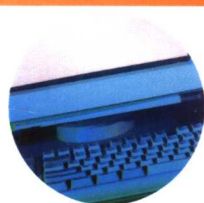
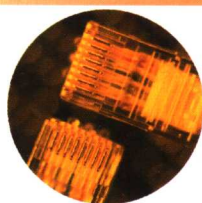


全国计算机等级考试

2006

历届笔试真题详解

二级 C++语言程序设计



全国计算机等级考试命题研究组 编



南开大学出版社

全国计算机等级考试

历届笔试真题详解

二级 C++ 语言程序设计

(2006)

全国计算机等级考试命题研究组 编

南开大学出版社

天 津

内容提要

本书主要内容有：(1) 3套笔试真题试卷和5套笔试全真模拟试卷。笔试真题试卷是从2004年9月到2005年9月的3套，5套全真模拟试卷是根据2004年版《全国计算机等级考试大纲》对最新考试的要求而编制的。(2) 针对8套试卷进行详解：精解考点，分析题眼，详解重点难点，并给出应试技巧。(3) 本书有光盘配套。其中有本考试的全真模拟笔试试卷和上机环境，并配有含大量全真模拟试题的题库，可用于考前实战训练。

本书完全针对准备参加全国计算机等级考试（二级 C++语言程序设计）的考生，同时也可以作为普通高校、大专院校、成人高等教育以及相关培训班的练习题和考试题使用。

图书在版编目(CIP)数据

全国计算机等级考试历届笔试真题详解. 二级 C++语言程序设计 / 全国计算机等级考试命题研究组编.

天津:南开大学出版社, 2006. 3

ISBN 7-310-02497-4

I. 全... II. 全... III. ①电子计算机—水平考试
—解题 ②C 语言—程序设计—水平考试—解题
IV. TP3-44

中国版本图书馆 CIP 数据核字(2005)第 153915 号

版权所有 侵权必究

南开大学出版社出版发行

出版人:肖占鹏

地址:天津市南开区卫津路 94 号 邮政编码:300071

营销部电话:(022)23508339 23500755

营销部传真:(022)23508542 邮购部电话:(022)23502200

*

天津市宝坻区第二印刷厂印刷

全国各地新华书店经销

*

2006 年 3 月第 1 版 2006 年 3 月第 1 次印刷

787×1092 毫米 16 开本 11.5 印张 283 千字

定价:24.00 元

如遇图书印装质量问题,请与本社营销部联系调换,电话:(022)23507125

前言

全国计算机等级考试(National Computer Rank Examination, NCRE)是由教育部考试中心主办、用于考查应试人员的计算机应用知识与能力的考试。本考试的证书已经成为许多单位招聘员工的一个必要条件,具有相当的“含金量”。

为了帮助考生更顺利地通过计算机等级考试,我们做了大量市场调查,根据考生的备考体会,以及培训教师的授课经验,推出了《全国计算机等级考试历届笔试真题详解——二级C++语言程序设计》。本书的主要组成有三部分。

一、历届真题和全真模拟试卷

◆ 3套笔试真题试卷

这3份试卷来自2004年9月到2005年9月的等级考试。对于备战等级考试而言,做真题是进行考前冲刺的最佳方式。这是因为它的针对性相当强,考生可以通过真题的实际练习,来检验自己是否真正掌握了相关知识点,了解考试重点,并且根据需要再对知识结构的薄弱环节进行强化。

◆ 5套笔试全真模拟试卷

根据教育部考试中心2004年版《全国计算机等级考试大纲》,按照最新的考试要求,完全按照真题的出题方式,编写了5套全真模拟试卷。

二、真题详解

在每套试卷的后面,都有针对各个试题的答案和详细分析,精解考点,分析题眼,详解重点难点,并给出应试技巧。

三、笔试和机试全真环境模拟光盘

本书配套光盘包含本考试的全真模拟笔试试卷和上机环境,并配有含全真模拟试题的题库,可用于考前实战训练。

为了保证本书及时面世和内容准确,很多朋友做出了贡献,陈河南、贺民、黄志雄、齐惠颖、贺军、于樊鹏、任世华、田民、许伟、侯佳宜、何雄、赵晓睿、戴文雅、戴军、汤效平、陈占军、李季、梁彩隆、李志云、陈安南、李晓春、王春桥、王雷、韦笑、龚亚萍等老师在编写文档、调试程序、排版、查错、预读、光盘制作与测试等工作中加班加点,付出了很多辛苦,在此一并表示感谢!

在学习的过程中,您如有问题或建议,请与我们联系。电子邮件为book_service@126.com,网址为www.baifen100.com。

全国计算机等级考试命题研究组

2005年12月

目 录

2004 年 9 月二级 C++ 笔试试卷	1
2004 年 9 月二级 C++ 笔试试卷答案和解析	12
2005 年 4 月二级 C++ 笔试试卷	19
2005 年 4 月二级 C++ 笔试试卷答案和解析	31
2005 年 9 月二级 C++ 笔试试卷	38
2005 年 9 月二级 C++ 笔试试卷答案和解析	49
二级 C++ 全真模拟试卷 (一)	56
二级 C++ 全真模拟试卷 (一) 答案和解析	72
二级 C++ 全真模拟试卷 (二)	80
二级 C++ 全真模拟试卷 (二) 答案和解析	95
二级 C++ 全真模拟试卷 (三)	103
二级 C++ 全真模拟试卷 (三) 答案和解析	120
二级 C++ 全真模拟试卷 (四)	128
二级 C++ 全真模拟试卷 (四) 答案和解析	143
二级 C++ 全真模拟试卷 (五)	151
二级 C++ 全真模拟试卷 (五) 答案和解析	167

2004 年 9 月二级 C++ 笔试试卷

(考试时间 90 分钟, 满分 100 分)

一、选择题 ((1) ~ (35) 每小题 2 分, 共 70 分)

下列各题 A)、B)、C)、D) 四个选项中, 只有一个选项是正确的, 请将正确选项涂写在答题卡相应位置上, 答在试卷上不得分。

- (1) 下面叙述正确的是
 - A) 算法的执行效率与数据的存储结构无关
 - B) 算法的空间复杂度是指算法程序中指令 (或语句) 的条数
 - C) 算法的有穷性是指算法必须能在执行有限个步骤之后终止
 - D) 以上三种描述都不对
- (2) 以下数据结构中不属于线性数据结构的是
 - A) 队列
 - B) 线性表
 - C) 二叉树
 - D) 栈
- (3) 在一棵二叉树上第 5 层的结点数最多是
 - A) 8
 - B) 16
 - C) 32
 - D) 15
- (4) 下面描述中, 符合结构化程序设计风格的是
 - A) 使用顺序、选择和重复 (循环) 三种基本控制结构表示程序的控制逻辑
 - B) 模块只有一个入口, 可以有多个出口
 - C) 注重提高程序的执行效率
 - D) 不使用 goto 语句
- (5) 下面概念中, 不属于面向对象方法的是
 - A) 对象
 - B) 继承
 - C) 类
 - D) 过程调用
- (6) 在结构化方法中, 用数据流程图 (DFD) 作为描述工具的软件开发阶段是
 - A) 可行性分析
 - B) 需求分析
 - C) 详细设计
 - D) 程序编码
- (7) 在软件开发中, 下面任务不属于设计阶段的是
 - A) 数据结构设计
 - B) 给出系统模块结构
 - C) 定义模块算法

D) 定义需求并建立系统模型

(8) 数据库系统的核心是

A) 数据模型

B) 数据库管理系统

C) 软件工具

D) 数据库

(9) 下列叙述中正确的是

A) 数据库系统是一个独立的系统，不需要操作系统的支持

B) 数据库设计是指设计数据库管理系统

C) 数据库技术的根本目标是要解决数据共享的问题

D) 数据库系统中，数据的物理结构必须与逻辑结构一致

(10) 下列模式中，能够给出数据库物理存储结构与物理存取方法的是

A) 内模式

B) 外模式

C) 概念模式

D) 逻辑模式

(11) 关于面向对象的程序设计方法, 下列说法正确的是

A) “封装性”指的是将不同类型的相关数据组合在一起，作为一个整体进行处理

B) “多态性”指的是对象的状态会根据运行时要求自动变化

C) 基类的私有成员在派生类的对象中不可访问, 也不占内存空间

D) 在面向对象的程序设计中, 结构化程序设计方法仍有着重要作用

(12) 判断字符型变量 ch 是否为大写英文字母, 应使用表达式

A) `ch>='A' & ch<='Z'`

B) `ch<='A' || ch>='Z'`

C) 'A' <= ch <= 'Z'

D) `ch>='A' && ch<='Z'`

(13) 已知下列语句中的 x 和 y 都是 int 型变量，其中错误的语句是

A) $x=y++;$

B) $x=++y$;

C) $(x+y)++$

D) ++x=y;

(14) 执行语句序列

```
int n;
```

```
cin>>n;
```

switch(n)

```

{   case 1:

```

```
case 2:cout<<'1';
```

case 3:

```
case 4:cout<<'2'; break;
```

```
default:cout<<'3':
```

}

时, 若键盘输入 1, 则屏幕显示

A) 1

B) 2

C) 3

D) 12

(15) 下列程序的输出结果是

```
#include <iostream>
using namespace std;
int main()
{
    char a[] = "Hello,World";
    char*ptr = a;
    while(*ptr)
    {
        if(*ptr>= 'a' &&*ptr <='z')
            cout<<char(*ptr + 'A' - 'a');
        else cout<<*ptr;
        ptr++;
    }
    retur 0;
}
```

A) HELLO, WORLD

C) Hello, World

C) hELLO, wORLD

D) hello, world

(16) 已知: int m=10; 在下列定义引用的语句中, 正确的是

A) int &x=m;

B) int y=&m;

C) int &z;

D) int &t=&m;

(17) 下列函数原型声明中错误的是

A) void Fun(int x=0,int y=0);

B) void Fun(int x, int y);

C) void Fun(int x,int y=0);

D) void Fun(int x=0,int y);

(18) 已知程序中已经定义了函数 test, 其原型是 int test(int,int,int); , 则下列重载形式中正确的是

A) char test(int,int,int);

B) double test(int,int,double);

C) int test(int,int,int=0);

D) float test(int,int,float=3.5F);

(19) 有以下程序

```
#include <iostream>
int i = 0;
void fun()
{
    {
        static int i = 1;
        std::cout<<i++<<'!';
    }
    std::cout<<i<<'!';
}
int main()
{
    fun();fun();
    return 0;
}
```

程序执行后的输出结果是

A) 1,2,1,2,

B) 1,2,2,3,

C) 2,0,3,0,

D) 1,0,2,0,

(20) 已知函数 f 的原型是: void f(int*a,long &b);, 变量 v1、v2;的定义是 int v1;long v2;. 正确的调用语句是

A) f(v1,&v2);

B) f(v1,v2);

C) f(&v1,v2);

D) f(&v1,&v2);

(21) 有以下类定义

```
class MyClass
{
public:
    MyClass() { cout<<1;}
};
```

则执行语句 MyClass a,b[2],*p[2];后, 程序的输出结果是

A) 11

B) 111

C) 1111

D) 11111

(22) 关于友元, 下列说法错误的是

A) 如果类 A 是类 B 的友元, 那么类 B 也是类 A 的友元

B) 如果函数 fun()被说明为类 A 的友元, 那么在 fun()中可以访问类 A 的私有成员

C) 友元关系不能被继承

D) 如果类 A 是类 B 的友元, 那么类 A 的所有成员函数都是类 B 的友元函数

(23) 关于动态存储分配, 下列说法正确的是

- A) new 和 delete 是 C++ 语言中专门用于动态内存分配和释放的函数
- B) 动态分配的内存空间也可以被初始化
- C) 当系统内存不够时, 会自动回收不再使用的内存单元, 因些程序中不必用 delete 释放内存空间
- D) 当动态分配内存失败时, 系统会立刻崩溃, 因此一定要慎用 new

(24) 有以下程序

```
#include <iostream>
using namespace std;
class MyClass
{
public:
    MyClass(int n){number =n; }
    //拷贝构造函数
    MyClass(MyClass& other) {number = other.number;}
    ~MyClass() {}
private:
    int number;
};
MyClass fun(MyClass p)
{
    MyClass temp(p);
    return temp;
}
int main()
{
    MyClass obj1(10),obj2(0);
    MyClass obj3(obj1);
    obj2=fun(obj3);
    return 0;
}
```

程序执行时, MyClass 类的拷贝构造函数被调用的次数是

- A) 5
- B) 4
- C) 3
- D) 2

(25) 在公有派生的情况下, 派生类中定义的成员函数只能访问原基类的

- A) 公有成员和私有成员
- B) 私有成员和保护成员
- C) 公有成员和保护成员

- D) 私有成员、保护成员和公有成员
- (26) 在 C++ 中用来实现运行时多态性的是
- A) 重载函数 B) 析构函数
C) 构造函数 D) 虚函数
- (27) 一个类可以同时继承多个类，称为多继承。下列关于多个继承和虚基类的表述中，错误的是
- A) 每个派生类的构造函数都要为虚基类构造函数提供实参
B) 多继承时有可能出现对基类成员访问的二义性问题
C) 使用虚基类可以解决二义性问题并实现运行时的多态性
D) 建立最派生类对象时，虚基类的构造函数会首先被调用
- (28) 在一个类体的下列声明中，正确的纯虚函数声明是
- A) virtual void vf()=0;
B) void vf(int)=0;
C) virtual int vf(int);
D) virtual void vf(int) {}
- (29) 在下面的运算符重载函数的原型中，错误的是
- A) Volume operator~ (double,double);
B) double Volume::operator~ (double);
C) Volume Volume::operator~ (Volume);
D) Volume operator~ (Volume,Volume);
- (30) 下列是模板声明的开始部分，其中正确的是
- A) template<T>
B) template<class T1,T2>
C) template<class T1,class T2>
D) template<class T1;class T2>
- (31) 执行语句序列
- ```
ofstream outfile("DATA.DAT");
if(...) cout<<"OK";else cout <<"FAIL";
```
- 后，如果文件打开成功显示“OK”，否则就显示“FAIL”。由此可知，上面 if 语句的...处的表达式应是
- A) outfile.fail() 或 outfile  
B) outfile.good() 或 !outfile  
C) outfile.good() 或 outfile  
D) outfile.fail() 或 !outfile
- (32) C++ 流中重载的运算符 >> 是一个
- A) 用于输出操作的非成员函数

- B) 用于输入操作的非成员函数
- C) 用于输出操作的成员函数
- D) 用于输入操作的成员函数

(33) 有以下类定义

```
class Point{
public:
 Point(int x = 0,int y =0) {_x = x;_y = y;}
 void Move(int xOff,int yOff)
 {_x +=xOff;_y+=yOff;}
 void Print() const
 {cout<<('<<_x<<','<<_y<<')<< endl;}
private:
 int _x,_y;
};
```

下列语句中会发生编译错误的是

- A) Point pt;pt.Print();
- B) const Point pt;pt.Print();
- C) Point pt;pt.Move(1, 2);
- D) const Point pt;pt.Move(1, 2);

(34) 有以下类定义

```
class MyClass
{
private:
 int id;
 char gender;
 char*phone;
public:
 MyClass():id(0),gender('#'),phone(NULL){}
 MyClass(int no,char ge= '#', char*ph=NULL)
 {id=no;gender=ge;phone=ph;}
};
```

下列类对象定义语句中错误的是

- A) MyClass myObj;
- B) MyClass myObj(11,"13301111155");
- C) MyClass myObj(12,'m');
- D) MyClass myObj(12);

(35) 有以下程序

```
#include <iostream>
```

```

using namespace std;
class Complex
{
public:
 Complex(double r =0,double i =0):re(r),im(i) {}
 double real() const {return re;}
 double imag()const { return im;}
 Complex operator + (Complex c) const
 {return Complex(re+c.re,im+c.im);}
private:
 double re,im;
};
int main()
{
 Complex a =Complex(1,1) + Complex(5);
 cout<<a.real()<<'+'<<a.imag() <<'i' <<endl;
 return 0;
}

```

程序执行后的输出结果是

- |         |         |
|---------|---------|
| A) 6+6i | B) 6+1i |
| C) 1+6i | D) 1+1i |

## 二、填空题（每空 2 分，共 30 分）

请将每一个空的正确答案写在答题卡【1】~【15】序号的横线上，答在试卷上不得分。

注意：以命令关键字填空的必须拼写完整。

- (1) 算法的复杂度主要包括【1】复杂度和空间复杂度。
- (2) 数据的逻辑结构在计算机存储空间中的存放形式称为数据的【2】。
- (3) 若按功能划分，软件测试的方法通常分为白盒测试方法和【3】测试方法。
- (4) 如果一个工人可管理多个设备，而一个设备只被一个工人管理，则实体“工人”与实体“设备”之间存在【4】关系。
- (5) 关系数据库管理系统能实现的专门关系运算包括选择、连接和【5】。
- (6) 设有定义语句：int a=12;，则表达式 a\*=2+3 的运算结果是【6】。
- (7) 从实现的角度划分，C++所支持的两种多态性分别是【7】时的多态性和运行时的多态性。
- (8) 将一个函数声明为一个类的友元函数必须使用关键字【8】。
- (9) 请按下面注释的提示，将类 B 的构造函数定义补充完整。

```

class A
{
 int a;
public:
 A(int aa=0) {a=aa;}
};
class B:public A {
 int b;
 A c;
public:
 //用 aa 初始化基类 A, 用 aa+1 初始化类对象成员 c
 B(int aa): 【9】 {b=aa+2}
};

```

(10) 下列程序的输出结果是 **【10】**。

```

#include <iostream>
using namespace std;
int main()
{
 int i=5;
 int &r = i;r = 7;
 cout<<i<<<endl;
 return 0;
}

```

(11) 下列程序的输出结果是 **【11】**。

```

#include <iostream>
using namespace std;
class Test {
public:
 Test() { cnt++;}
 ~Test() { cnt--;}
 static int Count() {return cnt;}
private:
 static int cnt;
};
int Test::cnt =0;
int main()
{
 cout<<Test::Count()<<' ';
}

```

```

 Test t1,t2;
 Test*pT3 = new Test;
 Test*pT4 = new Test;
 cout<<Test::Count()<<' ';
 delete pT4;
 delete pT3;
 cout<<Test::Count()<<endl;
 return 0;
}

```

- (12) 下面是用来计算  $n$  的阶乘的递归函数，请将该函数的定义补充完整。(注：阶乘的定义是  $n!=n*(n-1)*...*2*1$ )

```

unsigned fact(unsigned n)
{
 if (n <= 1)
 return 1;
 return 【12】;
}

```

- (13) 下列程序的输出结果是 **【13】**。

```

#include <iostream>
using namespace std;
template<typename T>
T fun(T a,T b) { return (a<=b)?a:b;}
int main()
{
 cout<<fun(3, 6)<<','<<fun(3.14F,6.28F) <<endl;
 return 0;
}

```

- (14) 与成员访问表达式  $p \rightarrow \text{name}$  等价的表达式是 **【14】**。

- (15) 下列程序的输出结果是 **【15】**。

```

#include <iostream>
using namespace std;
class base{
public:
 int n;
 base(int x) { n = x;}
 virtual void set(int m) { n = m;cout << n <<' ';}
};

```

```

class deriveA:public base {
 public:
 deriveA(int x):base(x) {}
 void set(int m) { n += m;cout << n <<' ';}
};
class deriveB:public base{
 public:
 deriveB(int x):base(x) {}
 void set(int m) { n +=m; cout << n<<' ';}
};
int main()
{
 deriveA d1(1);
 deriveB.d2(3);
 base *pbase;
 pbase = &d1;
 pbase->set(1);
 pbase= &d2;
 pbase->set(2);
 return 0;
}

```

## 2004 年 9 月二级 C++ 笔试试卷答案和解析

### 一、选择题

(1) 【答案】C

【解析】A 选项错误，因为算法的执行效率与算法执行过程中所需基本运算的执行次数有关；B 选项错误，原因是算法的空间复杂度是指执行这个算法所需要的内存空间；C 选项正确，故 D 选项不正确。

(2) 【答案】C

【解析】所谓的线性结构是指：如果一个非空的数据结构满足下列两个条件，即①有且只有一个根结点；②每一个结点最多有一个前驱，也最多有一个后继。同时满足两个条件的有队列、线性表和栈，而二叉树的结点可能存在两个后继，所以不是线性结构。

(3) 【答案】B

【解析】根据二叉树的性质，在二叉树的第 K 层上，最多有  $2^{k-1}$  次方个结点。所以，第五层的结点数最多为 16。

(4) 【答案】A

【解析】应该选择只有一个入口和一个出口的模块，故 B 选项错误；首先要保证程序正确，然后才要求提高效率，故 C 选项错误；严格控制使用 goto 语句，必要时可以使用，故 D 选项错误。

(5) 【答案】D

【解析】A、B、C 属于面向对象方法，D 属于面向过程方法，故答案为 D。

(6) 【答案】B

【解析】结构化分析方法是结构化程序设计理论在软件需求分析阶段的运用。而结构化分析就是使用数据流图 (DFD)、数据字典 (DD)、结构化英语、判定表和判定树等工具，来建立一种新的、称为结构化规格说明的目标文档。所以数据流程图是在需求分析阶段使用的。

(7) 【答案】D

【解析】数据结构设计、给出系统模块结构以及定义模块算法都属于设计阶段，而定义需求并建立系统模型属于分析阶段。

(8) 【答案】B

【解析】数据库管理系统是一种系统软件，负责数据库中的数据组织、数据操纵、数据维护、控制及保护和数据服务等，因此数据库管理系统是数据库系统的核心。

(9) 【答案】C