|  |
| --- |
| 全国计算机等级考试 历年真题　2008年9月计算机二级考试C++笔试题 |
| **一、选择题（每题2分，共计70分）** |
| |  |  | | --- | --- | | 1.一个栈的初始状态为空。现将元素1、2、3、4、5、A、B、C、D、E依次入栈，然后 再依次出栈，则元素出栈的顺序是 A)12345ABCDE　　　　　　B)EDCBA54321 C)ABCDE12345　　　　　　D)54321EDCBA |  | | A       B       C       D | |
|  |
| 2. 下列叙述中正确的是 A)循环队列有队头和队尾两个指针，因此，循环队列是非线性结构 B)在循环队列中，只需要队头指针就能反映队列中元素的动态变化情况 C)在循环队列中，只需要队尾指针就能反映队列中元素的动态变化情况 D)循环队列中元素的个数是由队头指针和队尾指针共同决定 |
| A       B       C       D |
| 3. 在长度为n的有序线性表中进行二分查找，最坏情况下需要比较的次数是 A)O(n)　　　　B)O(n2)　　　　C)O(log2n)　　　　D)O(nlog2n) |
| A       B       C       D |
| 4. 下列叙述中正确的是 A)顺序存储结构的存储一定是连续的，链式存储结构的存储空间不一定是连续的 B)顺序存储结构只针对线性结构，链式存储结构只针对非线性结构 C)顺序存储结构能存储有续表，链式存储结构不能存储有序表 D)链式存储结构比顺序存储结构节省存储空间 |
| A       B       C       D |
| 5. 数据流图中带有箭头的线段表示的是 A)控制流　　　　　　　　B)事件驱动 C)模块调用　　　　　　　D)数据流 |
| A       B       C       D |
| 6. 在软件开发中，需求分析阶段可以使用的同居是 A)N-S图　　　　　　B)DFD图　　　　　　C)PAD图　　　　D)程序流程图 |
| A       B       C       D |
| 7. 在面向对象方法中，不属于“对象”基本特点的是 A)一致性　　　　B)分类性　　　　C)多态性　　　　D)标识唯一性 |
| A       B       C       D |
| 8. 一间宿舍可以住多个学生，则实体宿舍和学生之间的联系是 A)一对一　　　　B)一对多　　　　C)多对一　　　　D)多对多 |
| A       B       C       D |
| 9. 在数据管理技术发展的三个阶段中，数据共享最好的是 A)人工管理阶段　　　　　　　　B)文件系统阶段 C)数据库系统阶段　　　　　　　D)三个阶段相同 |
| A       B       C       D |
| 10.有三个关系R、S、和T如下： http://www.passtwo.com/c15/access/x2008_09_10.jpg 由关系R和S通过运算得到关系T,则所使用的运算为 A)笛卡尔积　　　　B)交　　　　C)并　　　　D)自然连接 |
| A       B       C       D |
| 11. 在下列原型所示的C++函数中，按“传值”方式传递参数的是 A)void　f1(int　x);　　　　　　　　B)void　f2(int　\*x); C)void　f3(const　int　\*x);　　　　D)void　f4(int　&x); |
| A       B       C       D |
| 12. 在C++中，编译系统自动为一个类生成缺省构造函数的条件是 A)该类没有定义任何有参构造函数 B)该类没有定义任何无参构造函数 C)该类没有定义任何构造函数 D)该类没有定义任何成员函数 |
| A       B       C       D |
| 13. 在一个派生类的成员函数中，试图调用其基类的成员函数“void　f();”,但无法通过 编译，这说明 A)f()是基类的私有成员　　　　B)f()是基类的保护成员 C)派生类的继承方式为私有　　　D)派生类的继承方式为保护 |
| A       B       C       D |
| 14. 下列关于运算符重载的叙述中，错误的是 A)有的运算符可以作为非成员函数重载 B)所有的运算符都可以通过重载而被赋予新的含义 C)不得为重载的运算符函数的参数设置默认值 D)有的运算符只能作为成员函数重载 |
| A       B       C       D |
| 15. 下列关于模板的叙述中，错误的是 A)调用模板函数是，在一定条件下可以省略模板实参 B)可以用int,double这样的类型修饰符来生命模板参数 C)模板声明中的关键字class都可以用关键字typename代替 D)模板的形参表中可以有多个参数 |
| A       B       C       D |
| 16. 要利用c++流进行文件操作，必须在程序中包含的头文件是 A)iostream　　　　B)istream　　　　C)strstream　　　　D)iomanip |
| A       B       C       D |
| 17. 下列字符串中不能作为c++标识符使用的是 A)WHILE　　　　B)user　　C)\_lvar　　　　D)9stars |
| A       B       C       D |
| 18. 下列语句中错误的是 A)const　int　a;　　　　　　　　B)const　int　a=10; C)const　int　\*point=0;　　　　　D)const　int　\*point=new　int(10); |
| A       B       C       D |
| 19. 有如下程序 　　#include　<iostream> 　　using　namespace　std; 　　 　　int　main(){ 　　　　　　int　sum; 　　　　　　for(int　i=0;i<6;i+=3){ 　　　　　　　　　　sum=i; 　　　　　　　　　　for(int　j=i;j<6;j++)　sum+=j; 　　　　　　} 　　　　　　cout<<sum<<endl; 　　　　　　return　0; 　　} 运行时的输出结果是 A)3　　　　　　　B)10　　　　C)12　　　　　D)15 |
| A       B       C       D |
| 20. 下列语句中，正确的是 A)char　\*myString="Hello-World!"; B)char　myString="Hello-World!"; C)char　myString[11]="Hello-World!"; D)char　myString[12]="Hello-World!"; |
| A       B       C       D |
| 21. 若已经生命了函数原型“void　fun(int　a,double　b=0.0);”,则下列重载函数中正 确的是 A)void　fun(int　a=90,double　b=0.0);　　　　B)void　fun(int　a,double　B); C)void　fun(double　a,int　B);　　　　　　　　D)bool　fun(int　a,double　b=0.0); |
| A       B       C       D |
| 22. 有如下程序 　　#include　<iostream> 　　using　namespace　std; 　　class　Sample{ 　　　　　　public: 　　　　　　　　　　Sample(){} 　　　　　　　　　　~Sample(){cout<<"\*";} 　　}; 　　int　main(){ 　　　　　　Sample　temp[2],\*pTemp[2]; 　　　　　　return　0; 　　} 执行这个程序输出星号(\*)的个数为 A)1　　　　　B)2　　　　　C)3　　　　　D)4 |
| A       B       C       D |
| 23. 下列选项中，与实现运行时多态性无关的是 A)重载函数　　　　B)虚函数　　　　C)指针　　　　D)引用 |
| A       B       C       D |
| 24. 下列运算符函数中，肯定不属于类Value的成员函数的是 A)Value　operator+(Value);　　　　　B)Value　operator-(Value,Value); C)Value　operator\*(int);　　　　　　D)Value　operator/(Value); |
| A       B       C       D |
| 25. 下列模板生命中，有语法错误的是 A)template<typename　T>　fun(T　x){return　x;} B)template<typename　T>　fun(T　x,int　n){return　x\*n;} C)template<typename　T>　fun(T　\*p){return　\*p;} D)template<typename　T>　class　A{T　n;}; |
| A       B       C       D |
| 26. 在语句“cout<<'A';”中，cout是 A)类名　　　　　B)对象名　　　C)函数名　　　　　D)C++中的关键字 |
| A       B       C       D |
| 27. 有如下程序： 　　#include　<iostream> 　　using　namespace　std; 　　class　MyClass{ 　　　　　　public: 　　　　　　　　　　MYClass(int　i=0){cout<<i;} 　　　　　　　　　　MyClass(const　MyClass　&x){cout<<2;} 　　　　　　　　　　MyClass&　operator=(const　MyClass&x){cout<<3;return　\*this;} 　　　　　　　　　　~MyClass(){cout<<4;} 　　}; 　　int　main(){ 　　　　　　MyClass　obj1(1),obj2(2),obj3(obj1); 　　　　　　return　0; 　　} 运行时的输出结果是 A)112444　　　　　　B)11114444　　　　C)121444　　　　　D)11214444 |
| A       B       C       D |
| 28. 有如下程序： 　　#include　<iostream> 　　using　namespace　std; 　　class　MYClass{ 　　　　　　public: 　　　　　　　　　　MYClass(int　x):val(x){} 　　　　　　　　　　void　Set(int　x){val=x;} 　　　　　　　　　　void　Print()　const{cout<<"val="<<val<<'\t';} 　　　　　　private: 　　　　　　　　　　int　val; 　　}; 　　int　main(){ 　　　　　　const　MyClass　obj1(10); 　　　　　　MyClass　obj2(20); 　　　　　　obj1.Print();　　　　//语句1 　　　　　　obj2.Print();　　　　//语句2 　　　　　　obj1.Set(20);　　　　//语句3 　　　　　　obj2.Set(30);　　　　//语句4 　　　　　　return　0; 　　} 其主函数中错误的语句是 A)语句1　　　　　B)语句2　　　　　C)语句3　　　　　D)语句4 |
| A       B       C       D |
| 29. 在类生命中，紧跟在“public:”后生命的成员的访问权限是 A)私有　　　B)公有　　　C)保护　　　D)默认 |
| A       B       C       D |
| 30. 对于通过公有继承定义的派生类，若其成员函数可以直接访问基类的某个成员，说 明该基类成员的访问权限是 A)公有或私有　　　B)私有　　　C)保护或私有　　　D)公有或保护 |
| A       B       C       D |
| 31. 定义派生类是，若不使用关键字显示地规定采用何种继承方式，则默认方式为 A)私有继承　　　B)非私有继承　　　C)保护继承　　　D)公有继承 |
| A       B       C       D |
| 32. 建立一个有成员对象的派生类对象是，各构造函数体的执行次序为 A)派生类、成员对象类、基类　　　B)成员对象类、基类、派生类 C)基类、成员对象类、派生类　　　D)基类、派生类、成员对象类 |
| A       B       C       D |
| 33. 如果表达式a>=b中的">="是作为非成员函数重载的运算符，则可以等效地表示 为 A)a.operator>=(b)　　　B)b.operator>=(a) C)operator>=(a,b)　　　D)operator>=(b,a) |
| A       B       C       D |
| 34. 当使用ofstream流类定义一个流对象并打开一个磁盘文件是，文件的默认打开方式 为 A)ios\_base::in　　　　　　　　B)ios\_base::binary C)ios\_base::in|ios\_base:out　　D)ios\_base::out |
| A       B       C       D |
| 35. 在一个抽象类中，一定包含有 A)虚函数　　　B)纯虚函数　　　C)模板函数　　　D)重载函数 |
| A       B       C       D |
| **二、填空题（每空2分，共计30分）** | |
| 1.对下列二叉树进行中序遍历的结果是【1】 http://www.passtwo.com/c15/c/t2008_09_01.jpg |  |
| 输入答案，中间不含空格: |  |
| 2. 按照软件测试的一般步骤，继承测试应在【2】测试之后进行。 |  |
| 输入答案，中间不含空格: |  |
| 3. 软件工程三要素包括方法、工具和过程，其中，【3】支持软件开发的各个环节 的控制和管理。 |  |
| 输入答案，中间不含空格: |  |
| 4. 数据库设计包括概念设计、【4】和物理设计。 |  |
| 输入答案，中间不含空格: |  |
| 5. 在二维表中，元组的【5】不能再分成更小的数据项。 |  |
| 输入答案，中间不含空格: |  |
| 6. 当使用关键字【6】作为函数返回类型是，该函数不返回任何值。 |  |
| 输入答案，中间不含空格: |  |
| 7. 已知数组a中有n个元素，下列语句将数组a从下标x1开始的k个元素移动到从 下标x2开始的k个元素中，其中0<=x1<x2<n,x2+k<n,请将下列语句补充完整。 　　　　for(int　i=x1+k-1;i>=x1;i--) 　　　　　　a[【7】]=a[i]; |  |
| 输入答案，中间不含空格: |  |
| 8. 有如下程序： 　　#include　<iostream> 　　using　namespace　std; 　　int　fun1(int　x)　{return　++x;} 　　int　fun2(int　&x){return　++x;} 　　int　main(){ 　　　　　　int　x=1,y=2; 　　　　　　y=fun1(fun2(x)); 　　　　　　count<<x<<','<<y; 　　　　　　return　0; 　　}  程序的输出结果是【8】。 |  |
| 输入答案，中间不含空格: |  |
| 9. 有如下程序： 　　#include　<iostream> 　　using　namespace　std; 　　class　pumpkin{ 　　　　　　public: 　　　　　　　　　　pumpkin(){++count;} 　　　　　　　　　　~pumpkin(){--count;} 　　　　　　　　　　static　void　total\_count(){ 　　　　　　　　　　　　　　　　cout<<count<<"pumpkin(s)"<<endl; 　　　　　　　　　　} 　　　　　　private: 　　　　　　　　　　static　int　count; 　　}; 　　int　main(){ 　　　　　　pumpkin　p1[10]; 　　　　　　pumpkin:total\_count(); 　　　　　　return　0; 　　} 这个程序输出的结果是【9】 |  |
| 输入答案，中间不含空格: |  |
| 10. 有如下递归函数： 　　int　Fun(int　n){ 　　　　　if(n<=1)　return　1; 　　　　　【10】 　　} 请补充完整，使得函数Fun能够正确计算形参n的阶乘 |  |
| 输入答案，中间不含空格: |  |
| 11. 请在下列程序的空格出填写正确的语句： 　　class　Sample{ 　　　　　　public: 　　　　　　　　　　Sample(){} 　　　　　　　　　　~Sample(){} 　　　　　　　　　　void　SetData(int　data){//将Sample类成员变量data设置成形参的值 　　　　　　　　　　　　　【11】 　　　　　　　　　　} 　　　　　　private:int　data; 　　}; |  |
| 输入答案，中间不含空格: |  |
| 12. 有如下类定义，请将Sample类的拷贝构造函数补充完整。 　　class　Sample{ 　　　　　　public: 　　　　　　　　　　Sample(){} 　　　　　　　　　　~Sample(){if(p)　delete　p;} 　　　　　　　　　　Sample(const　Sample&　s){ 　　　　　　　　　　　　　　【12】 　　　　　　　　　　} 　　　　　　　　　　void　SetData(int　data){p=new　int(data);} 　　　　　　private: 　　　　　　　　　　int　\*p; 　　}; |  |
| 输入答案，中间不含空格: |  |
| 13. 重载加法运算符“+”，其函数名是【13】。 |  |
| 输入答案，中间不含空格: |  |
| 14. 有如下程序： #include　<iostream> using　namespace　std; class　Wages{　　//“工资”类 　　　　double　base;　//基本工资 　　　　double　bonus;//奖金 　　　　double　tax;//税金 　　　　public: 　　　　　　　　Wages(double　CBase,double　CBonus,double　CTax): 　　　　　　　　base(CBase),bonus(CBonus),tax(CTax){} 　　　　　　　　double　getPay()const;　　　//返回应付工资额 　　　　　　　　Wages　operator+(Wages　w)const;//重载加法 }; 　　double　Wages::getPay()const{return　base+bonus-tax;} 　　　Wages　Wages::operator　+(Wages　w)const{ 　　 return　Wages(base+w.base,bonus+w.bonus,tax+w.tax); 　　　}  int　main(){ 　　　　Wages　w1(2000,500,100),w2(5000,1000,300); 　　　　cout<<(w1+w2).getPay()<<endl; 　　　　return　0; } 程序的输出结果是【14】 |  |
| 输入答案，中间不含空格: |  |
| 15. 有如下程序： 　　#include　<iostream> 　　using　namespace　std; 　　class　Pet{ 　　　　　　char　name[10]; 　　　public: 　　　　　　　Pet(char　\*name){strcpy(this->name,name);} 　　　　　　　const　char　\*getName()const{return　name;} 　　　　　　　virtual　void　call()const=0; 　　}; 　　 　　class　Dog:public　Pet{ 　　public: 　　　　　　Dog(char　\*name):Pet(name){} 　　　　　　void　call()const{cout<<"汪汪叫";} 　　}; 　　 　　class　Cat:public　Pet{ 　　public: 　　　　　　Cat(char　\*name):Pet(name){} 　　　　　　void　call()const{cout<<"喵喵叫";} 　　}; 　　 　　int　main(){ 　　　　　　Pet　\*pet1=new　Dog("哈克"),\*pet2=new　Cat("吉米"); 　　　　　　cout<<pet1->getName();pet1->call();cout<<endl; 　　　　　　cout<<pet2->getName();pet2->call();cout<<endl; 　　　　　　return　0; 　　} 程序的输出结果是【15】。 |  |
| 输入答案，中间不含空格: |  |

|  |  |  |  |
| --- | --- | --- | --- |
| 序号 | 您的答案 | 正确答案 | 得分 |
| 1 |  | B | 0 |
| 2 |  | D | 0 |
| 3 |  | C | 0 |
| 4 |  | A | 0 |
| 5 |  | D | 0 |
| 6 |  | B | 0 |
| 7 |  | A | 0 |
| 8 |  | B | 0 |
| 9 |  | C | 0 |
| 10 |  | D | 0 |
| 11 |  | A | 0 |
| 12 |  | C | 0 |
| 13 |  | A | 0 |
| 14 |  | B | 0 |
| 15 |  | C | 0 |
| 16 |  | A | 0 |
| 17 |  | A | 0 |
| 18 |  | D | 0 |
| 19 |  | D | 0 |
| 20 |  | A | 0 |
| 21 |  | C | 0 |
| 22 |  | B | 0 |
| 23 |  | B | 0 |
| 24 |  | B | 0 |
| 25 |  | D | 0 |
| 26 |  | B | 0 |
| 27 |  | D | 0 |
| 28 |  | C | 0 |
| 29 |  | B | 0 |
| 30 |  | D | 0 |
| 31 |  | A | 0 |
| 32 |  | D | 0 |
| 33 |  | C | 0 |
| 34 |  | C | 0 |
| 35 |  | B | 0 |
| 36 |  | DBXEAYFZC | 0 |
| 37 |  | 单元 | 0 |
| 38 |  | 过程 | 0 |
| 39 |  | 逻辑设计 | 0 |
| 40 |  | 分量 | 0 |
| 41 |  | viod | 0 |
| 42 |  | x2+k-1 | 0 |
| 43 |  | 2,3 | 0 |
| 44 |  | 10pumpkin(s) | 0 |
| 45 |  | n\*Fun(n-1); | 0 |
| 46 |  | this->data=data; | 0 |
| 47 |  | p=new int(\*(s.p)); | 0 |
| 48 |  | operator + | 0 |
| 49 |  | 8100 | 0 |
| 50 |  | 哈克汪汪叫吉米喵喵叫 | 0 |