

## C++选择题

1. C++语言是从早期的C语言发展演变而来的。与C相比,它在求解问题方法上进行的最大改进是( B )  
A)面向过程 B)面向对象 C)安全性 D)复用性
2. 下列的各类函数中,不是类的成员函数。( C )  
A)构造函数 B)析构函数 C)友元函数 D)拷贝构造函数
3. 作用域运算符“::”的功能是:( B )  
A)标识作用域的级别的 B)指出作用域的范围的  
C)给定作用域的大小的 D)标识成员是属于哪个类的
4. 下列说明中 const char \*ptr; 其中 ptr 应该是( C )  
A)指向字符常量的指针 B)指向字符的常量指针  
C)指向字符串常量的指针 D)指向字符串的常量指针
5. 已知: print()函数是一个类的常成员函数,它无返回值,下列表示中正确的是( A )。  
A)void print() const; B)const void print();  
C)void const print(); D)void print(const);
6. 可以被类定义的外部访问的成员有( C )  
A)所有类成员 B)private 的类成员  
C)public 的类成员 D)public 或 private 的类成员
7. 关于类和对象不正确的说法是:( C )  
A)类是一种类型,封装了数据和操作 B)对象是类的实例  
C)一个类的对象只有一个 D)一个对象必属于某个类
8. 在C++中,类与类之间的继承关系具有( C )  
A)自反性 B)对称性 C)传递性 D)反对称性
9. 在C++中实现封装是借助于( B )  
A)枚举 B)类 C)数组 D)函数
10. 在C++中用( D )能够实现将参数值带回。  
A)数组 B)指针 C)引用 D)上述ABC都可
11. 考虑函数原型 void test(int a,int b=7,char ch="\*"),下面的函数调用中不合法的是( C )  
A)test(5) B)test(5,8) C)test(6,"#") D)test(0,0,"\*");
12. 下列关于类和对象的叙述中,错误的是:( A )  
A)一个类只能有一个对象; B)对象是类的具体实例;  
C)类是对某一类对象的抽象; D)类和对象的关系是一种数据类型与变量的关系。
13. 已知 int m=10; 正确表示引用的是( C )  
A)int &Z; B)int &t=10; C)int &X=m; D)float &f=&m;
14. 对于 int \*pa[5]; 的描述中,正确的是( D )  
A)pa 是一个指向数组的指针,所指向的数组是 5 个 int 型元素  
B)pa 是一个指向某数组中第 5 个元素的指针,该元素是 int 型变量  
C)pa[5]表示数组的第 5 个元素的值,是 int 型的值  
D)pa 是一个具有 5 个元素的指针数组,每个元素是一个 int 型指针
15. 定义析构函数时,应该注意( C )  
A)其名与类名完全相同 B)返回类型是 void 类型  
C)无形参,也不可重载 D)函数体中必须有 delete 语句
16. 在公有继承的情况下,基类的成员(私有的除外)在派生类中的访问权限( B )  
A)受限制 B)保持不变 C)受保护 D)不受保护
17. 下列不是函数重载所要求的条件是( D )  
A)函数名相同 B)参数个数不同  
C)参数类型不同 D)函数返回值类型不同
18. 在C++中执行以下 4 条语句后输出 rad 值为( C )  
static int hot=200; int &rad=hot;  
hot=hot+100; cout<<rad<<endl;  
A) 0 B) 100 C) 300 D) 200
19. 类 class CC1 的说明如下,错误的语句是( A )  
Class CC1{  
int a=2; //(A)  
CC1(); //(B)  
public:  
CC1(int val); //(C)  
~CC1(); }; //(D)
20. 类的指针成员的初始化是通过函数完成的,这个函数通常是( C )  
A)析构函数 B)构造函数 C)其它成员函数 D)友元函数
21. 关于构造函数的说法,不正确的是( A )  
A)没有定义构造函数时,系统将不会调用它  
B)其名与类名完全相同 C)它在对象被创建时由系统自动调用  
D)没有返回值
22. 在下列程序划线处填入的正确语句是( C )  
#include <iostream.h>  
class Base  
{ public:  
void fun(){cout<<"Base::fun"<<endl; } };  
class Derived:public Base{  
public:  
void fun()  
{ //此处调用基类的函数 fun()  
cout<<"Derived::fun"<<endl; } };  
A)fun(); B)Base.fun(); C)Base::fun(); D)Base->fun();
23. 在保护继承的情况下,基类的成员(私有的除外)在派生类中的访问权限( C )  
A)受限制 B)保持不变 C)受保护 D)不受保护
24. 依据如下一段程序,不合法的是( C )  
int f1(float); int f2(char); int f3(float); int f4(float);  
int (\*pf)(float);  
A) pf=&f1; B) pf=&f4; C) pf=&f2; D) pf=&f3;
25. 通常拷贝构造函数的参数是:( C )  
A)对象名 B)对象的成员名  
C)对象的引用名 D)对象的指针名
26. 系统在调用重载函数时,不能作为确定哪个重载函数被调用的依据是( D )  
A)参数个数 B)参数类型  
C)函数名称 D)函数的返回值类型
27. 下列描述中,正确的是( D )  
A)所有的运算符都可以重载。 B)基类类型的指针可以指向子类,子类类型的指针也可以指向基类。  
C)可以在类的构造函数中对静态数据成员进行初始化。 D)动态联编要满足两个条件:被调用的成员函数是虚函数;用指针或引用调用虚函数。
28. 类 TM 和类 TN 是两个独立的类,那么类 TM 中哪种形式的数据成员一定是不允许的( C )  
A)TM \* a B)TN & a C)TM a D)TN a;

29. 关于成员函数特征的描述中, 错误的是 ( A )  
A) 成员函数一定是内联函数 B) 成员函数可以重载  
C) 成员函数可以设置参数的默认值  
D) 成员函数可以是静态的

30. 下列关于构造函数的描述中, 错误的是 ( D )  
A) 构造函数可以设置默认参数; B) 构造函数在定义类对象时自动执行  
C) 构造函数可以是内联函数;  
D) 构造函数不可以重载

31. 下面描述中, 表达错误的是 ( B )  
A) 公有继承时, 基类中的 public 成员在派生类中仍是 public 的 B) 公有继承时, 基类中的 private 成员在派生类中仍是 private 的  
C) 公有继承时, 基类中的 protected 成员在派生类中仍是 protected 的 D) 私有继承时, 基类中的 public 成员在派生类中是 private 的

32. 有如下类声明“class A{int x; .....};”, 则 A 类的成员 x 是 ( B )

A) 公有数据成员 B) 私有数据成员  
C) 公有成员函数 D) 私有成员函数

33. 有如下程序: 执行后的输出结果应该是 ( A )

```
#include<iostream.h>
class cla{
    static int n;
public:
    cla(){n++;}
    ~cla(){n--;}

    static int get_n(){return n;} };
int cla::n=0;
int main()
{ cla * p=new cla;
  delete p;
  cout<<"n="<<cla::get_n()<<endl;
  return 0; }
```

A)n=0 B)n=1 C)n=3 D)n=4

34. 有如下程序: 执行后的输出结果应该是 ( A )

```
#include <iostream.h>
class A{
public:
    A(){cout<<"A"; } };
class C:public A{
public:
    C(){cout<<"C"; } };
void main(){ C cobj; }
```

A) AC B) CA C) A D) C

35. 有如下程序: 执行后的输出结果应该是: ( D )

```
#include <iostream.h>
class BASE{
public:
    ~BASE(){cout<<"BASE"; } };
class DERIVED: public BASE{
public:
    ~DERIVED(){cout<<"DERIVED"; } };
void main(){ DERIVED x; }
```

A) BASE B) DERIVED  
C) BASEDERIVED D) DERIVEDBASE

36. 创建动态对象, 需要使用操作符 ( A )

A) new B) delete C) . D) \*

37. 删除动态对象, 需要使用操作符 ( B )

A) new B) delete C) . D) \*

38. C++中规定不能被重载的一组运算符是 ( A )

A) :: ?: . sizeof B) ?: . sizeof ++ ->

C) :: ?: = , D) ++ -- ?: sizeof

39. 适宜采用 inline 定义内联函数情况是 ( C )

A) 函数体含有循环语句 B) 函数体含有递归语句  
C) 函数代码少、频繁调用 D) 函数体含有 switch 语句

40. 下述调用虚函数时采用动态联编的是 ( A )

A) 对象指针 B) 对象名 C) 成员名限定 D) 派生类名

41. 如果类 A 被说明成类 B 的友元, 则 ( C )

A) 类 A 的成员即是类 B 的成员  
B) 类 B 的成员即是类 A 的成员  
C) 类 A 的成员函数可以访问类 B 的所有成员  
D) 类 B 的成员函数可以访问类 A 的所有成员

42. 下面关于友元的描述中, 错误的是 ( D )

A) 友元函数可以直接访问该类的私有成员。 B) 一个类的友元类中的所有成员函数都是这个类的友元函数。  
C) 利用友元可以提高程序的运行效率, 但却破坏了封装性。 D) 友元关系不能被继承, 是双向可交换的。

43. 下列虚基类的声明中, 正确的是 ( B )

A) class virtual B: public A B) class B: virtual public A  
C) class B: public A virtual D) virtual class B: public A

44. C++设置虚基类的目的是 ( A )

A) 消除二义性 B) 简化程序 C) 提高运行效率  
D) 减少目标代码

45. 依据如下模板定义, 调用 fun 时错误的是 ( C )

```
template <class T>T fun(T x,T y){return x*x+y*y; }
```

A) fun(2,8) B) fun(2.0,8.2)  
C) fun(2.3,8) D) fun<float>(2,8.3)

46. 一般情况函数模板中的类型参数个数不能是 ( A )

A) 0 个 B) 1 个 C) 2 个 D) 3 个

47. 以下关于 this 指针的叙述中正确的是 ( D )

A) 任何与类相关的函数都有 this 指针; B) 类的成员函数都有 this 指针; C) 类的友元函数都有 this 指针; D) 类的非静态成员函数才有 this 指针;

48. 当一个类的某个函数被说明为 virtual 时, 在该类的所有派生类中的同原型函数 ( A )

A) 都是虚函数 B) 只有被重新说明时才是虚函数  
C) 都不是虚函数 D) 只有重新说明为 virtual 才是虚函数

49. 在派生类中重新定义虚函数时, 必须在下列哪方面与基类保持一致。 ( A )

A) 参数类型 B) 参数名字 C) 操作内容 D) 赋值

50. 关于 const 关键字说法错误的是 ( D )

A) const 关键字可以修饰对象和成员函数 B) const 对象不能被修改 C) const 成员函数不能修改类数据成员 D) const 可以用于说明类

51. 在 C++中使用流进行输入输出, 其中用于屏幕输出的对象是: ( C )

A) cerr B) cin C) cout D) cfile

52. 不能用来声明类的访问权限的关键字是 ( C )

A) public B) private C) static D) protected

53. 执行如下程序后的输出结果是: ( B )

```
#include<iostream.h>
class test{
    static int count;
public:
    test(){count++; }
    ~test(){count--; }
    static int getCount(){return count; }    };
int test::count=0;
int main()
{ test * p=new test;
  test * q=new test;
  delete p;
  cout<<"count="<<test::getCount()<<endl;
  return 0; }
```

A)count=0 B)count=1 C)count=2 D)count=3

54. 关于类模板的说法正确的是: ( B )

- A)类模板的主要作用是生成抽象类 B)类模板实例化时, 编译器将根据给出的模板实参生成一个类  
C)在类模板中的数据成员具有同样类型  
D)类模板中的成员函数没有返回值

55. 实现运行时的多态性采用( D )

- A)重载函数 B)构造函数 C)析构函数 D)虚函数

56. 下列说明语句中正确的是 ( D )

- A)int a, &ra=a, &&ref=ra; B)int &refa[10];  
C)int a, &ra=a, &\*refp=&ra; D)int \*pi, \*&pref=pi;

57. 使用 setw()格式操纵符时需要包含头文件 ( C )

- A)iostream.h B)fstream.h C)iomanip.h D)stdlib.h

58. 未使用关键字定义类的数据成员, 默认为 ( A )

- A)private B)public C)protected D)friend

59. 值传递方式将实参传给形参, 正确说法是 ( A )

- A)形参是实参的备份 B)实参是形参的备份  
C)形参和实参是同一对象 D)形参和实参无联系

60. 函数调用时对某一参数指明一个特定值, ( A )

- A)其之前所有参数都必须赋值 B)其之后所有参数都必须赋值  
C)其前、后所有参数都必须赋值  
D)其前、后所有参数都不必赋值

61. 设函数 int max(int, int)返回两参数中较大值, 若求 22、59、70 中最大值, 不正确的表达式是 ( C )

- A)int m = max(22, max(59, 70));  
B)int m = max(max(22, 59), 70);  
C)int m = max(22, 59, 70);  
D)int m = max(59, max(22, 70));

62. 如果没有为一个类定义任何构造函数的情况下, 下列描述正确的是 ( A )

- A)编译器总是自动创建一个不带参数的构造函数  
B)这个类没有构造函数  
C)这个类不需要构造函数 D)该类不能通过编译

63. 下列哪个类型函数不适合声明为内联函数 ( A )

- A)函数体语句较多 B)函数体语句较少  
C)函数执行时间较短 D)函数执行时间过长

64. int F(int, int); 不可与哪个函数构成重载 ( B )

- A)int F(int, int, int); B)double F(int, int);  
C)double F(double, double); D)double F(int, double);

65. 类的私有成员可在何处访问 ( D )

- A)通过子类的对象访问 B)本类及子类的成员函数中  
C)通过该类对象访问 D)本类的成员函数中

66. 一个类可包含析构函数的个数是: ( B )

- A)0 个 B)1 个 C)0 个或 1 个 D)0 个或多个

67. 一个类可包含构造函数的个数是 ( D )

- A)0 个 B)0 个或 1 个 C)0 个或多个 D)1 个或多个

68. this 指针存在的目的是 ( B )

- A)保证基类公有成员在子类中可以被访问  
B)保证每个对象拥有自己的数据成员, 但共享处理这些数据成员的代码  
C)保证基类保护成员在子类中可以被访问  
D)保证基类私有成员在子类中可以被访问

69. 下列关于类的权限的描述错误的是 ( A )

- A)类本身的成员函数只能访问自身的私有成员  
B)类的对象只能访问该类的公有成员  
C)普通函数不能直接访问类的公有成员, 必须通过对象访问  
D)一个类可以将另一个类的对象作为成员

70. 在编译指令中, 宏定义使用哪个指令 ( B )

- A)#include B)#define C)#if D)#else

71. 设类 A 将其它类对象作为成员, 则建立 A 类对象时, 下列描述正确的是 ( B )

- A)A 类构造函数先执行 B)成员构造函数先执行  
C)两者并行执行 D)不能确定

72. 对于友元描述正确的是 ( B )

- A)友元是本类的成员函数 B)友元不是本类的成员函数  
C)友元不是函数 D)友元不能访问本类私有成员

73. 下列描述错误的是 ( A )

- A)在创建对象前, 静态成员不存在 B)静态成员是类的成员  
C)静态成员不能是虚函数  
D)静态成员函数不能直接访问非静态成员

74. 在何种派生方式中, 派生类可以访问基类中的 protected 成员 ( B )

- A)public 和 private B)public 和 protected  
C)protected 和 private D)仅 protected

75. 假定有类 AB, 有相应的构造函数定义, 能正确执行“AB a(4),b(5),c[3], \*p[2]={&a,&b};”语句, 请问执行完此语句后共调用该类构造函数的次数为 ( C )

- A)3 B)4 C)5 D)6

76. 对类中声明的变量, 下列描述中正确的是: ( B )

- A)属于全局变量 B)属于该类, 某些情况下也可被该类不同实例所共享  
C)只属于该类  
D)任何情况下都可被该类所有实例共享

77. 若定义了函数 double \*function(), 则函数 function 的返回值为 ( B )。

- A. 实数型 B. 实数的地址  
C. 指向函数的指针 D. 函数的地址

78. 下列的符号常量定义中, 错误的定义是 ( C )。

- A. const M=10; B. const int M=20;  
C. const char ch; D. const bool mark=true;



79. 要对类 AB 定义加号操作符重载成员函数，实现两个 AB 类对象的加法，并返回相加结果，则该成员函数的声明语句为（ C ）

- A) AB operator+(AB & A, AB & B);
- B) AB & operator+(AB A);
- C) AB operator+(AB & A);
- D) AB \* operator+(AB A);

80. 41. 以下关于函数模板叙述正确的是（ C ）。

- A. 函数模板也是一个具体类型的函数
- B. 函数模板的类型参数与函数的参数是同一个概念
- C. 通过使用不同的类型参数，函数模板可以生成不同类型的函数
- D. 用函数模板定义的函数没有类型

81. 下列（ C ）的调用方式是引用调用。

- A. 形参和实参都是变量
- B. 形参是指针实参是地址值
- C. 形参是引用实参是变量
- D. 形参是变量实参是地址值

82. 为了提高程序的运行速度，可将不太复杂的功能用函数实现，此函数应选择（ A ）。

- A. 内联函数
- B. 重载函数
- C. 递归函数
- D. 函数模板

83. 函数原型语句正确的是（ B ）。

- A. int Function(void a);
- B. void Function (int);
- C. int Function(a);
- D. void int(double a);

84. 以下函数的返回结果是（ A ）。

```
int function(char *x) {  
    char *p=x; while(*p++); return(p-x-1);  
}
```

- A. 求字符串的长度
- B. 将字符串 x 连接到字符串 p 后面
- C. 将字符串 x 复制到字符串 p
- D. 将字符串 x 反向存放

85. 用数组名作为函数调用的实参，（ A ）传递给形参

- A. 数组的首地址
- B. 数组中第一个元素的值
- C. 数组全部元素的值
- D. 数组元素的个数

86. 以下正确的描述是（ C ）。

- A. 不允许设置参数的默认值
- B. 设置参数的默认值只能在定义函数时设置
- C. 设置参数的默认值时，应该设置右边的参数
- D. 设置参数的默认值时，应该全部参数都设置

87. 采用重载函数的目的是（ D ）。

- A. 实现共享
- B. 减少空间
- C. 提高速度
- D. 使用方便，提高可读性

88. 假定 p 是一个指向 float 型数据的指针，则 p+1 所指数据的地址比 p 所指数据的地址大（ C ）个字节。

- A. 1
- B. 2
- C. 4
- D. 8

89. 在下面的函数声明中，存在着语法错误的是（ D ）。

- A. BC(int a, int);
- B. BC(int, int);
- C. BC(int, int=5);
- D. BC(int x; int y);

90. 假定 AB 为一个类，px 为指向该类长度为 n 的动态对象数组的指针，则执行“delete []px;”语句时，自动调用该类析构函数的次数为（ B ）。

- A. 1
- B. n
- C. n-1
- D. n+1

91. “void BC(int a, int b);”的重载函数是（ B ）。

- A. int BC(int x, int y);
- B. void BC(int a, char b);

C. float BC(int a, int b, int c=0);

D. int BC(int a, int b=0);

92. 打开文件“d:\file.dat”写入数据，正确语句是（ B ）

- ifstream infile("d:\file.dat", ios::in);
- ifstream infile("d:\\file.dat", ios::in);
- ofstream infile("d:\file.dat", ios::out);
- fstream infile("d:\\file.dat", ios::in|ios::out);

93. 进行文件操作时需要包含头文件（ C ）

- A) iostream.h
- B) stdio.h
- C) fstream.h
- D) stdlib.h

94. 有如下说明语句 int a[10]={1,2,3,4,5,6,7,8,9,10}; int \*p=a; 则数值为 9 的表达式是：（ B ）。

- A)\*p+9
- B)\*(p+8)
- C)\*p+=9
- D)p+7

95. 程序运行中需要从键盘上输入多于一个数据时，各数据之间应使用（ D ）符号作为分隔符。

- A) 空格或逗号
- B) 逗号或回车
- C) 分号
- D) 空格或回车

96. 关于构造函数的规定有（ ABCD ）

- A) 构造函数必须与相应的类同名
- B) 可以重载
- C) 可以带参数或不带参数
- D) 不能制定其返回值的类型

97. 关于析构函数的规定有（ ABCD ）

- A) 每个类最多只能有一个析构函数
- B) 应为 public 类型
- C) 不能被重载
- D) 不能返回任何值

98. 对象赋初值的常见方式有（ ACD ）

- A) 通过默认构造函数
- B) 通过析构函数
- C) 通过重载构造函数
- D) 通过对象赋值语句

99. 需要调用拷贝构造函数的情况有（ ACD ）

- A) 用对象给对象赋值
- B) 调用友员函数
- C) 对象作为函数的参数
- D) 对象作为函数的返回值

100. 类的对象成员的初始化方式有（ CD ）

- A) 直接访问对象的公有数据成员
- B) 访问对象的公有成员函数
- C) 构造函数方式
- D) 对象成员列表方式

101. 类的静态数据成员和静态成员函数的目的是（ AC ）

- A) 供所用对象共享
- B) 打破类的封装限制
- C) 节省内存
- D) 增加对象的访问权限

102. 在类中定义一个静态数据成员需要（ BD ）

- A) 保存静态数据成员的内存地址
- B) 在类中使用关键字 static 声明静态数据成员
- C) 在类中为静态数据成员分配内存空间并初始化
- D) 在类外为静态数据成员分配内存空间并初始化

103. 一个类的友元可以是（ ACD ）

- A) 外部的友元函数
- B) 友元数据成员
- C) 友员类
- D) 友元成员函数

104. C++ 中关键字 Const 修饰数据成员后，这些常数据成员在使用上的限制包括（ ABD ）

- A) 常数据成员的值不能更新
- B) 不能接收另一个变量的赋值
- C) 不能被访问
- D) 常数据成员必须进行初始化

105. 常成员函数是用 const 声明的成员函数，在常成员函数中不能（ AD ）

- A) 调用非 const 成员函数
- B) 调用 const 成员函数
- C) 访问类的私有成员
- D) 对数据成员进行更改