚参和实参相结合的规则、变量的作用域、指针和引用以及外部函数等。

六. 例题

1. 下面的程序定义了一个 Point 类，找出程序中的错误（已经指出错误所在的范围），修改后请运行程序并保存结果。

```

#include <iostream.h>
class Point
{
public:
int x;
//下面三行中有一个错误 ①
Point(int a){x=a;}
int Getx(){return x;}
cout<<Getx()<<endl;
//下面三行中有一个错误 ②
} A
void main()
{
//下面三行中有一个错误③
A.Point(76);
Cout<<A.x;
Cout<<A.Getx()<<endl;
}

```

答案:

- ① 应该注释掉 `cout<<Getx()<<endl`。在类定义中不能直接使用 `cout` 语句向屏幕输出内容。
- ② `A` 后应该加上“;”，类的声明之后必须加上“;”。
- ③两种改法：为类 `Point` 增加一个成员函数如 `void Setx(int a){x=a;}`，然后将 `A.Point(76)`改为 `A.setx(76)` 去掉；或者去掉类声明后的 `A` 以及 `A.Point(76)`一句，在 `main` 函数中，声明一个 `Point` 的对象并初始化它，如 `Point B(76)`；然后将 `main` 函数中的 `A` 换成 `B`。构造对象时，不能显式地调用构造函数。

2. 根据基类的定义和测试程序以及程序中的注释信息，完成程序。

```

class Point
{
public:
Point(float xx=0,float yy=0){X=xx;Y=yy;}
void Move(float xOff, float yOff){X+=xOff;Y+=yOff;}
float Getx(){return X;}
float Gety(){return Y;}
private:
float x,y;
};
class Rectangle:private Point
{
public:
Rectangle(float x,float y,float w,float h)
:Point( ① ){ ② }
void Move(float xOff, float yOff){ ③ }
float GetX(){ ④ }
float GetY(){ ⑤ }
float GetH(){ ⑥ }
float GetW(){ ⑦ }
private:
float W,H;
}

```

```
//测试程序
#include <iostream.h>
#include<math.h>
void main()
{
Rectangle rect(5,8n25,15);
void Move(3,4);
cout<<"The data of rect(X,Y,W,H):"<<endl;
cout<<rect.GetX()<<"", "<< rect.GetY()<<"", "<< rect.GetW()
<<"", "<< rect.GetH()<<endl;
}
```

答案:

- ① x,y
- ② W=w;H=h;
- ③ Point::Move(xOff,yOff);
- ④ return Point::GetX();
- ⑤ return Point::GetY();
- ⑥ return H;
- ⑦ return W;

3. 用模板函数实现找出两个数值中最小值的程序，并给出一段简单的测试程序。

```
#include "iostream.h"
template <class T>
T tmin(T x,T y)
{
return (x<y?(x):(y));
}
main()
{
double m=3244.6577,n=899.9999;
int x=12,y=21;
long a=4356,b=98766;

cout<<tmin(x,y)<<endl;
cout<<tmin(a,b)<<endl;
cout<<tmin(m,n)<<endl;
}
```