

得分	评阅人

一、单项选择题：（共 20 题，每小题 2 分，共 40 分）

1. C++语言程序能够在不同操作系统下编译、运行，说明 C++具有良好的()
A. 适应性 **B. 移植性** C. 兼容性 D. 操作性
2. C++中，关于默认形参值，正确的描述是()
A. 设置默认形参值时，形参名不能缺省
B. 只能在函数定义时设置默认形参值
C. 应该先从右边的形参开始向左边依次设置
D. 应该全部设置
3. 对重载函数的要求中，正确的为()
A. 要求参数的个数不同 B. 要求参数中至少一个类型不同
C. 要求参数个数相同时，类型不同 D. 要求函数的返回类型不同
4. 若同时定义了如下函数，fun(8, 3.1) 调用的是下列哪个函数() **没有匹配的选项，选模板**
A. `template <class T1, class T2> fun (T1, T2)` B. `fun (double, int)`
C. `fun (char, float)` D. `fun (double, double)`
5. `int i=100;` 下列引用方法中，正确的是()
A. `int &r=i;` B. `int &r=100;` C. `int &r;` D. `int &r=&i;`
6. 类的实例化是指()
A. 定义类 **B. 创建类的对象**
C. 指明具体类 D. 调用类的成员
7. 不属于成员函数的是()
A. 静态成员函数 **B. 友元函数**
C. 构造函数 D. 析构函数
8. 下列运算符中，()运算符在 C++中不能重载。
A. `&&` B. `[]` **C. `:::`** D. `new`

9. 下列说法中正确的是()
- A. 类定义中只能说明函数成员的函数头, 不能定义函数体
 - B. 类中的函数成员可以在类体中定义, 也可以在类体之外定义**
 - C. 类中的函数成员在类体之外定义时必须要与类声明在同一文件中
 - D. 在类体之外定义的函数成员不能操作该类的私有数据成员
10. 已知类A是类B的友元, 类B是类C的友元, 则()
- A. 类A一定是类C的友元
 - B. 类C一定是类A的友元
 - C. 类C的成员函数可以访问类B的对象的所有成员
 - D. 类A的成员函数可以访问类B的对象的所有成员**

11. 有如下类定义:

```
class Point
{
    int x_, y_;
public:
    Point():x_(0), y_(0) {}
    Point(int x, int y=0):x_(x), y_(y) {}
};
```

若执行语句

```
Point a(2), b[3], *c[4];
```

则Point类的构造函数被调用的次数是()

- A. 2次
 - B. 3次
 - C. 4次**
 - D. 5次
12. 下面对派生类的描述中, 错误的是()
- A. 一个派生类可以作为另外一个派生类的基类
 - B. 派生类至少有一个基类
 - C. 派生类的成员除了它自己的成员外, 还包含了它的基类的成员
 - D. 派生类中继承的基类成员的访问权限到派生类中保持不变**

13. 有如下程序: **(先调用派生析构, 再调用基类析构)**

```
#include <iostream>
using namespace std;
class BASE{
public:
    ~BASE() { cout<<"BASE"; }
};
class DERIVED: public BASE {
public:
    ~DERIVED() { cout<<"DERIVED"; }
};
int main() {DERIVED x; return 0 ;}
```

执行后的输出结果是()

- A. BASE
- B. DERIVED
- C. BASEDERIVED
- D. DERIVEDBASE**

14. 当保护继承时,基类的()在派生类中成为保护成员,不能通过派生类的对象来直接访问。

- A. 任何成员
- B. 公有成员和保护成员
- C. 公有成员和私有成员
- D. 私有成员

15. 以下基类中的成员函数,哪个表示纯虚函数()

- A. virtual void vf(int);
- B. void vf(int)=0;
- C. virtual void vf()=0; (参数是空,把0赋值)
- D. virtual void yf(int) {}

16. 抽象类应含有()。

- A. 至多一个虚函数
- B. 至少一个虚函数
- C. 至多一个纯虚函数
- D. 至少一个纯虚函数

17. 有如下程序:

```
#include <iostream>
using namespace std;
class Base {
public:
    void fun1() {cout<<"Base\n";}
    virtual void fun2() {cout<<"Base\n";}
};
class Derived:public Base {
public:
    void fun1() {cout<<"Derived\n";}
    void fun2() {cout <<"Derived\n";}
};
void f(Base& b) {b.fun1();b.fun2();}
void main() {
    Derived obj;
    f(obj);
}
```

执行这个程序,输出结果是()

- | | | | |
|---------|---------|------------|------------|
| A. Base | B. Base | C. Derived | D. Derived |
| Base | Derived | Derived | Base |

18. 类模板的实例化()

- A. 在编译时进行
- B. 属于动态联编
- C. 在运行时进行
- D. 在连接时进行

19. 类模板的参数()

- A. 可以有多个
- B. 不能有基本数据类型
- C. 可以是0个
- D. 参数不能给初值

20. 以下类模板定义正确的为()

- A. template <class T, int i=0>
- B. template <class T, class int i>
- C. template <class T, typename T>
- D. template <class T1, T2> (没类型)

得分	评阅人

二、填空题题：（共 20 空，每空 1 分，共 20 分）

21. 布尔型数值只有两个： true ， false 。在 C++的算术运算式中，分别当作 1 ， 0 。
22. 带 inline 关键字定义的函数为 内联函数 ，在 编译 时将函数体展开到所有调用处。内联函数的好处是节省 内存 开销。
23. 每个对象都有一个指向自身的指针，称为 this 指针，通过使用它来确定其自身的地址。
24. 运算符 new 自动建立一个大小合适的对象并返回一个具有正确类型的指针。
25. C++程序设计的关键之一是利用 继承 实现软件重用，有效的缩短程序的开发时间。
26. 当私有派生时，基类的公有成员成为派生类的 私有成员 ；保护成员成为派生类的 私有成员 ；私有成员成为派生类的 不可访问成员 。
27. C++的动态多态性是通过 虚函数 实现的。（静态多态用 重载函数 实现）
28. 当通过 基类指针 使用虚函数时，C++会在与对象关联的派生类中正确的选择重定义的函数。

得分	评阅人

三、程序改错题：（共 2 题，每题 5 分，共 10 分）

29. 下列程序中如有错误与不妥当之处请指出。
- ```

//*****去掉“//”剩下“/*”
* 程序文件名: p1_2. cpp *
*****/
Using namespace std(小写 u, 加分号)
#include <iostream>; (不用分号)
using std::endl; 多余
void main()缺{
 float num1, num2, num3; //定义三个数
 cin << num1 << num2<<num3;
 cout >> “The average is:” >> setw(30) >>(num1+num2+num3)/3>> endl;
}加头文件 iomanip

```

错误原因:

改正:

30. 下面的程序有多处错误，说明错误原因，并改正错误。
- ```

Class A {
    int a(0), b(0);不能直接初始化
public:
    A(int aa, int bb) {a=aa;b=bb;}
}
A    x(2, 3), y(4);要额外加一个构造函数（重载）

```

错误原因:

得分	评阅人

四、程序分析题：（共 2 题，共 10 分）

31.

```
#include<iostream>
using namespace std;
void swap1(int x,int y) {
    int t;
    t=x,x=y,y=t;
}
void swap2(int *x,int *y) {
    int t;
    t=*x,*x=*y,*y=t;
}
void swap3(int& x, int& y) {
    int t;
    t=x,x=y,y=t;
}
void main()
{ int a=3,b=4;
  swap1(a,b);
  cout<<"a="<<a<<" b="<<b<<endl;
  swap2(&a,&b);
  cout<<"a="<<a<<" b="<<b<<endl;
  swap3(a,b);
  cout<<"a="<<a<<" b="<<b<<endl;
}
```

程序运行结果为： a=3, b=4 (没换)

a=4, b=3 (换了)

a=3, b=4 (再换一次)

32.

```
#include<iostream>
using namespace std;
class A{
public:
    virtual ~A(){
        cout<<"A::~~A() called "<<endl;    }
};
class B:public A{
    char *buf;
public:
    B(int i){    buf=new char[i];    }
    virtual ~B(){
        delete []buf;
        cout<<"B::~~B()    called"<<endl;
    }
};
void fun(A    *a) {
    delete a;
}
void main()
{    A    *a=new B(10); (类型兼容)
    fun(a);
}
```

运行结果为: B::~~B() called
A::~~A() called

得分	评阅人

五、程序设计题: (共 2 题, 共 25 分)

33. 定义一个 Point 类, 派生出 Rectangle 类和 Circle 类, 计算各派生类对象的面积 Area()。

34. 重载运算符<<, 使之能够使用 cout 将 Date 类对象的值以日期格式输出。(或将 Clock 类对象的值以时间格式输出)

还有简答题