ORACLE数据库基础知识 216 page

中兴笔试题目

公共部分（50分）

　　1：作业调度程序从处于（B）状态的队列中选择适当的作业的作业投入运行。（3分）

　　A 运行 B 提交 C 完成 D后备

　　2：SQL语言中，删除一个表的命令是（B）（3分）

　　A CLEAR TABLE B DROP TABLE

　　C DELETE TABLE D REMOVE TABLE

　　3：ATM采用的线路复用方式为（C）（3分）

　　A 频分多路复用

　　B 同步时分多路复用

　　C 异步时分多路复用

　　D 独占信道

　　4：数据库中只存放视图的（C）（3分）

　　A操作

　　B 对应的数据

　　C 定义

　　D 限制

　　5：什么是虚拟设备？为什么在操作系统中引入虚拟设备？（10分）

SPOOLing是Simultaneous Peripheral Operation On-Line （即外部设备联机并行操作）的缩写，它是关于慢速字符设备如何与计算机主机交换信息的一种技术，通常称为“假脱机技术”。

　　6：TCP为何采用三次握手来建立连接，若采用二次握手可以吗，请说明原因？（16分）

三次握手是为了防止已失效的连接请求再次传送到服务器端。

二次握手不可行，因为：如果由于网络不稳定，虽然客户端以前发送的连接请求以到达服务方，但服务方的同意连接的应答未能到达客户端。则客户方要重新发送连接请求，若采用二次握手，服务方收到重传的请求连接后，会以为是新的请求，就会发送同意连接报文，并新开进程提供服务，这样会造成服务方资源的无谓浪费。

　　7：什么是分布式数据库？（12分）

分布式数据库系统是在集中式数据库系统成熟技术的基础上发展起来的，但不是简单地把集中式数据库分散地实现，它具有自己的性质和特征。集中式数据库系统的许多概念和技术，如数据独立性、数据共享和减少冗余度、并发控制、完整性、安全性和恢复等在分布式数据库系统中都有了不同的、更加丰富的内容。

（1）数据独立性。数据独立性是数据库方法追求的主要目标之一。在集中式数据库中，数据独立性包括两方面：数据的逻辑独立性和物理独立性。其意义在于程序和数据的逻辑结构和数据的存储结构无关。在分布式系统中，数据库独立性除了上面所说之外，还有数据分布独立性亦称分布透明性，即用户不必关心数据的逻辑分片，不必关心数据的物理位置分布的细节，也不必关心重复副本（冗余数据）的一致性问题。有了分布透明性，用户的应用程序书写起来就如同数据没有分布一样。在集中式数据库中，数据的独立性是通过系统的三级模式和它们之间的二级映象得到的。分布式数据库，分布透明性是由于引入新的模式和模式之间的映象得到的。

（2）集中与自治相结合的控制结构。数据库是供用户共享的，在集中式数据库中，为保证数据的安全性和完整性，对数据库的控制是集中的。由数据库管理员（DBA）负责监督和维护系统的正常运行。

在分布式数据库中，数据的共享有两个层次：一是局部共享，即在局部场地上存储局部用户的共享数据。二是全局共享，即在分布式数据库的各个场地也存储可供网络中其他场地的用户共享的数据，支持全局引用。因此，相应的控制结构也具有两个层次：集中和自治。各局部的DBMS可以独立地管理局部数据库，具有自治的功能。同时，系统又设有集中控制机制，协调各局部DBMS的工作，执行全局应用。

（3）适当增加数据冗余度。在集中式数据库中，尽量减少冗余度是系统目标之一。其原因是，冗余数据浪费存储空间，而且容易造成个副本之间的不一致性。减少冗余度的目标是用数据共享来达到的。而在分布式系统中却希望增加冗余数据，在不同的场地存储同一数据的多个副本。其原因是提高系统的可靠性和性能，当某一场地出现故障，系统可以对另一场地上的相同副本进行操作，不会造成系统的瘫痪。系统可以根据距离选择离用户最近的数据副本进行操作，减少通信代价。但是增加冗余会碰到集中式数据库同样的问题，即不利于更新，增加了系统维护代价，需要在这些方面作出权衡。

（4）全局的一致性、可串行性和可恢复性。分布式数据库中各局部数据库应满足集中式数据库的一致性、可串行性和可恢复性。除此以外，还要保证数据库的全局一致性、可串行性和可恢复性。例如，在前面提到的银行转帐事务中，包括两个节点上的更新操作，当其中一个节点出现故障，应使全局事务回滚，在一个节点撤销已经执行的操作等。

　　C++部分（50分）

　　1：设有“int w[3][4]；”，pw是与数组名w等价的数组指针，则pw的初始化语句为int (\*pw)[4] = w;(3分)

　　2：要使引用pr代表变量“char \*p”，则pr的初始化语句为char \* &pr = p;（3分）

　　3：“零值”可以是0，0.0，FALSE或者“空指针”。例如int变量n与“零值”比较的if语句为：if（n==0），则BOOL flag与“零值”比较的if语句为 if(!a) ；float x与“零值”比较的if语句为：const float EPSINON = 0.00001; if ((x >= - EPSINON) && (x <= EPSINON)。（6分）

　　4：设有如下语句：（3分）

　　Int x;

　　(C)

　　\*px=0;

　　则选择以下哪条语句可将x值置为0。

　　A int \*px； B int const \*px=&x；

　　C int \*const px=&x； D const int \*px=&x；

　　5：设void f1（int \*m，long &n）；int a；long b；则以下调用合法的是（B）（3分）

　　A f1（a，b） B f1（&a，b）

　　C f1（a，&b） D f1（&a，&b）

　　6：请写出如下代码的运行结果（6分）6

　　Int main()

　　{int a,b,c,d;

　　a=0;

　　b=1;

　　c=2;

　　d=3;

　　printf(“%d”,a+++b+c+++d++);

}

　　7：写出下面函数的功能（12分）将数组a中的元素(如果为类类型，调用拷贝赋值算符)逆序放置到数组b中

　　Template<class Type>

　　void WE(Type a[],Type b[],int n){

　　for(int i=0;i<n;i++) b[n-i-1]=a[i];}

8写一段代码判断一个单向链表中是否有环。（14分）

　　给出如下结构

　　Struct node

　　{steuct \*next；

　　}；

　　Typedef stuct node Node；

答：

#include <stddef.h>

struct listtype

{

int data;

struct listtype \* next;

};

typedef struct listtype \* list;

/\* Check that whether there is loop in the singly linked list sll or not. \*/

int find\_circle(list sll)

{

list fast = sll;

list slow = sll;

if (NULL == fast)

{

return -1;

}

while (fast && fast->next)

{

fast = fast->next->next;

slow = slow->next;

if (fast == slow)

{

return 1;

}

}

return 0;

}

　　JAVA部分（50分）

　　1：判断下列对的是哪个（B）（3分）

　　A short s1=1;s1=s1+1

　　B short s1=1;s1+=1

　　C short s1=1;s1=s1-1

　　D short s1=1;s1=s1\*1

　　2:main方法是Java Application程序执行的入口点，关于main方法的方法头以下哪项是合法的（C）（3分）

　　A public static void main （）

　　B public static void main （String args）

　　C public static int main （String []arg）

　　D public void main （String args）

　　3：设float x=1，y=2，z=3，则表达式y+=z--/++x的值是（A）（3分）

　　A 3.5 B 3

　　C 4 D 5

　　4：Math.round（11.5）=12

　　Math.round（-11.5）=－11(4分)

　　5：假设x=10，y=20，z=30；计算下列表达式的值（6分）

　　A x<10||x>10\_\_\_\_\_\_\_\_\_false

　　B !(x<=20)\_\_\_\_\_\_\_\_\_\_\_false

　　C z-y==x&&Math.abs(y-z)!=x\_\_\_\_\_\_\_\_\_\_\_\_\_false

　　6：方法重建Overload和方法的重写Overriding的区别。Overload的方法是否可以改变返回值的类型？（10分）

方法的重写Overriding和重载Overloading是Java多态性的不同表现。重写Overriding是父类与子类之间多态性的一种表现，重载Overloading是一个类中多态性的一种表现。如果在子类中定义某方法与其父类有相同的名称和参数，我们说该方法被重写（Overriding）。子类的对象使用这个方法时，将调用子类中的定义，对它而言，父类中的定义如同被“屏蔽”了。如果在一个类中定义了多个同名的方法，它们或有不同的参数个数或有不同的参数类型，则称为方法的重载（Overloading）。 Overloaded的方法是可以改变返回值的类型。

7：谈谈HashMap和Hashtable的区别（6分）

答：1.Hashtable是Dictionary的子类，HashMap是Map接口的一个实现类；

2.Hashtable中的方法是同步的，而HashMap中的方法在缺省情况下是非同步的。即是说，在多线程应用程序中，不用专门的操作就安全地可以使用Hashtable了；而对于HashMap，则需要额外的同步机制。但HashMap的同步问题可通过Collections的一个静态方法得到解决：

Map Collections.synchronizedMap(Map m)

这个方法返回一个同步的Map，这个Map封装了底层的HashMap的所有方法，使得底层的HashMap即使是在多线程的环境中也是安全的。

3. 在HashMap中，null可以作为键，这样的键只有一个；可以有一个或多个键所对应的值为null。当get()方法返回null值时，即可以表示 HashMap中没有该键，也可以表示该键所对应的值为null。

因此，在HashMap中不能由get()方法来判断HashMap中是否存在某个键，而应该用containsKey()方法来判断。

4.其底层的实现机制不同，hashmap的访问速度要快于hashtable，因为它不需要进行同步检验，建议在非多线程环境中使用hashmap代替hashtable .

　　8：构造一个类来描述屏幕上的一个点，该类的构成包括点x和y两个坐标，以及一些对点进行的操作，包括：取得点的坐标值，对点的坐标进行赋值，编写应用程序生成该类的对象并对其进行操作。（15分）

class Point{

private int x;

private int y;

public int getX() {

return x;

}

public int getY() {

return y;

}

public void setX(int x) {

this.x = x;

}

public void setY(int y) {

this.y = y;

}

}

数据结构和算法

1. 假设执行语句S的时间为O(1)，则执行下列程序短的时间为（B）

for(i=1;i<=n;i++)

for(j=i;j<=n;j++)

S;

A. O(n)

B. O(n2)

C. O(n\*i)

D. O(n+1)

2. 二位数组A[10…20，5…10]采用行序为主序方式存储，每个数据元素占4个存储单元，且A[10][5]的存储地址是1000，则A[18][9]的地址是（A）

A. 1208

B. 1212

C. 1368

D. 1364

3. 设栈最大长度为3，入栈序列为1，2，3，4，5，6，则不可能得出栈序列是（D）

A. 1，2，3，4，5，6

B. 2，1，3，4，5，6

C. 3，4，2，1，5，6

D. 4，3，2，1，5，6

4. 设有98个已排序列元素，采用二分法查找时，最大比较次数是（D）

A. 49

B. 15

C. 20

D. 7

5. Hash表示用于数据存储的一种有效的数据结构，Hash表等查找复杂度依赖于Hash值算法的有效性，在最好的情况下，Hash表的查找复杂度为（A）

A. O(1)

B. O(logn)

C. O(n)

D. O(nlogn)

第二部分 软件工程

1. 软件能力成熟度模型CMM共分为（C）成熟度等级

A. 3

B. 4

C. 5

D. 6

2. 按照是否了解软件的内部构造，可以将测试分为黑盒测试和白盒测试。考虑以下算法的PDL语句，如果要对其进行完全路径覆盖的白盒测试，则需要（）条路径。

do ehile records remain

read record;

if record field 1 = 0

then process record;

store in buffer;

increment counter;

elseif record field 2 = 0

then reset counter;

else process record;

store in file;

endif

endif

enddo

A. 4

B. 5

C. 6

D. 7

3. 按照瀑布模型的阶段划分，软件测试可以分为单元测试，集成测试，系统测试。请问以下那项测试不属于系统测试的内容（）

A. 压力测试

B. 接口测试

C. 功能测试

D. 安全测试

E. 性能测试

4. 测试用例的设计是测试的重要阶段。系统测试用例设计应该从什么时候开始（）

A. 需求完成

B. 详细设计完成

C. 编码完成

D. 系统集成完毕，提交系统测试

5. 在UML方法中，使用多种类型的图形来帮助进行设计，请问一下那些图形不是UML的图形类型（B）

A. 类图

B. 实体关系图

C. 序列图

D. 活动图

第三部分 Java语言及其他

1. 下面哪个是short型的取值范围：（C）

A. -27--- 27-1

B. 0 --- 216-1

C. -215--- 215-1

D. -231---231-1

2. 下面哪项是不合法的标识符：（C）

A. $persons

B. TwoUsers

C. \*point

D. \_endline

3. 设 float x = 1,y = 2,z = 3，则表达式 y+=z--/++x 的值是（A）

A. 3.5

B. 3

C. 4

D. 5

4. 下列哪些关键字即能够被用于局部变量的修饰，也可以用做类变量的修饰（）

A. public

B. transient

C. static

D. finally

5. 以下的语句构造了几个JAVA对象？（B）

String S = new String(“aaa”);

A. 1

B. 2

C. 3

6. 下面的哪些叙述为真（C）

A. equals()方法判定引用值是否指向同一对象

B. = = 操作符判定两个不同的对象的内容和类型是否一致

C. equal() 方法只有在两个对象的内容一致时返回ture

D. 类File重写方法equals()在两个不同的对象的内容和类型一致时返回ture

7. 如果一个对象仅仅声明实现了cloneable接口，但是不声明clone方法，外部能够调用其clone方法吗？（A）

A. 能

B. 不能

C. 不确定

8. 考虑在C/S结构下，服务器接受并处理请求，那么关于服务器处理请求的模式，哪些描述是错误的（）

A. 单线程模式下，服务器使用一个线程顺序的处理所有的请求，可能导致阻塞

B. 服务器可以为每一个请求创建一个线程来处理该请求，这样做比单线程模式更加稳定

C. 线程池模式下，当线程使用达到最大数量限制之后，线程池中没用可用线程的时候，服务器将阻塞或者拒绝某个请求的处理

9. 使用Swing做界面的时候，如果把一个JButton放在一个JFrame中，在JFrame改变时只影响JButton的高度而宽度不受影响，应该使用哪个布局管理器？（D）

A. FlowLayout

B. CardLayout

C. North and South of BorderLayout

D. East and West of BorderLayout

E. GridLayout

10. 以下的那个状态是SessionBean所具有，而StatelessBean不具有的（D）

A. 池态

B. 就绪态

C. 不存在态

D. 钝化态

11. 以下关于数据库范式的描述，哪些是错误的（B）

A. 如果把多个数据项用一个大的String表示为一个字段，则不满足第一范式

B. 满足数据库范式使得数据库的效率更高

C. 如果满足更高的范式，则必须首先满足低级别的范式

D. 数据库第二范式要求把表中与所有键没有直接关系的数据全部拆分到其他表中

12. 考虑一下需求：包括学生、课程和教师的信息。其中学生的信息包括学生姓名、年龄地址等；课程信息包括课程号、课程名、课程学分等；教师信息包括教师的姓名、教师的地址等。一个学生可以选修多门课，而每门课也能有多个学生选修；一位教师可以教多门课，每门课也可以由多个教师讲授。请问如果使用关系数据库，并且达到第二范式的要求，需要设计（C）张表

A. 3

B. 4

C. 5

D. 6

13. 为了加快数据库查找的速度，需要对数据表添加索引，请问以下关于索引的描述，哪些是错误的（D）

A. 聚类索引中，表中行的物理次序与索引键值的逻辑顺序相同

B. 使用索引会使得数据更新的效率降低

C. 在大多数数据库系统中，每张表只能够有一个聚类索引

D. 考虑这个SQL语句：Select S.name，S.age from Student S where S.address=”成都”，如果我们对表Student建一个复合索引(age，address)，可以改善上述查询的效率。

第四部分 问答题

1. 简述一个Linux驱动程序的主要流程与功能。

2. 请列举一个软件中时间换空间或者空间换时间的例子。

数组成倍增长（空间换时间）。

3. 简述进程与线程的区别。

多线程共存于应用程序中是现代操作系统中的基本特征和重要标志。用过UNIX操作系统的读者知道进程，在UNIX操作系统中，每个应用程序的执行都在操作系统内核中登记一个进程标志，操作系统根据分配的标志对应用程序的执行进行调度和系统资源分配，但进程和线程有什么区别呢？

进程和线程都是由操作系统所体会的程序运行的基本单元，系统利用该基本单元实现系统对应用的并发性。进程和线程的区别在于：

线程的划分尺度小于进程，使得多线程程序的并发性搞。

另外，进程在执行过程中拥有独立的内存单元，而多个线程共享内存，从而极大地提高了程序的运行效率。

线程在执行过程中与进程还是有区别的。每个独立的线程有一个程序运行的入口、顺序执行序列和程序的出口。但是线程不能够独立执行，必须依存在应用程序中，由应用程序提供多个线程执行控制。

从逻辑角度来看，多线程的意义在于一个应用程序中，有多个执行部分可以同时执行。但操作系统并没有将多个线程看做多个独立的应用，来实现进程的调度和管理以及资源分配。这就是进程和线程的重要区别。

进程（Process）是最初定义在Unix等多用户、多任务操作系统环境下用于表示应用程序在内存环境中基本执行单元的概念。以Unix操作系统为例，进程是Unix操作系统环境中的基本成分、是系统资源分配的基本单位。Unix操作系统中完成的几乎所有用户管理和资源分配等工作都是通过操作系统对应用程序进程的控制来实现的。

C、C++、Java等语言编写的源程序经相应的编译器编译成可执行文件后，提交给计算机处理器运行。这时，处在可执行状态中的应用程序称为进程。从用户角度来看，进程是应用程序的一个执行过程。从操作系统核心角度来看，进程代表的是操作系统分配的内存、CPU 时间片等资源的基本单位，是为正在运行的程序提供的运行环境。进程与应用程序的区别在于应用程序作为一个静态文件存储在计算机系统的硬盘等存储空间中，而进程则是处于动态条件下由操作系统维护的系统资源管理实体。多任务环境下应用程序进程的主要特点包括：

●进程在执行过程中有内存单元的初始入口点，并且进程存活过程中始终拥有独立的内存地址空间；

●进程的生存期状态包括创建、就绪、运行、阻塞和死亡等类型；

●从应用程序进程在执行过程中向CPU发出的运行指令形式不同，可以将进程的状态分为用户态和核心态。处于用户态下的进程执行的是应用程序指令、处于核心态下的应用程序进程执行的是操作系统指令。

在Unix操作系统启动过程中，系统自动创建swapper、init等系统进程，用于管理内存资源以及对用户进程进行调度等。在Unix环境下无论是由操作系统创建的进程还要由应用程序执行创建的进程，均拥有唯一的进程标识（PID）。

4. static有什么用途？（请至少说明两种）

static关键字是C, C++中都存在的关键字, 它主要有三种使用方式, 其中前两种只指在C语言中使用, 第三种在C++中使用(C,C++中具体细微操作不尽相同, 本文以C++为准). from http://topic.csdn.net/t/20060414/11/4686455.html

(1)局部静态变量

(2)外部静态变量/函数

(3)静态数据成员/成员函数

5. 头文件中的ifndef/define/endif做什么用？条件编译

6. 请问一下程序将输出什么结果？乱码

char \*RetMenory(void)

{

char p[] = “hellow world”;

return p;

}

void Test(void)

{

char \*str = NULL;

str = RetMemory();

printf(str);

}

7. 引用与指针有什么区别？

8. 描述实时系统的基本特性

9. 全局变量和局部变量在内存中是否有区别？如果有，是什么区别？

10. 什么是平衡二叉树？

11. 堆栈溢出一般是由什么原因导致的？

递归调用深度过深（栈）

12. 什么函数不能声明为虚函数？

一个类中将所有的成员函数都尽可能地设置为虚函数总是有益的。

设置虚函数须注意：

1：只有类的成员函数才能说明为虚函数；

2：静态成员函数不能是虚函数；

3：内联函数不能为虚函数；

4：构造函数不能是虚函数；

5：析构函数可以是虚函数，而且通常声明为虚函数。

13. 冒泡排序算法的时间复杂度是什么？

n\*n

14. #include <filename.h> 和 #include “filename.h” 有什么区别？

用 #include <filename.h> 格式来引用标准库的头文件（编译器将从标准库目录开始搜索）。

用 #include “filename.h” 格式来引用非标准库的头文件（编译器将从用户的工作目录开始搜索）。

15. 写出float x 与“零值”比较的if语句。

const float EPSINON = 0.00001;

if ((x >= - EPSINON) && (x <= EPSINON)

16. 操作系统中进程调度策略有哪几种？

先来先服务 短作业优先 高优先权优先 时间片轮转

17. Internet采用哪种网络协议？该协议的主要层次结构？

TCP/IP

18. Internet物理地址和IP地址转换采用什么协议？

ARP RARP

19. IP地址的编码分为哪俩部分？

网络号 主机号

2002年度软件开发人员招聘笔试试题

应聘者资料:

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 姓名 |  | 性别 | |  | |
| 出生日期 |  | 出生地 | |  | |
| 地址 |  | | | 邮政编码 |  |
| 电话 |  | Email | |  | |
| 毕业院校 |  | | | 毕业时间 |  |
| 专业 |  | | | | |
| 学历 |  | | 学位 |  | |
| 英语水平（资格） |  | | 计算机水平（资格） |  | |
| 要求职位和希望担任的角色 |  | | | | |
| 待遇要求 |  | | | | |
| 简历（如已提供可不填） | | | | | |
|  | | | | | |
| 其他特长（如已提供可不填） | | | | | |
|  | | | | | |
| 评估意见（由招聘小组填写） | | | | | |
|  | | | | | |

**试题一：基础知识**

1、从供选择的答案中，选出应填入下面叙述中\_?\_内的最确切的解答，把相应编号写在答卷的对应栏内。

假设某计算机具有1M 字节的内存（目前使用的计算机往往具有64M字节以上的内存），并按字节编址，为了能存取该内存各地址的内容，其地址寄存器至少需要二进制\_A\_位。为使4字节组成的字能从存储器中一次读出，要求存放在存储器中的字边界对齐，一个字的地址码应\_B\_。若存储周期为200NS，且每个周期可访问4个字节，则该存储器带宽为\_C\_BIT/S。假如程序员可用的存储空间为4M字节，则程序员所用的地址为\_D\_，而真正访问内存的地址称为\_E\_。

供选择的答案：

A： ①10 ②16 ③20 ④32

B： ①最低两位为00 ②最低两位为10 ③最高两位为00 ④最高两位为10

C： ①20M ②40M ③80M ④160M

D： ①有效地址 ②程序地址 ③逻辑地址 ④物理地址

E： ①指令 ②物理地址 ③内存地址 ④数据地址

2、从供选择的答案中。选出应填入下面叙述中＿?＿内的最确切的解答，把相应编号写在答卷的对应栏内。

给定结点的关键字序列（Ｆ、Ｂ、Ｊ、Ｇ、Ｅ、Ａ、Ｉ、Ｄ、Ｃ、Ｈ），对它按字母的字典顺序进行排列，采用不同方法，其最终结果相同。但中间结果是不同的。

Shell排序的第一趟扫描（步长为５）结果应为＿Ａ＿。

冒泡排序（大数下沉）的第一趟起泡的效果是＿B＿3．

快速排序的第一趟结果是＿Ｃ＿。

二路归并排序的第一趟结局是 ＿D＿。

供选择的答案

Ａ：①（B、F、G、J、A、D、I、E、H、C）

②（B、F、G、J、A、E、D、I、C、H）

③（A、B、D、C、E、F、I、J、G、H）

④（C、B、D、A、E、F、I、G、J、H）

Ｂ：①（A、B、D、C、F、E、I、J、H、G）

②（A、B、D、C、E、F、I、H、G、J）

③（B、F、G、E、A、I、D、C、H、J）

④（B、F、G、J、A、E、D、I、C、H）

Ｃ：①（C、B、D、A、F、E、I、J、G、H）

②（C、B、D、A、E、F、I、G、J、H）

③（B、A、D、E、F、G、I、J、H、C）

④（B、C、D、A、E、F、I、J、G、H）

Ｄ：①（B、F、G、J、A、E、D、I、G、H）

②（B、A、D、E、F、G、I、J、H、C）

③（A、B、D、C、E、F、I、J、G、H）

④（A、B、D、C、F、E、J、I、H、C）

3、从供选择的答案中，选出应填入下面叙述中＿？＿内的最确切的解答．把相应编号写在答卷的对应栏内。

进程是操作系统中的一个重要概念。进程是一个具有一定独立功能的程序在某个数据集合上的一次＿A2＿。

进程是一个＿B3＿的概念，而程序是一个＿C3＿的概念。

进程的最基本状态有＿D4＿。在一个单处理机中，若有6个用户进程，在非管态的某一时刻，处于就绪状态的用户进程最多有＿E5＿个。

供选择的答案

A：①单独操作 ②关联操作 ③运行活动 ④并发活动

B：①静态 ②动态 ③逻辑 ④物理

C：①物理 ②逻辑 ③动态 ④静态

D：①就绪、运行、隐蔽 ②停止、就绪、运行

③运行、就绪、阻塞 ④就绪、撤消、运行

E：①5 ②6 ③1 ④4

4、软件设计中划分模块的一个准则是＿A2＿。两个模块之间的耦合方式中，＿Ｂ3＿耦合的耦合度最高，＿Ｃ4＿耦合的耦合度最低。一个模块内部的内聚种类中＿Ｄ4＿内聚的内聚度最高，＿E1＿内聚的内聚度最低。

供选择的答案

A：①低内聚低耦合②低内聚高耦合③高内聚低耦合④高内聚高耦合

B：①数据 ②非直接 ③控制 ④内容

C：①数据 ②非直接 ③控制 ④内容

D：①偶然 ②逻辑 ③功能 ④过程

E：①偶然 ②逻辑 ③功能 ④过程

5、从供选择的答案中选出应填入下面叙述中＿？＿内的最确切的解答，把相应编号写在答卷的对应栏内。

最常用的一种基本数据模型是关系数据模型，它用统一的＿A 1＿结构来表示实体及实体之间的联系。关系数据库的数据操作语言（ＤＭＬ）主要包括＿Ｂ2＿两类操作。

关系运算以关系代数为理论基础，关系代数的最基本操作是并、差、笛卡尔积、＿Ｃ4＿。用Ｒ∣×∣Ｓ 表示关系 Ｒ和关系 Ｓ的＿Ｄ1＿。

设关系Ｒ和关系Ｓ图示如下：

Ｒ： A B C S: B C D T: A B C D

则关系Ｔ是关系Ｒ和关系Ｓ＿E ＿的结果。

供选择的答案

A：①树 ②网络 ③图 ④二维表

B：①插入和删除 ②检索和更新 ③查询和编辑 ④统计和修改

C：①投影、联接 ②联接、选择 ③选择、投影 ④交、选择

D：①联接 ②笛卡尔积 ③日联接 ④自然联接

E:①自然联接 ②θ联接 ③笛卡尔积 ④并

**试题二：程序设计**

用你所熟悉的任意一种程序语言，编写一个完整的过程，将一个字符串插入到另一个字符串的某个位置后面（例如：将“abc”插入到“abcdef”的第三个字符位置后面，结果为“abcabcdef”）。编写程序时，请在必要的地方加以注释(注：不能用该程序语言的内置函数或过程)。

用你所熟悉的任意一种程序语言，编写一个完整的过程，完成从一个给定的完整的文件路径（如“C:\My Documents\Software Test 1.00.doc”)中，析取文件名，扩展名和文件所处目录的功能，编写程序时，请在必要的地方加以注释(注：不能用该程序语言的内置函数或过程)。

**试题三：数据库设计**

下面的表有问题吗？如果有，如何改进？

|  |  |  |
| --- | --- | --- |
| 城市\* | 街道\* | 邮政编码 |
| 佛山 | 金鱼街 | 528000 |
| 佛山 | 大福路 | 528000 |
| 佛山 | 季华路 | 528000 |
| 广州 | 北京路 | 510000 |
| 广州 | 三元里 | 510000 |
| 广州 | 中山路 | 510000 |
| … | … | … |

\*城市和街道联合做主键

分成二个表,第一个表有以下字段:城市和街道,以城市作为主键;第二个表:街道和邮政编码,以街道为主键.通过街道与第一个表的街道相关联

2、假设有以下的两个表：

Cus\_A

|  |  |  |
| --- | --- | --- |
| ID\* | Name | Address |
| … | … | … |

Cus\_B

|  |  |  |
| --- | --- | --- |
| ID\* | Name | Address |
| … | … | … |

\*主键

表Cus\_A和表Cus\_B的结构完全相同，表Cus\_A和表Cus\_B中既存在ID相同的记录，也存在ID不同的记录。现要求将ID只存在于表Cus\_A中而不存在于表Cus\_B中的记录全部插入到Cus\_B表中，并用表Cus\_A中的记录更新表Cus\_B中相同的ID的记录，请写出完成这一功能的存储过程。

Create or procedure test is

Declare id1 CusA.id%type;

Begin

Select id into id1 from CusA a,CusB b where a.id=b.id;

3、某公司正在开发一个档案管理系统，要求在关系数据库中实现和Windows文件系统完全一致的树状文件目录。为了实现这一目录结构，至少需求哪些表？请详细描述这些表的用途和结构（如有必要，可用图表进行描述）。最后，请用伪编码（或自然语言）描述按树状结构遍历所有档案的算法。

**试题四：系统设计**

简述“面向对象”的概念，并设计一个学校选课系统的对象模型。

简述“软件系统的两层和多层体系结构”的概念（如有必要，可用图表进行描述），并指出两者的主要区别。

3、简述一个软件工程的主要步骤，并指出每个步骤的明确目标及实施办法。最后分析哪些是关键步骤。

**试题五：综合**

你认为在团队开发中，你能发挥自己的能力吗？为什么？当个人意见与主管出现矛盾时，你用什么方法解决？

一.选择题

1.一株查找二叉树,其结点A、B、C、D、E、F依次存放在一个起始地址为n(假定地址以字节为单位顺序编号)的连续区域中,每个节点占4个字节:前两个字节存放结点值,后两个字节依次放左指针、右指针.

若该查找二叉树的根结点为E,则它的一种可能的前序遍历为\_\_\_\_ ,相应的层次遍历为\_\_\_\_.

在以上两种遍历情况下,结点C的左指针LC的存放地址为\_\_\_\_\_ ,LC的内容为\_\_\_\_\_\_ 结点A的左指针RA的内容为\_\_\_\_\_\_\_.

供选择的答案

(1) A. EAFCBD B.EFACDB C.EABCFD D.EACBDF

(2) A. EAFCBD B.EFACDB C.EABCFD D.EACBDF

(3) A.n+4 B.n+10 C.n+12 D.n+13

(4) A.n+9 B.n+8 C.n+12 D.n+13

(5) A.n+4 B.n+8 C.n+12 D.n+16

2.虚存页面调整算法有多种,\_\_\_\_\_\_ 调度算法不是页面调度算法.

供选择的答案

A.后进先出 B.先进先出 C.最近最少使用 D.随机选择

3.在软件开发过程中常用图作为描述工具.如DFD就是面向\_\_\_\_\_\_\_分析方法的描述工具.在一套分层DFD中,如果某一张图中有N个加工(Process),则这张图允许有\_\_\_\_\_ 张子图.在一张DFD图中,任意两个加工之间\_\_\_\_\_ .在画分层DFD时,应保持\_\_\_\_\_ 之间的平衡.DFD中从系统的输出流到系统的输出流的一连串连续变换形成一种信息流,这种信息可分为\_\_\_\_\_两类.

A.(1)数据结构 (2)数据流 (3)对象 (4)构件

B.(1)0 (2)1 (3)1-N (4)0-N

C.(1)有且仅有一条数据流

(2)至少有一条数据流

(3)可以有0条或多条名字互不相同的数据流

(4)可以有0或多条数据流,但允许其中存若干条名字相同的数据流.

D.(1)父图与其子图 (2)同一父图的所有子图 (3)不同父图的所有子图 (4)同一子图的所有直接父图.

E.(1)控制流和变换流

(2)变换流和事务流

(3)事务流和事件流

(4)事件流和控制流

4.用二进制加法器对二一十进制编码的十进制数求和,当和的本位十进制数二一十进制编码小于等于1001且向高位无进位时,\_\_\_\_\_ ;当和小于等于1001且向高位存进位时,\_\_\_\_\_;当和大于1001时,\_\_\_\_\_

(1)-(3) A:不需进行修改

B:需进行加6修改

C:需进行减6修改

D:进行加6或减6修改,需进一步判别.

5.www页面访问的大致过程如下:

用户在浏览器中输入要访问的WWW页面的\_\_\_\_地址(<http://hostname/directory/file>);浏览器通过\_\_\_\_ 查询上述输入信息所指的WEB服务器的IP地址;浏览器通过网络与该IP地址处的WEB服务器的\_\_\_\_\_\_服务端之间建立一条\_\_\_\_\_\_连接;浏览器依照相关协议发送\_\_\_\_\_命令;WEB服务器根据命令取出文档,发送回来;浏览器释放连接,显示该文档.

(1) A.URL B.EMS C.NDS D.DNS

(2)A.NAT B.EMS C.NDS D.DNS

(3)A.HTML B.HTTP C.SMTP D.SNMP

(4)A.RTP B.IP C.TCP D.UDP

(5)A.TCP B.GET C.UDP D.PUT

6.假设某计算机具有1MB的内存(目前使用的计算机往往具有64MB以上内存),并按字节编址,为了能存取该内存各地址的内容,其地址寄存器至少需要二进制\_\_\_\_位.为使4字节组成的字段从存储器中一次读出,要求存放存储器中的字边界对齐,一个字节的地址码应\_\_\_\_\_若存储器周期为200ns,且每个周期可访问4个字节,则该存储器带宽为\_\_\_\_\_bit/s假如程序员所用的地址为\_\_\_\_\_\_,而真正访问内存的地址称为\_\_\_\_\_\_\_

A.(1)10 (2)16 (3) 20 (4)32

B.(1)最低两位00 (2)最低两位为10 (3)最高两位为00 (4)最高两位为10

C.(1)20M (2)40M (3)80M (4)160M

D.(1)有效地址 (2)程序地址 (3)逻辑地址 (4)物理地址

E.(1)指令地址 (2)物理地址 (3)内存地址 (4)数据地址

7.英语题

Soon,more of the information we receive via the internet could come \_\_\_\_\_in digital wrappers.

Wrappers are made up \_\_\_\_\_\_ softwore code that’s targeted to do specific things with the data \_\_\_\_\_within them such as helping to define queries for search engines They also keep \_\_\_\_\_from\_\_\_\_\_access to that code.

(1) A.Package B.packaged C.packages D.packaging

(2)A.of B.off C.on D.out

(3)A.close B.closed C.enclose D.enclosed

(4)A.insiders B.money C.outsiders D.warehouse

(5)A.gain B.gained C.gains D.gaining

二.设计题

1.在VC中怎样获得父窗口的指针(写出代码)

2.怎样创建一个临时文件

3.怎样获得状态栏和工具栏的指针.

4.访问控件存几种方法

三.填空题:

1.ODBC的数据类型分为\_\_\_\_\_\_\_\_\_和\_\_\_\_\_\_\_\_\_

2.VC访问数据库的方式\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

3.VC的线路分为\_\_\_\_\_\_\_\_\_和\_\_\_\_\_\_\_\_\_,它是用什么对象表示的\_\_\_\_\_\_\_\_\_

4.下列中a的值是\_\_\_\_\_\_\_\_\_

#define AAA 200

#define BBB AAA+100

int a= BBB\*2

java华为面试题

JAVA方面

1 面向对象的特征有哪些方面

2 String是最基本的数据类型吗?

3 int 和 Integer 有什么区别

4 String 和StringBuffer的区别

5 运行时异常与一般异常有何异同？

异常表示程序运行过程中可能出现的非正常状态，运行时异常表示虚拟机的通常操作中可能遇到的异常，是一种常见运行错误。java编译器要求方法必须声明抛出可能发生的非运行时异常，但是并不要求必须声明抛出未被捕获的运行时异常。

6 说出一些常用的类，包,接口，请各举5个

7 说出ArrayList,Vector, LinkedList的存储性能和特性

ArrayList和Vector都是使用数组方式存储数据，此数组元素数大于实际存储的数据以便增加和插入元素，它们都允许直接按序号索引元素，但是插入元素要涉及数组元素移动等内存操作，所以索引数据快而插入数据慢，Vector由于使用了synchronized方法（线程安全），通常性能上较ArrayList差，而LinkedList使用双向链表实现存储，按序号索引数据需要进行前向或后向遍历，但是插入数据时只需要记录本项的前后项即可，所以插入速度较快。

8设计4个线程，其中两个线程每次对j增加1，另外两个线程对j每次减少1。写出程序。

以下程序使用内部类实现线程，对j增减的时候没有考虑顺序问题。

public class ThreadTest1{

private int j;

public static void main(String args[]){

ThreadTest1 tt=new ThreadTest1();

Inc inc=tt.new Inc();

Dec dec=tt.new Dec();

for(int i=0;i<2;i++){

Thread t=new Thread(inc);

t.start();

t=new Thread(dec);

t.start();

}

}

private synchronized void inc(){

j++;

System.out.println(Thread.currentThread().getName()+"-inc:"+j);

}

private synchronized void dec(){

j--;

System.out.println(Thread.currentThread().getName()+"-dec:"+j);

}

class Inc implements Runnable{

public void run(){

for(int i=0;i<100;i++){

inc();

}

}

}

class Dec implements Runnable{

public void run(){

for(int i=0;i<100;i++){

dec();

}

}

}

}

9.JSP的内置对象及方法。

request request表示HttpServletRequest对象。它包含了有关浏览器请求的信息，并且提供了几个用于获取cookie, header, 和session数据的有用的方法。

response response表示HttpServletResponse对象，并提供了几个用于设置送回 浏览器的响应的方法（如cookies,头信息等）

out out 对象是javax.jsp.JspWriter的一个实例，并提供了几个方法使你能用于向浏览器回送输出结果。

pageContext pageContext表示一个javax.servlet.jsp.PageContext对象。它是用于方便存取各种范围的名字空间、servlet相关的对象的API，并且包装了通用的servlet相关功能的方法。

session session表示一个请求的javax.servlet.http.HttpSession对象。Session可以存贮用户的状态信息

application applicaton 表示一个javax.servle.ServletContext对象。这有助于查找有关servlet引擎和servlet环境的信息

config config表示一个javax.servlet.ServletConfig对象。该对象用于存取servlet实例的初始化参数。

page page表示从该页面产生的一个servlet实例

10.用socket通讯写出客户端和服务器端的通讯，要求客户发送数据后能够回显相同的数据。

参见课程中socket通讯例子。

11说出Servlet的生命周期，并说出Servlet和CGI的区别。

Servlet被服务器实例化后，容器运行其init方法，请求到达时运行其service方法，service方法自动派遣运行与请求对应的doXXX方法（doGet，doPost）等，当服务器决定将实例销毁的时候调用其destroy方法。

与cgi的区别在于servlet处于服务器进程中，它通过多线程方式运行其service方法，一个实例可以服务于多个请求，并且其实例一般不会销毁，而CGI对每个请求都产生新的进程，服务完成后就销毁，所以效率上低于servlet。

12.EJB是基于哪些技术实现的?并说出SessionBean和EntityBean的区别，StatefulBean和StatelessBean的区别。

13．EJB包括（SessionBean,EntityBean）说出他们的生命周期，及如何管理事务的？

14．说出数据连接池的工作机制是什么?

15.同步和异步有和异同，在什么情况下分别使用他们？举例说明。

16.应用服务器有那些？

17你所知道的集合类都有哪些？主要方法？

18给你一个:驱动程序A,数据源名称为B,用户名称为C,密码为D,数据库表为T，请用JDBC检索出表T的所有数据。

19．说出在JSP页面里是怎么分页的?

页面需要保存以下参数：

总行数：根据sql语句得到总行数

每页显示行数：设定值

当前页数：请求参数

页面根据当前页数和每页行数计算出当前页第一行行数，定位结果集到此行，对结果集取出每页显示行数的行即可。

－－－－－－－－－－－－－－－－－－

数据库方面：

1.存储过程和函数的区别

存储过程是用户定义的一系列sql语句的集合，涉及特定表或其它对象的任务，用户可以调用存储过程，而函数通常是数据库已定义的方法，它接收参数并返回某种类型的值并且不涉及特定用户表。

2.事务是什么？

事务是作为一个逻辑单元执行的一系列操作，一个逻辑工作单元必须有四个属性，称为 ACID（原子性、一致性、隔离性和持久性）属性，只有这样才能成为一个事务：

原子性

事务必须是原子工作单元；对于其数据修改，要么全都执行，要么全都不执行。

一致性

事务在完成时，必须使所有的数据都保持一致状态。在相关数据库中，所有规则都必须应用于事务的修改，以保持所有数据的完整性。事务结束时，所有的内部数据结构（如 B 树索引或双向链表）都必须是正确的。

隔离性

由并发事务所作的修改必须与任何其它并发事务所作的修改隔离。事务查看数据时数据所处的状态，要么是另一并发事务修改它之前的状态，要么是另一事务修改它之后的状态，事务不会查看中间状态的数据。这称为可串行性，因为它能够重新装载起始数据，并且重播一系列事务，以使数据结束时的状态与原始事务执行的状态相同。

持久性

事务完成之后，它对于系统的影响是永久性的。该修改即使出现系统故障也将一直保持。

3.游标的作用？如何知道游标已经到了最后？

游标用于定位结果集的行，通过判断全局变量@@FETCH\_STATUS可以判断是否到了最后，通常此变量不等于0表示出错或到了最后。

4.触发器分为事前触发和事后触发，这两种触发有和区别。语句级触发和行级触发有何区别。

事前触发器运行于触发事件发生之前，而事后触发器运行于触发事件发生之后。通常事前触发器可以获取事件之前和新的字段值。

语句级触发器可以在语句执行前或后执行，而行级触发在触发器所影响的每一行触发一次。

=================================

综合面试

Q1：请你分别划划OSI的七层网络结构图，和TCP/IP的五层结构图？

　　Q2：请你详细的解释一下IP协议的定义，在哪个层上面，主要有什么作用？ TCP与UDP呢？

　　Q3：请问交换机和路由器分别的实现原理是什么？分别在哪个层次上面实现的？

　　Q4:请问C++的类和C里面的struct有什么区别？

　　Q5:请讲一讲析构函数和虚函数的用法和作用？

　　Q6:全局变量和局部变量有什么区别？实怎么实现的？操作系统和编译器是怎么知道的

　　Q7:一些寄存器的题目，主要是寻址和内存管理等一些知识。

　　Q8:8086是多少尉的系统？在数据总线上是怎么实现的？

-----------------------------------------------------

网络工程师：

1 H.323协商。（笔试题）

2 ipsec为什么是三层的。l2tp为什么是二层的？

3 ospf中包的ttl值是多少？

4 OSPF为什么要划分区域？

5 MPLS VPN的标签一共有几层。内网的标签放在哪里。

6 MPLS中RD和RT的作用分别是什么？

7 RR防止环路的机制。

8 BGP控制out-bound用local-pre，控制进来的用med.（笔试题）

9 ospf是工作在哪个协议上的？

10 ospf的LSA类型。

11 简述OSPF的基本工作机制。

12 ppp的lcp和ncp协商过程。

13 笔试中还有一道PSTN的信令控制有哪三种？（笔试题）

14sloari 8.0查看进程的命令是什么？linux 7.3查看IP的命令是什么？（笔试题）

15 IP是5.32.0.0,掩码255.224.0.0。请问最大的有效地址是多少。（笔试题）

16 下列哪一项不属于于7号信令标准？

17 lx/???的有效距离是多少？

18 IP 包头几个字节？加上数据部分几个字节

19 QOS有一点点。

20 CQ能不能有一种流量dominate第二种流量? （笔试题）

21 FTP下载一个文件完成。有几个TCP连接？？（笔试题）

星期三 2005年9月7日　denny

c语言常见笔试题总结

【1 使用宏】

1.1

#ifdef NDEBUG

#define TRACE(S) S

#else

#define TRACE(S) printf("%s;\n", #S); S

#endif

问：以上TRACE()宏的作用是什么？

1.2 #error的作用？

1.3 定义一个宏，求出给定数组中的元素的个数

#define NELEMENTS(array) ??

1.4 定义一个宏，求出给定结构中给定成员的偏移量

#define OFFSET(structure, member) ??

【2 数据声明和定义】

给定以下类型的变量a的定义式：

a) An integer

b) A pointer to an integer

c) A pointer to a pointer to an integer

d) An array of 10 integers

e) An array of 10 pointers to integers

f) A pointer to an array of 10 integers

g) A pointer to a <I>function</I> that takes an integer as an argument and returns an integer

h) An array of ten pointers to <I>function</I>s that take an integer argument and return an integer

【3 复杂类型（1）】

有如下表达式：

char (\*(\*x())[])();

请用文字描述x是什么。

【4 复杂类型（2）】

jmp\_buf的定义：

typedef struct \_jmp\_buf

{

REG\_SET reg;

int extra[3];

} jmp\_buf[1];

setjmp函数的原型：

extern int setjmp (jmp\_buf \_\_env);

问：调用setjmp时传递\_\_env的内容，还是传递指针？

【5 头文件】

问：为什么标准头文件都有类似以下的结构？

#ifndef \_\_INCvxWorksh

#define \_\_INCvxWorksh

#ifdef \_\_cplusplus

extern "C" {

#endif

/\*...\*/

#ifdef \_\_cplusplus

}

#endif

#endif /\* \_\_INCvxWorksh \*/

【6 static关键字】

请说出static关键字的3种用处：

（1）用于全局变量；

（2）用于局部变量；

（3）用于函数。

/\* file.c \*/

static int a;

int b;

static int fn()

{

static int x;

int y;

}

【7 const关键字】

7.1 const关键字的意义是什么？

7.2 解释以下的变量定义：

const int a1;

int const a2;

const int \*a3;

int \* const a4;

int const \* const a5;

【8 volatile关键字】

8.1 volatile意义？例如

volatile int \*p;

8.2 volatile能和const一起使用吗？例如

volatile const int \*p;

【9 sizeof()】

有以下定义：

char \*pmsg = "A";

char msg[] = "A";

char ch = 'A';

问：

sizeof(pmsg) = ?

sizeof(msg) = ?

sizeof(“A”) = ?

sizeof(ch) = ?

sizeof(‘A’) = ? （在C++中等于多少？）

void f(char param[100])

{

// sizeof(param) = ?

}

【10 字符串】

有以下代码

char \*pmsg = "hello, world!";

strcpy(pmsg, "hi, there.");

试评论该代码。

【11 混合运算】

有以下代码：

void foo()

{

unsigned int a = 6;

int b = -20;

(a+b > 6) ? puts("> 6") : puts(" < = 6");

}

请问调用foo()的输出？

【12 内存访问】

有以下代码：

void fn()

{

int a[100];

int \*p;

p = (int \*)((unsigned int)a + 1);

printf(“p=0x%x\n”, \*p);

}

试评论以上代码。

【13 C库函数】

请说明以下函数的意义：

void perror(const char \*\_\_s);

fdprintf(int, const char \*, ...);

isspace(), isxdigit(), strerr(), sprintf()

coon @ 23:44:01 | 阅读全文 | 评论 0 | 引用 0 | 编辑

c语言笔试题(九)

2006-09-06

Tag: C语言

1．

#include "stdio.h"

int main()

{

int a;

int \*p;

p = &a;

\*p = 0x500;

a = (int )(\*(&p));

a = (int )(&(\*p));

if(a == (int)p)

printf("equal !\n");

else

printf("not equal !\n");

}

请问本程序的输出显示是什么？

答案：输出显示为”equal!”

2．struct {

signed int bit0:1;

signed int bit1:1;

signed int bit2:1;

signed int bit3:1;

signed int bit4:1;

signed int bit5:1;

signed int bit6:1;

signed int bit7:1;

}bits;

请问sizeof(bits)是否是正确的表达式？

请问语句bits mybits; 的定义是否正确？如果不正确，要如何修改上述的结构定义才能使该语句正确？修改后的结构定义是否会影响sizeof(bits)的正确性？如果正确则该表达式的值为多少？如果将上述的结构中int类型改为char类型，此时sizeof(bits)的大小为多少？

答案：1）是正确的表达式，因为sizeof后面的内容可以是类型，也可以是变量。

2）该语句的定义不正确，因为此时的bits为一个变量；应该这样修改结构的定义

typedef struct {

signed int bit0:1;

signed int bit1:1;

signed int bit2:1;

signed int bit3:1;

signed int bit4:1;

signed int bit5:1;

signed int bit6:1;

signed int bit7:1;

}bits;

修改后sizeof(bits)表达式依然正确，其值为4；类型改为char后其值为1，注意该值是在VC环境中的32位程序中得到的值，在不同的编译器其值有可能不同，因此在编程时不能自己假定类似结构的大小。

3．

struct bit{

unsigned int a[0]:1,a[1]:1,a[2]:1….a[7]:1;

}

请问这种写法是否正确？为什么？

答案：不正确，位域中的变量不能是数组。

4．

struct a {

int x;

char y;

struct a z;

struct a \*p;

}

请问这种定义结构正确否？ 如果有问题，问题在哪里？

答案：结构中不能对定义结构本身的非指针变量，如果编译器支持则会导致无限嵌套，因此一般编译器都会认为struct a是未定义的类型，即使提前声明也不会有任何用处。

5． 什么是可重入函数？C语言中写可重入函数，应注意的事项？

答案：可重入函数是指能够被多个线程“同时”调用的函数，并且能保证函数结果的正确性的函数。在编写可重入函数时通常要注意如下的一些问题：

尽量不要使用全局变量，静态变量，如果使用了应该注意对变量访问的互斥。通常可以根据具体的情况采用：信号量机制，关调度机制，关中断机制等方式来保证函数的可重入性。

不要调用不可重入的函数，调用了不可重入的函数会使该函数也变为不可重入的函数。

注意对系统中的临界资源，互斥资源的访问方式，防止使函数成为不可重入的函数。

一般驱动程序都是不可重入的函数，因此在编写驱动程序时一定要注意重入的问题。

6． 简述stack frame 的含义。

答案：stack frame的中文译名为：栈框架，表示函数在栈空间的调用层次，以x86平台的函数调用为例，通常一个函数编译成汇编程序，都有如下的结构：

其中的leave指令相当于：mov ebp,esp ；pop ebp

各个函数在栈空间的映象为：

test1函数 test2函数 test3函数

因此在函数test3中，就可以根据这种栈框架的形式得到函数调用层次上的每个函数的基址指针，当前栈指针，以及函数调用点等信息。

7． printf (“%d%d\n”,++n, power(2,n)); 其中power(2,n)为实现一定功能的函数 如 2^n 。

请问这种表示方法有什么潜在的问题？

答案：编译器的不同，对++n 和power(2,n)处理的先后顺序不一样，形成二义性，造成程

序的移植性差，因此最好把++n 写在printf函数外面，以消除二义性。

printf (s);

请问这样的语句有没有问题？（s为一指向有效字符串的指针）

答案：没有%的话，可以这样表达，如果有%在s中的话，有意想不到的输出结果。

9． 两段代码共存于一个文件，编译时有选择的编译其中的一部分，请问如何实现？

答案：有两种简单的办法可以实现：

在源码中使用条件编译语句，然后在程序文件中定义宏的形式来选择需

要的编译代码。

在源码中使用条件编译语句，然后在编译命令的命令中加入宏定义命令

来实现选择编译。

10．数据结构指针传给函数，函数能访问数据单元，但不能修改实际的内容，如何实现？

答案：定义为指向常量的指针，这样指针所指的数据结构中的内容就不会被改变。如：

const 类型 \*p 或 类型 const \*p

11． 在头文件中定义静态变量，可能产生什么问题？

答案：在使用了该头文件的每个c程序文件中都单独存在一个该静态变量，这样造成空间的浪费并且很容易引起错误。因此建议不要在头文件中定义任何变量。

12．malloc()与 calloc()的区别？

答案：

1）参数上的区别

malloc (size\_t size);

calloc (size\_t n , size\_t size);

malloc分配一块size大小的内存块，而calloc分配一个n\*size大小的内存块

2）返回内存块的状态不同

malloc分配的内存块没有被清零，而calloc分配的内存块是清了零的。但是建议在使用内存时，如果需要初始化，则最好自己按照需要来进行初试化，不要依赖函数的实现说明。

13．寄存器变量可不可以访问其地址？可否是全局变量？在什么场合使用寄存器变量？

答案：这些问题都与编译器的实现有关，建议不要声明全局变量为寄存器变量，即使是局部变量都最好不要声明其为寄存器变量，现在的编译器在优化时都会较为合理的安排寄存器变量的使用，而人为的安排有时会造成优化的低效。

14．"\n" '\n' 的区别？

答案：前者是一个字符串并且以’/0’结束，而后者只是一个简单的字符。

15．包含预定义头文件< > 和" "的区别？

答案：< >只在指定的目录里寻找被包含文件；" "先在当前目录下查找 ，再在指定目录下查找；通常<>方式用于系统的头文件，而一般用户的头文件用" "的方式。

16．strunt S\_A{

int a[10];

};

void f()

{

int i;

strunt S\_A \*s\_ptr;

for (i=0,i<10,i++)

s\_ptr -> a[i] = i;

}

请问这段代码正确否？

答案：这段代码不正确，没有对s\_ptr指针进行初始化，在编程中要注意此类低级错误的发生。

coon @ 23:37:37 | 阅读全文 | 评论 0 | 引用 0 | 编辑

c语言笔试题(八)

2006-09-06

Tag: C语言

　　#pragma pack(8)

　　struct s1{

　　short a;

　　long b;

　　};

　　struct s2{

　　char c;

　　s1 d;

　　long long e;

　　};

　　#pragma pack()

　　问

　　1.sizeof(s2) = ?

　　2.s2的s1中的a后面空了几个字节接着是b?

　　如果您知道答案请在讨论中写出，以下是部份网友的答案，供参考：

　　网友rwxybh(行云)的答案：

　　内存布局是

　　1\*\*\* 11\*\*

　　1111 \*\*\*\*

　　1111 1111

　　所以答案就是24和3

　　下面是一个测试的程序，试一试就知道了，我用的是VC2005

　　#pragma pack(8)

　　struct s1{

　　short a; // 2 BYtes

　　long b; // 4 Bytes

　　};

　　struct s2{

　　char c; // 1 Byte

　　s1 d; // 8 Bytes

　　long long e; // 8 Bytes

　　};

　　// 1\*\*\* 11\*\*

　　// 1111 \*\*\*\*

　　// 1111 1111

　　//

　　// 00 01 02 03 04 05 06 07

　　// 00 01 02 03 04 05 06 07

　　// 00 01 02 03 04 05 06 07

　　//

　　#pragma pack()

　　int main(int argc, char\* argv[])

　　{

　　s2 a;

　　char \*p = (char \*)&a;

　　for(int i=0;i<24;++i)

　　p[i] = (char)(i%8);

　　printf("%d\n",sizeof(a));

　　printf("c=0x%lx\n",a.c);

　　printf("d.a=0x%x\n",a.d.a);

　　printf("d.b=0x%x\n",a.d.b);

　　printf("e=0x%llx\n",a.e);

　　return 0;

　　}

　　结果：

　　24

　　c=0x0

　　d.a=0x504

　　d.b=0x3020100

　　e=0x706050403020100

　　网友 redleaves (ID最吊的网友)的答案和分析：

　　如果代码:

　　#pragma pack(8)

　　struct S1{

　　char a;

　　long b;

　　};

　　struct S2 {

　　char c;

　　struct S1 d;

　　long long e;

　　};

　　#pragma pack()

　　sizeof(S2)结果为24.

　　成员对齐有一个重要的条件,即每个成员分别对齐.即每个成员按自己的方式对齐.

　　也就是说上面虽然指定了按8字节对齐,但并不是所有的成员都是以8字节对齐.其对齐的规则是,每个成员按其类型的对齐参数(通常是这个类型的大小)和指定对齐参数(这里是8字节)中较小的一个对齐.并且结构的长度必须为所用过的所有对齐参数的整数倍,不够就补空字节.

　　S1中,成员a是1字节默认按1字节对齐,指定对齐参数为8,这两个值中取1,a按1字节对齐;成员b是4个字节,默认是按4字节对齐,这时就按4字节对齐,所以sizeof(S1)应该为8;

　　S2中,c和S1中的a一样,按1字节对齐,而d 是个结构,它是8个字节,它按什么对齐呢?对于结构来说,它的默认对齐方式就是它的所有成员使用的对齐参数中最大的一个,S1的就是4.所以,成员d就是按4字节对齐.成员e是8个字节,它是默认按8字节对齐,和指定的一样,所以它对到8字节的边界上,这时,已经使用了12个字节了,所以又添加了4个字节的空,从第16个字节开始放置成员e.这时,长度为24,已经可以被8(成员e按8字节对齐)整除.这样,一共使用了24个字节.

　　a b

　　S1的内存布局：11\*\*,1111,

　　c S1.a S1.b d

　　S2的内存布局：1\*\*\*,11\*\*,1111,\*\*\*\*11111111

　　这里有三点很重要:

　　1.每个成员分别按自己的方式对齐,并能最小化长度

　　2.复杂类型(如结构)的默认对齐方式是它最长的成员的对齐方式,这样在成员是复杂类型时,可以最小化长度

　　3.对齐后的长度必须是成员中最大的对齐参数的整数倍,这样在处理数组时可以保证每一项都边界对齐

　　网友xue23(xue23) 的答案和分析：

　　有程序查一下各个变量的内存地址得知：

　　各个变量在内存中的位置为

　　c\*\*\*aa\*\*

　　bbbb\*\*\*\*

　　dddddddd

　　测试代码为：

　　s2 ss;

　　cout << "ss.c = " << &ss << endl ;

　　cout << "ss.d.a = " <<&ss.d.a << endl;

　　cout << "ss.d.b = " <<&(ss.d.b) <　　cout << "ss.d = " <<&ss.e << endl;

　　print out 各个变量的内存地址不就可以看出来了吗。

　　所以答案是24,2.

　　但是我的想像中应该是这样的分布情况：

　　c\*\*\*\*\*\*\*

　　aa\*\*bbbb

　　dddddddd

　　不知为什么会c和a放在一起，组成8位长度。

coon @ 23:35:17 | 阅读全文 | 评论 0 | 引用 0 | 编辑

c语言笔试题(七)

2006-09-06

Tag: C语言

找错

Void test1()

{

char string[10];

char\* str1="0123456789";

strcpy(string, str1);

}

Void test2()

{

char string[10], str1[10];

for(I=0; I<10;I++)

{

str1[i] ='a';

}

strcpy(string, str1);

}

Void test3(char\* str1)

{

char string[10];

if(strlen(str1)<=10)

{

strcpy(string, str1);

}

}

2. 找错

#define MAX\_SRM 256

DSN get\_SRM\_no()

{

static int SRM\_no;

int I;

for(I=0;I{

SRM\_no %= MAX\_SRM;

if(MY\_SRM.state==IDLE)

{

break;

}

}

if(I>=MAX\_SRM)

return (NULL\_SRM);

else

return SRM\_no;

}

3. 写出程序运行结果

int sum(int a)

{

auto int c=0;

static int b=3;

c+=1;

b+=2;

return(a+b+C);

}

void main()

{

int I;

int a=2;

for(I=0;I<5;I++)

{

printf("%d,", sum(a));

}

}

4.

int func(int a)

{

int b;

switch(a)

{

case 1: 30;

case 2: 20;

case 3: 16;

default: 0

}

return b;

}

则func(1)=?

5:

int a[3];

a[0]=0; a[1]=1; a[2]=2;

int \*p, \*q;

p=a;

q=&a[2];

则a[q-p]=?

6.

定义 int \*\*a[3][4], 则变量占有的内存空间为：\_\_\_\_\_

7.

编写一个函数，要求输入年月日时分秒，输出该年月日时分秒的下一秒。如输入2004年12月31日23时59分59秒，则输出2005年1月1日0时0分0秒。

coon @ 23:34:38 | 阅读全文 | 评论 0 | 引用 0 | 编辑

c语言笔试题(六)

2006-09-06

Tag: C语言

4. static有什么用途？（请至少说明两种）

1.限制变量的作用域

2.设置变量的存储域

7. 引用与指针有什么区别？

1) 引用必须被初始化，指针不必。

2) 引用初始化以后不能被改变，指针可以改变所指的对象。

2) 不存在指向空值的引用，但是存在指向空值的指针。

8. 描述实时系统的基本特性

在特定时间内完成特定的任务，实时性与可靠性

9. 全局变量和局部变量在内存中是否有区别？如果有，是什么区别？

全局变量储存在静态数据库，局部变量在堆栈

10. 什么是平衡二叉树？

左右子树都是平衡二叉树 且左右子树的深度差值的绝对值不大于1

11. 堆栈溢出一般是由什么原因导致的？

没有回收垃圾资源

12. 什么函数不能声明为虚函数？

constructor

13. 冒泡排序算法的时间复杂度是什么？

O(n^2)

14. 写出float x 与“零值”比较的if语句。

if(x>0.000001&&x<-0.000001)

16. Internet采用哪种网络协议？该协议的主要层次结构？

tcp/ip 应用层/传输层/网络层/数据链路层/物理层

17. Internet物理地址和IP地址转换采用什么协议？

ARP (Address Resolution Protocol)（地址解析協議）

18.IP地址的编码分为哪俩部分？

IP地址由两部分组成，网络号和主机号。不过是要和“子网掩码”按位与上之后才能区分哪些是网络位哪些是主机位。

2.用户输入M,N值，从1至N开始顺序循环数数，每数到M输出该数值，直至全部输出。写出C程序。

循环链表，用取余操作做

3.不能做switch()的参数类型是：

switch的参数不能为实型。

華為

1、局部变量能否和全局变量重名？

答：能，局部会屏蔽全局。要用全局变量，需要使用"::"

局部变量可以与全局变量同名，在函数内引用这个变量时，会用到同名的局部变量，而不会用到全局变量。对于有些编译器而言，在同一个函数内可以定义多个同名的局部变量，比如在两个循环体内都定义一个同名的局部变量，而那个局部变量的作用域就在那个循环体内

2、如何引用一个已经定义过的全局变量？

答：extern

可以用引用头文件的方式，也可以用extern关键字，如果用引用头文件方式来引用某个在头文件中声明的全局变理，假定你将那个变写错了，那么在编译期间会报错，如果你用extern方式引用时，假定你犯了同样的错误，那么在编译期间不会报错，而在连接期间报错

3、全局变量可不可以定义在可被多个.C文件包含的头文件中？为什么？

答：可以，在不同的C文件中以static形式来声明同名全局变量。

可以在不同的C文件中声明同名的全局变量，前提是其中只能有一个C文件中对此变量赋初值，此时连接不会出错

4、语句for( ；1 ；)有什么问题？它是什么意思？

答：和while(1)相同。

5、do……while和while……do有什么区别？

答：前一个循环一遍再判断，后一个判断以后再循环

6、请写出下列代码的输出内容

#include

main()

{

int a,b,c,d;

a=10;

b=a++;

c=++a;

d=10\*a++;

printf("b，c，d：%d，%d，%d"，b，c，d）;

return 0;

}

答：10，12，120

1、static全局变量与普通的全局变量有什么区别？static局部变量和普通局部变量有什么区别？static函数与普通函数有什么区别？

全局变量(外部变量)的说明之前再冠以static 就构成了静态的全局变量。全局变量本身就是静态存储方式，静态全局变量当然也是静态存储方式。这两者在存储方式上并无不同。这两者的区别虽在于非静态全局变量的作用域是整个源程序，当一个源程序由多个源文件组成时，非静态的全局变量在各个源文件中都是有效的。 而静态全局变量则限制了其作用域，即只在定义该变量的源文件内有效，在同一源程序的其它源文件中不能使用它。由于静态全局变量的作用域局限于一个源文件内，只能为该源文件内的函数公用，因此可以避免在其它源文件中引起错误。

从以上分析可以看出， 把局部变量改变为静态变量后是改变了它的存储方式即改变了它的生存期。把全局变量改变为静态变量后是改变了它的作用域， 限制了它的使用范围。

static函数与普通函数作用域不同。仅在本文件。只在当前源文件中使用的函数应该说明为内部函数(static)，内部函数应该在当前源文件中说明和定义。对于可在当前源文件以外使用的函数，应该在一个头文件中说明，要使用这些函数的源文件要包含这个头文件

static全局变量与普通的全局变量有什么区别：static全局变量只初使化一次，防止在其他文件单元中被引用;

static局部变量和普通局部变量有什么区别：static局部变量只被初始化一次，下一次依据上一次结果值；

static函数与普通函数有什么区别：static函数在内存中只有一份，普通函数在每个被调用中维持一份拷贝

2、程序的局部变量存在于（堆栈）中，全局变量存在于（静态区 ）中，动态申请数据存在于（ 堆）中。

3、设有以下说明和定义：

typedef union {long i; int k[5]; char c;} DATE;

struct data { int cat; DATE cow; double dog;} too;

DATE max;

则语句 printf("%d",sizeof(struct date)+sizeof(max));的执行结果是：\_\_\_52\_\_\_\_

答：DATE是一个union, 变量公用空间. 里面最大的变量类型是int[5], 占用20个字节. 所以它的大小是20

data是一个struct, 每个变量分开占用空间. 依次为int4 + DATE20 + double8 = 32.

所以结果是 20 + 32 = 52.

当然...在某些16位编辑器下, int可能是2字节,那么结果是 int2 + DATE10 + double8 = 20

4、队列和栈有什么区别？

队列先进先出，栈后进先出

5、写出下列代码的输出内容

#include

int inc(int a)

{

return(++a);

}

int multi(int\*a,int\*b,int\*c)

{

return(\*c=\*a\*\*b);

}

typedef int(FUNC1)(int in);

typedef int(FUNC2) (int\*,int\*,int\*);

void show(FUNC2 fun,int arg1, int\*arg2)

{

INCp=&inc;

int temp =p(arg1);

fun(&temp,&arg1, arg2);

printf("%d\n",\*arg2);

}

main()

{

int a;

show(multi,10,&a);

return 0;

}

答：110

7、请找出下面代码中的所以错误

说明：以下代码是把一个字符串倒序，如“abcd”倒序后变为“dcba”

1、#include"string.h"

2、main()

3、{

4、 char\*src="hello,world";

5、 char\* dest=NULL;

6、 int len=strlen(src);

7、 dest=(char\*)malloc(len);

8、 char\* d=dest;

9、 char\* s=src[len];

10、 while(len--!=0)

11、 d++=s--;

12、 printf("%s",dest);

13、 return 0;

14、}

答：

方法1：

int main(){

char\* src = "hello,world";

int len = strlen(src);

char\* dest = (char\*)malloc(len+1);//要为分配一个空间

char\* d = dest;

char\* s = &src[len-1];//指向最后一个字符

while( len-- != 0 )

\*d++=\*s--;

\*d = 0;//尾部要加

printf("%s\n",dest);

free(dest);// 使用完，应当释放空间，以免造成内存汇泄露

return 0;

}

方法2：

#include

#include

main()

{

char str[]="hello,world";

int len=strlen(str);

char t;

for(int i=0; i

{

t=str[i];

str[i]=str[len-i-1]; str[len-i-1]=t;

}

printf("%s",str);

return 0;

}

1.-1,2,7,28,,126请问28和126中间那个数是什么？为什么？

第一题的答案应该是4^3-1=63

规律是n^3-1(当n为偶数0，2，4)

n^3+1(当n为奇数1，3，5)

答案：63

2.用两个栈实现一个队列的功能？要求给出算法和思路！

设2个栈为A,B, 一开始均为空.

入队:

将新元素push入栈A;

出队:

(1)判断栈B是否为空；

(2)如果不为空，则将栈A中所有元素依次pop出并push到栈B；

(3)将栈B的栈顶元素pop出；

这样实现的队列入队和出队的平摊复杂度都还是O(1), 比上面的几种方法要好。3.在c语言库函数中将一个字符转换成整型的函数是atool()吗，这个函数的原型是什么？

函数名: atol

功 能: 把字符串转换成长整型数

用 法: long atol(const char \*nptr);

程序例:

#include

#include

int main(void)

{

long l;

char \*str = "98765432";

l = atol(lstr);

printf("string = %s integer = %ld\n", str, l);

return(0);

}

2.对于一个频繁使用的短小函数,在C语言中应用什么实现,在C++中应用什么实现?

c用宏定义，c++用inline

3.直接链接两个信令点的一组链路称作什么?

PPP点到点连接

4.接入网用的是什么接口?

5.voip都用了那些协议?

6.软件测试都有那些种类?

黑盒：针对系统功能的测试 白合：测试函数功能，各函数接口

7.确定模块的功能和模块的接口是在软件设计的那个队段完成的?

概要设计阶段

8.enum string

{

x1,

x2,

x3=10,

x4,

x5,

}x;

问x= 0x801005，0x8010f4 ;

9.unsigned char \*p1;

unsigned long \*p2;

p1=(unsigned char \*)0x801000;

p2=(unsigned long \*)0x810000;

请问p1+5= ;

p2+5= ;

三.选择题:

1.Ethternet链接到Internet用到以下那个协议?

A.HDLC;B.ARP;C.UDP;D.TCP;E.ID

2.属于网络层协议的是:

A.TCP;B.IP;C.ICMP;D.X.25

3.Windows消息调度机制是:

A.指令队列;B.指令堆栈;C.消息队列;D.消息堆栈;

4.unsigned short hash(unsigned short key)

{

return (key>>)%256

}

请问hash(16),hash(256)的值分别是:

A.1.16;B.8.32;C.4.16;D.1.32

四.找错题:

1.请问下面程序有什么错误?

int a[60][250][1000],i,j,k;

for(k=0;k<=1000;k++)

for(j=0;j<250;j++)

for(i=0;i<60;i++)

a[i][j][k]=0;

把循环语句内外换一下

2.#define Max\_CB 500

void LmiQueryCSmd(Struct MSgCB \* pmsg)

{

unsigned char ucCmdNum;

......

for(ucCmdNum=0;ucCmdNum

{

......;

}

死循环

3.以下是求一个数的平方的程序,请找出错误:

#define SQUARE(a)((a)\*(a))

int a=5;

int b;

b=SQUARE(a++);

4.typedef unsigned char BYTE

int examply\_fun(BYTE gt\_len; BYTE \*gt\_code)

{

BYTE \*gt\_buf;

gt\_buf=(BYTE \*)MALLOC(Max\_GT\_Length);

......

if(gt\_len>Max\_GT\_Length)

{

return GT\_Length\_ERROR;

}

.......

}

五.问答题:

1.IP Phone的原理是什么?

IPV6

2.TCP/IP通信建立的过程怎样，端口有什么作用？

三次握手，确定是哪个应用程序使用该协议

3.1号信令和7号信令有什么区别，我国某前广泛使用的是那一种？

4.列举5种以上的电话新业务？

微软亚洲技术中心的面试题！！！

1．进程和线程的差别。

线程是指进程内的一个执行单元,也是进程内的可调度实体.

与进程的区别:

(1)调度：线程作为调度和分配的基本单位，进程作为拥有资源的基本单位

(2)并发性：不仅进程之间可以并发执行，同一个进程的多个线程之间也可并发执行

(3)拥有资源：进程是拥有资源的一个独立单位，线程不拥有系统资源，但可以访问隶属于进程的资源.

(4)系统开销：在创建或撤消进程时，由于系统都要为之分配和回收资源，导致系统的开销明显大于创建或撤消线程时的开销。

2.测试方法

人工测试：个人复查、抽查和会审

机器测试：黑盒测试和白盒测试

2．Heap与stack的差别。

Heap是堆，stack是栈。

Stack的空间由操作系统自动分配/释放，Heap上的空间手动分配/释放。

Stack空间有限，Heap是很大的自由存储区

C中的malloc函数分配的内存空间即在堆上,C++中对应的是new操作符。

程序在编译期对变量和函数分配内存都在栈上进行,且程序运行过程中函数调用时参数的传递也在栈上进行

3．Windows下的内存是如何管理的？

4．介绍.Net和.Net的安全性。

5．客户端如何访问.Net组件实现Web Service？

6．C/C++编译器中虚表是如何完成的？

7．谈谈COM的线程模型。然后讨论进程内/外组件的差别。

8．谈谈IA32下的分页机制

小页(4K)两级分页模式，大页(4M)一级

9．给两个变量，如何找出一个带环单链表中是什么地方出现环的？

一个递增一，一个递增二，他们指向同一个接点时就是环出现的地方

10．在IA32中一共有多少种办法从用户态跳到内核态？

通过调用门，从ring3到ring0，中断从ring3到ring0，进入vm86等等

11．如果只想让程序有一个实例运行，不能运行两个。像winamp一样，只能开一个窗口，怎样实现？

用内存映射或全局原子（互斥变量）、查找窗口句柄..

FindWindow，互斥，写标志到文件或注册表,共享内存。.

12．如何截取键盘的响应，让所有的‘a’变成‘b’？

键盘钩子SetWindowsHookEx

　13．Apartment在COM中有什么用？为什么要引入？

　14．存储过程是什么？有什么用？有什么优点？

我的理解就是一堆sql的集合，可以建立非常复杂的查询，编译运行，所以运行一次后，以后再运行速度比单独执行SQL快很多

　15．Template有什么特点？什么时候用？

16．谈谈Windows DNA结构的特点和优点。

网络编程中设计并发服务器，使用多进程 与 多线程 ，请问有什么区别？

1，进程：子进程是父进程的复制品。子进程获得父进程数据空间、堆和栈的复制品。

2，线程：相对与进程而言，线程是一个更加接近与执行体的概念，它可以与同进程的其他线程共享数据，但拥有自己的栈空间，拥有独立的执行序列。

两者都可以提高程序的并发度，提高程序运行效率和响应时间。

线程和进程在使用上各有优缺点：线程执行开销小，但不利于资源管理和保护；而进程正相反。同时，线程适合于在SMP机器上运行，而进程则可以跨机器迁移。

思科

1. 用宏定义写出swap（x，y）

#define swap(x, y)\

x = x + y;\

y = x - y;\

x = x - y;

2.数组a[N]，存放了1至N-1个数，其中某个数重复一次。写一个函数，找出被重复的数字.时间复杂度必须为o（N）函数原型：

int do\_dup(int a[],int N)

3 一语句实现x是否为2的若干次幂的判断

int i = 512;

cout << boolalpha << ((i & (i - 1)) ? false : true) << endl;

4.unsigned int intvert(unsigned int x,int p,int n)实现对x的进行转换,p为起始转化位,n为需要转换的长度,假设起始点在右边.如x=0b0001 0001,p=4,n=3转换后x=0b0110 0001

unsigned int intvert(unsigned int x,int p,int n){

unsigned int \_t = 0;

unsigned int \_a = 1;

for(int i = 0; i < n; ++i){

\_t |= \_a;

\_a = \_a << 1;

}

\_t = \_t << p;

x ^= \_t;

return x;

}

慧通：

什么是预编译

何时需要预编译：

１、总是使用不经常改动的大型代码体。

２、程序由多个模块组成，所有模块都使用一组标准的包含文件和相同的编译选项。在这种情况下，可以将所有包含文件预编译为一个预编译头。

char \* const p;

char const \* p

const char \*p

上述三个有什么区别？

char \* const p; //常量指针，p的值不可以修改

char const \* p；//指向常量的指针，指向的常量值不可以改

const char \*p； //和char const \*p

char str1[] = "abc";

char str2[] = "abc";

const char str3[] = "abc";

const char str4[] = "abc";

const char \*str5 = "abc";

const char \*str6 = "abc";

char \*str7 = "abc";

char \*str8 = "abc";

cout << ( str1 == str2 ) << endl;

cout << ( str3 == str4 ) << endl;

cout << ( str5 == str6 ) << endl;

cout << ( str7 == str8 ) << endl;

结果是：0 0 1 1

解答：str1,str2,str3,str4是数组变量，它们有各自的内存空间；

而str5,str6,str7,str8是指针，它们指向相同的常量区域。

12. 以下代码中的两个sizeof用法有问题吗？[C易]

void UpperCase( char str[] ) // 将 str 中的小写字母转换成大写字母

{

for( size\_t i=0; i

if( 'a'<=str[i] && str[i]<='z' )

str[i] -= ('a'-'A' );

}

char str[] = "aBcDe";

cout << "str字符长度为: " << sizeof(str)/sizeof(str[0]) << endl;

UpperCase( str );

cout << str << endl;

答：函数内的sizeof有问题。根据语法，sizeof如用于数组，只能测出静态数组的大小，无法检测动态分配的或外部数组大小。函数外的str是一个静态定义的数组，因此其大小为6，函数内的str实际只是一个指向字符串的指针，没有任何额外的与数组相关的信息，因此sizeof作用于上只将其当指针看，一个指针为4个字节，因此返回4。

一个32位的机器,该机器的指针是多少位

指针是多少位只要看地址总线的位数就行了。80386以后的机子都是32的数据总线。所以指针的位数就是4个字节了。

main()

{

int a[5]={1,2,3,4,5};

int \*ptr=(int \*)(&a+1);

printf("%d,%d",\*(a+1),\*(ptr-1));

}

输出：2,5

\*(a+1）就是a[1]，\*(ptr-1)就是a[4],执行结果是2，5

&a+1不是首地址+1，系统会认为加一个a数组的偏移，是偏移了一个数组的大小（本例是5个int）

int \*ptr=(int \*)(&a+1);

则ptr实际是&(a[5]),也就是a+5

原因如下：

&a是数组指针，其类型为 int (\*)[5];

而指针加1要根据指针类型加上一定的值，

不同类型的指针+1之后增加的大小不同

a是长度为5的int数组指针，所以要加 5\*sizeof(int)

所以ptr实际是a[5]

但是prt与(&a+1)类型是不一样的(这点很重要)

所以prt-1只会减去sizeof(int\*)

a,&a的地址是一样的，但意思不一样，a是数组首地址，也就是a[0]的地址，&a是对象（数组）首地址，a+1是数组下一元素的地址，即a[1],&a+1是下一个对象的地址，即a[5].

1.请问以下代码有什么问题：

int main()

{

char a;

char \*str=&a;

strcpy(str,"hello");

printf(str);

return 0;

}

没有为str分配内存空间，将会发生异常

问题出在将一个字符串复制进一个字符变量指针所指地址。虽然可以正确输出结果，但因为越界进行内在读写而导致程序崩溃。

char\* s="AAA";

printf("%s",s);

s[0]='B';

printf("%s",s);

有什么错？

"AAA"是字符串常量。s是指针，指向这个字符串常量，所以声明s的时候就有问题。

cosnt char\* s="AAA";

然后又因为是常量，所以对是s[0]的赋值操作是不合法的。

1、写一个“标准”宏，这个宏输入两个参数并返回较小的一个。

.#define Min(X, Y) ((X)>(Y)?(Y):(X))//结尾没有;

2、嵌入式系统中经常要用到无限循环，你怎么用C编写死循环。

while(1){}或者for(;;)

3、关键字static的作用是什么？

定义静态变量

4、关键字const有什么含意？

表示常量不可以修改的变量。

5、关键字volatile有什么含意？并举出三个不同的例子？

提示编译器对象的值可能在编译器未监测到的情况下改变。

int (\*s[10])(int) 表示的是什么啊

int (\*s[10])(int) 函数指针数组，每个指针指向一个int func(int param)的函数。

1.有以下表达式：

int a=248; b=4;int const c=21;const int \*d=&a;

int \*const e=&b;int const \*f const =&a;

请问下列表达式哪些会被编译器禁止？为什么？

\*c=32;d=&b;\*d=43;e=34;e=&a;f=0x321f;

\*c 这是个什么东东，禁止

\*d 说了是const， 禁止

e = &a 说了是const 禁止

const \*f const =&a; 禁止

2.交换两个变量的值，不使用第三个变量。即a=3,b=5,交换之后a=5,b=3;

有两种解法, 一种用算术算法, 一种用^(异或)

a = a + b;

b = a - b;

a = a - b;

or

a = a^b;// 只能对int,char..

b = a^b;

a = a^b;

or

a ^= b ^= a;

3.c和c++中的struct有什么不同？

c和c++中struct的主要区别是c中的struct不可以含有成员函数，而c++中的struct可以。c++中struct和class的主要区别在于默认的存取权限不同，struct默认为public，而class默认为private

4.#include

#include

void getmemory(char \*p)

{

p=(char \*) malloc(100);

strcpy(p,"hello world");

}

int main( )

{

char \*str=NULL;

getmemory(str);

printf("%s/n",str);

free(str);

return 0;

}

程序崩溃，getmemory中的malloc 不能返回动态内存， free（）对str操作很危险

5.char szstr[10];

strcpy(szstr,"0123456789");

产生什么结果？为什么？

长度不一样，会造成非法的OS

6.列举几种进程的同步机制，并比较其优缺点。

原子操作

信号量机制

自旋锁

管程，会合，分布式系统

7.进程之间通信的途径

共享存储系统

消息传递系统

管道：以文件系统为基础

11.进程死锁的原因

资源竞争及进程推进顺序非法

12.死锁的4个必要条件

互斥、请求保持、不可剥夺、环路

13.死锁的处理

鸵鸟策略、预防策略、避免策略、检测与解除死锁

15. 操作系统中进程调度策略有哪几种？

FCFS(先来先服务)，优先级，时间片轮转，多级反馈

8.类的静态成员和非静态成员有何区别？

类的静态成员每个类只有一个，非静态成员每个对象一个

9.纯虚函数如何定义？使用时应注意什么？

virtual void f()=0;

是接口，子类必须要实现

10.数组和链表的区别

数组：数据顺序存储，固定大小

连表：数据可以随机存储，大小可动态改变

12.ISO的七层模型是什么？tcp/udp是属于哪一层？tcp/udp有何优缺点？

应用层

表示层

会话层

运输层

网络层

物理链路层

物理层

tcp /udp属于运输层

TCP 服务提供了数据流传输、可靠性、有效流控制、全双工操作和多路复用技术等。

与 TCP 不同， UDP 并不提供对 IP 协议的可靠机制、流控制以及错误恢复功能等。由于 UDP 比较简单， UDP 头包含很少的字节，比 TCP 负载消耗少。

tcp: 提供稳定的传输服务，有流量控制，缺点是包头大，冗余性不好

udp: 不提供稳定的服务，包头小，开销小

1：(void \*)ptr 和 (\*(void\*\*))ptr的结果是否相同？其中ptr为同一个指针

.(void \*)ptr 和 (\*(void\*\*))ptr值是相同的

2：int main()

{

int x=3;

printf("%d",x);

return 1;

}

问函数既然不会被其它函数调用，为什么要返回1？

mian中，c标准认为0表示成功，非0表示错误。具体的值是某中具体出错信息

1，要对绝对地址0x100000赋值，我们可以用

(unsigned int\*)0x100000 = 1234;

那么要是想让程序跳转到绝对地址是0x100000去执行，应该怎么做？

\*((void (\*)( ))0x100000 ) ( );

首先要将0x100000强制转换成函数指针,即:

(void (\*)())0x100000

然后再调用它:

\*((void (\*)())0x100000)();

用typedef可以看得更直观些:

typedef void(\*)() voidFuncPtr;

\*((voidFuncPtr)0x100000)();

2，已知一个数组table，用一个宏定义，求出数据的元素个数

#define NTBL

#define NTBL (sizeof(table)/sizeof(table[0]))

面试题: 线程与进程的区别和联系? 线程是否具有相同的堆栈? dll是否有独立的堆栈?

进程是死的，只是一些资源的集合，真正的程序执行都是线程来完成的，程序启动的时候操作系统就帮你创建了一个主线程。

每个线程有自己的堆栈。

DLL 中有没有独立的堆栈，这个问题不好回答，或者说这个问题本身是否有问题。因为DLL中的代码是被某些线程所执行，只有线程拥有堆栈，如果DLL中的代码是 EXE中的线程所调用，那么这个时候是不是说这个DLL没有自己独立的堆栈？如果DLL中的代码是由DLL自己创建的线程所执行，那么是不是说DLL有独立的堆栈？

以上讲的是堆栈，如果对于堆来说，每个DLL有自己的堆，所以如果是从DLL中动态分配的内存，最好是从DLL中删除，如果你从DLL中分配内存，然后在EXE中，或者另外一个DLL中删除，很有可能导致程序崩溃

unsigned short A = 10;

printf("~A = %u\n", ~A);

char c=128;

printf("c=%d\n",c);

输出多少？并分析过程

第一题，～A ＝0xfffffff5,int值 为－11，但输出的是uint。所以输出4294967285

第二题，c＝0x10,输出的是int，最高位为1，是负数，所以它的值就是0x00的补码就是128，所以输出－128。

这两道题都是在考察二进制向int或uint转换时的最高位处理。

分析下面的程序：

void GetMemory(char \*\*p,int num)

{

\*p=(char \*)malloc(num);

}

int main()

{

char \*str=NULL;

GetMemory(&str,100);

strcpy(str,"hello");

free(str);

if(str!=NULL)

{

strcpy(str,"world");

}

printf("\n str is %s",str);

getchar();

}

问输出结果是什么？希望大家能说说原因，先谢谢了

输出str is world。

free 只是释放的str指向的内存空间,它本身的值还是存在的.

所以free之后，有一个好的习惯就是将str=NULL.

此时str指向空间的内存已被回收,如果输出语句之前还存在分配空间的操作的话,这段存储空间是可能被重新分配给其他变量的,

尽管这段程序确实是存在大大的问题（上面各位已经说得很清楚了），但是通常会打印出world来。

这是因为，进程中的内存管理一般不是由操作系统完成的，而是由库函数自己完成的。

当你malloc一块内存的时候，管理库向操作系统申请一块空间（可能会比你申请的大一些），然后在这块空间中记录一些管理信息（一般是在你申请的内存前面一点），并将可用内存的地址返回。但是释放内存的时候，管理库通常都不会将内存还给操作系统，因此你是可以继续访问这块地址的，只不过。。。。。。。。楼上都说过了，最好别这么干。

char a[10],strlen(a)为什么等于15？运行的结果

#include "stdio.h"

#include "string.h"

void main()

{

char aa[10];

printf("%d",strlen(aa));

}

sizeof()和初不初始化，没有关系；

strlen()和初始化有关。

char (\*str)[20];/\*str是一个数组指针，即指向数组的指针．\*/

char \*str[20];/\*str是一个指针数组，其元素为指针型数据．\*/

long a=0x801010;

a+5=?

0x801010用二进制表示为：“1000 0000 0001 0000 0001 0000”，十进制的值为8392720，再加上5就是8392725罗

1)给定结构struct A

{

char t:4;

char k:4;

unsigned short i:8;

unsigned long m;

};问sizeof(A) = ?

给定结构struct A

{

char t:4; 4位

char k:4; 4位

unsigned short i:8; 8位

unsigned long m; // 偏移2字节保证4字节对齐

}; // 共8字节

2)下面的函数实现在一个数上加一个数，有什么错误？请改正。

int add\_n ( int n )

{

static int i = 100;

i += n;

return i;

}

当你第二次调用时得不到正确的结果，难道你写个函数就是为了调用一次？问题就出在 static上？

// 帮忙分析一下

#include

#include

#include

#include

#include

#include

typedef struct AA

{

int b1:5;

int b2:2;

}AA;

void main()

{

AA aa;

char cc[100];

strcpy(cc,"0123456789abcdefghijklmnopqrstuvwxyz");

memcpy(&aa,cc,sizeof(AA));

cout << aa.b1 <

cout << aa.b2 <

}

答案是 -16和１

首先sizeof(AA)的大小为4,b1和b2分别占5bit和2bit.

经过strcpy和memcpy后,aa的4个字节所存放的值是:

0,1,2,3的ASC码，即00110000,00110001,00110010,00110011

所以，最后一步：显示的是这４个字节的前５位，和之后的２位

分别为：10000,和01

因为int是有正负之分　　所以：答案是-16和１

求函数返回值，输入x=9999;

int func （ x ）

{

int countx = 0;

while ( x )

{

countx ++;

x = x&(x-1);

}

return countx;

}

结果呢？

知道了这是统计9999的二进制数值中有多少个1的函数，且有

9999＝9×1024＋512＋256＋15

9×1024中含有1的个数为2；

512中含有1的个数为1；

256中含有1的个数为1；

15中含有1的个数为4；

故共有1的个数为8，结果为8。

1000 - 1 = 0111，正好是原数取反。这就是原理。

用这种方法来求1的个数是很效率很高的。

不必去一个一个地移位。循环次数最少。

int a,b,c 请写函数实现C=a+b ,不可以改变数据类型,如将c改为long int,关键是如何处理溢出问题

bool add (int a, int b,int \*c)

{

\*c=a+b;

return (a>0 && b>0 &&(\*c<0 && b<0 &&(\*c>a || \*c>b)));

}

分析：

struct bit

{ int a:3;

int b:2;

int c:3;

};

int main()

{

bit s;

char \*c=(char\*)&s;

cout<<

\*c=0x99;

cout << s.a <<<<<

int a=-1;

printf("%x",a);

return 0;

}

输出为什么是

4

1

-1

-4

ffffffff

因为0x99在内存中表示为 100 11 001 , a = 001, b = 11, c = 100

当c为有符合数时, c = 100, 最高1为表示c为负数，负数在计算机用补码表示，所以c = -4;同理

b = -1;

当c为有符合数时, c = 100,即 c = 4,同理 b = 3

位域 ：

有些信息在存储时，并不需要占用一个完整的字节， 而只需占几个或一个二进制位。例如在存放一个开关量时，只有0和1 两种状态，用一位二进位即可。为了节省存储空间，并使处理简便，Ｃ语言又提供了一种数据结构，称为“位域”或“位段”。所谓“位域”是把一个字节中的二进位划分为几个不同的区域，并说明每个区域的位数。每个域有一个域名，允许在程序中按域名进行操作。这样就可以把几个不同的对象用一个字节的二进制位域来表示。一、位域的定义和位域变量的说明位域定义与结构定义相仿，其形式为：

struct 位域结构名

{ 位域列表 };

其中位域列表的形式为： 类型说明符 位域名：位域长度

例如：

struct bs

{

int a:8;

int b:2;

int c:6;

};

位域变量的说明与结构变量说明的方式相同。 可采用先定义后说明，同时定义说明或者直接说明这三种方式。例如：

struct bs

{

int a:8;

int b:2;

int c:6;

}data;

说明data为bs变量，共占两个字节。其中位域a占8位，位域b占2位，位域c占6位。对于位域的定义尚有以下几点说明：

1. 一个位域必须存储在同一个字节中，不能跨两个字节。如一个字节所剩空间不够存放另一位域时，应从下一单元起存放该位域。也可以有意使某位域从下一单元开始。例如：

struct bs

{

unsigned a:4

unsigned :0 /\*空域\*/

unsigned b:4 /\*从下一单元开始存放\*/

unsigned c:4

}

在这个位域定义中，a占第一字节的4位，后4位填0表示不使用，b从第二字节开始，占用4位，c占用4位。

2. 由于位域不允许跨两个字节，因此位域的长度不能大于一个字节的长度，也就是说不能超过8位二进位。

3. 位域可以无位域名，这时它只用来作填充或调整位置。无名的位域是不能使用的。例如：

struct k

{

int a:1

int :2 /\*该2位不能使用\*/

int b:3

int c:2

};

从以上分析可以看出，位域在本质上就是一种结构类型， 不过其成员是按二进位分配的。

二、位域的使用位域的使用和结构成员的使用相同，其一般形式为： 位域变量名?位域名 位域允许用各种格式输出。

main(){

struct bs

{

unsigned a:1;

unsigned b:3;

unsigned c:4;

} bit,\*pbit;

bit.a=1;

bit.b=7;

bit.c=15;

pri

改错：

#include

int main(void) {

int \*\*p;

int arr[100];

p = &arr;

return 0;

}

解答：

搞错了,是指针类型不同,

int \*\*p; //二级指针

&arr; //得到的是指向第一维为100的数组的指针

#include

int main(void) {

int \*\*p, \*q;

int arr[100];

q = arr;

p = &q;

return 0;

}

下面这个程序执行后会有什么错误或者效果:

#define MAX 255

int main()

{

unsigned char A[MAX],i;//i被定义为unsigned char

for (i=0;i<=MAX;i++)

A[i]=i;

}

解答：死循环加数组越界访问（C/C++不进行数组越界检查）

MAX=255

数组A的下标范围为:0..MAX-1,这是其一..

其二.当i循环到255时,循环内执行:

A[255]=255;

这句本身没有问题..但是返回for (i=0;i<=MAX;i++)语句时,

由于unsigned char的取值范围在(0..255),i++以后i又为0了..无限循环下去.

struct name1{

char str;

short x;

int num;

}

struct name2{

char str;

int num;

short x;

}

sizeof(struct name1)=8,sizeof(struct name2)=12

在第二个结构中，为保证num按四个字节对齐，char后必须留出3字节的空间；同时为保证整个结构的自然对齐（这里是4字节对齐），在x后还要补齐2个字节，这样就是12字节。

intel：

A.c 和B.c两个c文件中使用了两个相同名字的static变量,编译的时候会不会有问题?这两个static变量会保存到哪里（栈还是堆或者其他的）?

static的全局变量，表明这个变量仅在本模块中有意义，不会影响其他模块。

他们都放在数据区，但是编译器对他们的命名是不同的。

如果要使变量在其他模块也有意义的话，需要使用extern关键字。

struct s1

{

int i: 8;

int j: 4;

int a: 3;

double b;

};

struct s2

{

int i: 8;

int j: 4;

double b;

int a:3;

};

printf("sizeof(s1)= %d\n", sizeof(s1));

printf("sizeof(s2)= %d\n", sizeof(s2));

result: 16, 24

第一个struct s1

{

int i: 8;

int j: 4;

int a: 3;

double b;

};

理论上是这样的，首先是i在相对0的位置，占8位一个字节，然后，j就在相对一个字节的位置，由于一个位置的字节数是4位的倍数，因此不用对齐，就放在那里了，然后是a，要在3位的倍数关系的位置上，因此要移一位，在15位的位置上放下，目前总共是18位，折算过来是2字节2位的样子，由于double是8 字节的，因此要在相对0要是8个字节的位置上放下，因此从18位开始到8个字节之间的位置被忽略，直接放在8字节的位置了，因此，总共是16字节。

第二个最后会对照是不是结构体内最大数据的倍数，不是的话，会补成是最大数据的倍数

1）读文件file1.txt的内容（例如）：

12

34

56

输出到file2.txt：

56

34

12

（逆序）

2）输出和为一个给定整数的所有组合

例如n=5

5=1+4；5=2+3（相加的数不能重复）

则输出

1，4；2，3。

望高手赐教！！

第一题,注意可增长数组的应用.

#include

#include

int main(void)

{

int MAX = 10;

int \*a = (int \*)malloc(MAX \* sizeof(int));

int \*b;

FILE \*fp1;

FILE \*fp2;

fp1 = fopen("a.txt","r");

if(fp1 == NULL)

{printf("error1");

exit(-1);

}

fp2 = fopen("b.txt","w");

if(fp2 == NULL)

{printf("error2");

exit(-1);

}

int i = 0;

int j = 0;

while(fscanf(fp1,"%d",&a[i]) != EOF)

{

i++;

j++;

if(i >= MAX)

{

MAX = 2 \* MAX;

b = (int\*)realloc(a,MAX \* sizeof(int));

if(b == NULL)

{

printf("error3");

exit(-1);

}

a = b;

}

}

for(;--j >= 0;)

fprintf(fp2,"%d\n",a[j]);

fclose(fp1);

fclose(fp2);

return 0;

}

第二题.

#include

int main(void)

{

unsigned long int i,j,k;

printf("please input the number\n");

scanf("%d",&i);

if( i % 2 == 0)

j = i / 2;

else

j = i / 2 + 1;

printf("The result is \n");

for(k = 0; k < j; k++)

printf("%d = %d + %d\n",i,k,i - k);

return 0;

}

#include

void main()

{

unsigned long int a,i=1;

scanf("%d",&a);

if(a%2==0)

{

for(i=1;i

printf("%d",a,a-i);

}

else

for(i=1;i<=a/2;i++)

printf(" %d, %d",i,a-i);

}

兄弟,这样的题目若是做不出来实在是有些不应该, 给你一个递规反向输出字符串的例子,可谓是反序的经典例程.

void inverse(char \*p)

{

if( \*p = = '' )

return;

inverse( p+1 );

printf( "%c", \*p );

}

int main(int argc, char \*argv[])

{

inverse("abc");

return 0;

}

借签了楼上的“递规反向输出”

#include

void test(FILE \*fread, FILE \*fwrite)

{

char buf[1024] = {0};

if (!fgets(buf, sizeof(buf), fread))

return;

test( fread, fwrite );

fputs(buf, fwrite);

}

int main(int argc, char \*argv[])

{

FILE \*fr = NULL;

FILE \*fw = NULL;

fr = fopen("data", "rb");

fw = fopen("dataout", "wb");

test(fr, fw);

fclose(fr);

fclose(fw);

return 0;

}

在对齐为4的情况下

struct BBB

{

long num；

char \*name;

short int data;

char ha;

short ba[5];

}\*p;

p=0x1000000;

p+0x200=\_\_\_\_;

(Ulong)p+0x200=\_\_\_\_;

(char\*)p+0x200=\_\_\_\_;

希望各位达人给出答案和原因，谢谢拉

解答：假设在32位CPU上，

sizeof(long) = 4 bytes

sizeof(char \*) = 4 bytes

sizeof(short int) = sizeof(short) = 2 bytes

sizeof(char) = 1 bytes

由于是4字节对齐，

sizeof(struct BBB) = sizeof(\*p)

= 4 + 4 + 2 + 1 + 1/\*补齐\*/ + 2\*5 + 2/\*补齐\*/ = 24 bytes (经Dev-C++验证)

p=0x1000000;

p+0x200=\_\_\_\_;

= 0x1000000 + 0x200\*24

(Ulong)p+0x200=\_\_\_\_;

= 0x1000000 + 0x200

(char\*)p+0x200=\_\_\_\_;

= 0x10000

# C++/C试题

本试题仅用于考查C++/C程序员的基本编程技能。内容限于C++/C常用语法，不涉及数据结构、算法以及深奥的语法。考试成绩能反映出考生的编程质量以及对C++/C的理解程度，但不能反映考生的智力和软件开发能力。

笔试时间90分钟。请考生认真答题，切勿轻视。

**一、请填写BOOL , float, 指针变量 与“零值”比较的 if 语句。（10分）**

提示：这里“零值”可以是0, 0.0 , FALSE或者“空指针”。例如 int 变量 n 与“零值”比较的 if 语句为：

if ( n == 0 )

if ( n != 0 )

以此类推。

|  |
| --- |
| 请写出 BOOL flag 与“零值”比较的 if 语句： |
| 请写出 float x 与“零值”比较的 if 语句： |
| 请写出 char \*p 与“零值”比较的 if 语句： |

**二、以下为Windows NT下的32位C++程序，请计算sizeof的值（10分）**

|  |  |
| --- | --- |
| char str[] = “Hello” ;  char \*p = str ;  int n = 10;  请计算  sizeof (str ) =    sizeof ( p ) =    sizeof ( n ) = | void Func ( char str[100])  {  请计算  sizeof( str ) =  } |
| void \*p = malloc( 100 );  请计算  sizeof ( p ) = |

**三、简答题（25分）**

1、头文件中的 ifndef/define/endif 干什么用？

2、#include <filename.h> 和 #include “filename.h” 有什么区别？

3、const 有什么用途？（请至少说明两种）

4、在C++ 程序中调用被 C编译器编译后的函数，为什么要加 extern “C”声明？

5、请简述以下两个for循环的优缺点

|  |  |
| --- | --- |
| // 第一个  for (i=0; i<N; i++)  {  if (condition)  DoSomething();  else  DoOtherthing();  } | // 第二个  if (condition)  {  for (i=0; i<N; i++)  DoSomething();  }  else  {  for (i=0; i<N; i++)  DoOtherthing();  } |
| 优点：  缺点： | 优点：  缺点： |

**四、有关内存的思考题（20分）**

|  |  |
| --- | --- |
| void GetMemory(char \*p)  {  p = (char \*)malloc(100);  }  void Test(void)  {  char \*str = NULL;  GetMemory(str);  strcpy(str, "hello world");  printf(str);  }  请问运行Test函数会有什么样的结果？  答： | char \*GetMemory(void)  {  char p[] = "hello world";  return p;  }  void Test(void)  {  char \*str = NULL;  str = GetMemory();  printf(str);  }  请问运行Test函数会有什么样的结果？  答： |
| Void GetMemory2(char \*\*p, int num)  {  \*p = (char \*)malloc(num);  }  void Test(void)  {  char \*str = NULL;  GetMemory(&str, 100);  strcpy(str, "hello");  printf(str);  }  请问运行Test函数会有什么样的结果？  答： | void Test(void)  {  char \*str = (char \*) malloc(100);  strcpy(str, “hello”);  free(str);  if(str != NULL)  {  strcpy(str, “world”);  printf(str);  }  }  请问运行Test函数会有什么样的结果？  答： |

**五、编写strcpy函数（10分）**

已知strcpy函数的原型是

char \*strcpy(char \*strDest, const char \*strSrc);

其中strDest是目的字符串，strSrc是源字符串。

（1）不调用C++/C的字符串库函数，请编写函数 strcpy

（2）strcpy能把strSrc的内容复制到strDest，为什么还要char \* 类型的返回值？

**六、编写类String的构造函数、析构函数和赋值函数（25分）**

已知类String的原型为：

class String

{

public:

String(const char \*str = NULL); // 普通构造函数

String(const String &other); // 拷贝构造函数

~ String(void); // 析构函数

String & operate =(const String &other); // 赋值函数

private:

char \*m\_data; // 用于保存字符串

};

请编写String的上述4个函数。

# C++/C试题的答案与评分标准

一、请填写BOOL , float, 指针变量 与“零值”比较的 if 语句。（10分）

|  |  |
| --- | --- |
| 请写出 BOOL flag 与“零值”比较的 if 语句。（3分） | |
| **标准答案：**  **if ( flag )**  **if ( !flag )** | 如下写法均属不良风格，不得分。  if (flag == TRUE)  if (flag == 1 )  if (flag == FALSE)  if (flag == 0) |
| 请写出 float x 与“零值”比较的 if 语句。（4分） | |
| **标准答案示例：**  **const float EPSINON = 0.00001;**  **if ((x >= - EPSINON) && (x <= EPSINON)**  不可将浮点变量用“==”或“！=”与数字比较，应该设法转化成“>=”或“<=”此类形式。 | 如下是错误的写法，不得分。  if (x == 0.0)  if (x != 0.0) |
| 请写出 char \*p 与“零值”比较的 if 语句。（3分） | |
| **标准答案：**  **if (p == NULL)**  **if (p != NULL)** | 如下写法均属不良风格，不得分。  if (p == 0)  if (p != 0)  if (p)  if (!) |

**二、以下为Windows NT下的32位C++程序，请计算sizeof的值（10分）**

|  |  |
| --- | --- |
| char str[] = “Hello” ;  char \*p = str ;  int n = 10;  请计算  **sizeof (str ) = 6 （2分）**    **sizeof ( p ) = 4 （2分）**    **sizeof ( n ) = 4 （2分）** | void Func ( char str[100])  {  请计算  **sizeof( str ) = 4 （2分）**  } |
| void \*p = malloc( 100 );  请计算  **sizeof ( p ) = 4 （2分）** |

**三、简答题（25分）**

1、头文件中的 ifndef/define/endif 干什么用？（5分）

**答：防止该头文件被重复引用。**

2、#include <filename.h> 和 #include “filename.h” 有什么区别？（5分）

**答：对于#include <filename.h> ，编译器从标准库路径开始搜索 filename.h**

**对于#include “filename.h” ，编译器从用户的工作路径开始搜索 filename.h**

3、const 有什么用途？（请至少说明两种）（5分）

**答：（1）可以定义 const 常量**

**（2）const可以修饰函数的参数、返回值，甚至函数的定义体。被const修饰的东西都受到强制保护，可以预防意外的变动，能提高程序的健壮性。**

4、在C++ 程序中调用被 C编译器编译后的函数，为什么要加 extern “C”？ （5分）

**答：C++语言支持函数重载，C语言不支持函数重载。函数被C++编译后在库中的名字与C语言的不同。假设某个函数的原型为： void foo(int x, int y);**

**该函数被C编译器编译后在库中的名字为\_foo，而C++编译器则会产生像\_foo\_int\_int之类的名字。**

**C++提供了C连接交换指定符号extern“C”来解决名字匹配问题。**

**5、请简述以下两个for循环的优缺点（5分）**

|  |  |
| --- | --- |
| for (i=0; i<N; i++)  {  if (condition)  DoSomething();  else  DoOtherthing();  } | if (condition)  {  for (i=0; i<N; i++)  DoSomething();  }  else  {  for (i=0; i<N; i++)  DoOtherthing();  } |
| **优点：程序简洁**  **缺点：多执行了N-1次逻辑判断，并且打断了循环“流水线”作业，使得编译器不能对循环进行优化处理，降低了效率。** | **优点：循环的效率高**  **缺点：程序不简洁** |

**四、有关内存的思考题（每小题5分，共20分）**

|  |  |
| --- | --- |
| void GetMemory(char \*p)  {  p = (char \*)malloc(100);  }  void Test(void)  {  char \*str = NULL;  GetMemory(str);  strcpy(str, "hello world");  printf(str);  }  请问运行Test函数会有什么样的结果？  **答：程序崩溃。**  **因为GetMemory并不能传递动态内存，**  **Test函数中的 str一直都是 NULL。**  **strcpy(str, "hello world");将使程序崩溃。** | char \*GetMemory(void)  {  char p[] = "hello world";  return p;  }  void Test(void)  {  char \*str = NULL;  str = GetMemory();  printf(str);  }  请问运行Test函数会有什么样的结果？  **答：可能是乱码。**  **因为GetMemory返回的是指向“栈内存”的指针，该指针的地址不是 NULL，但其原现的内容已经被清除，新内容不可知。** |
| void GetMemory2(char \*\*p, int num)  {  \*p = (char \*)malloc(num);  }  void Test(void)  {  char \*str = NULL;  GetMemory(&str, 100);  strcpy(str, "hello");  printf(str);  }  请问运行Test函数会有什么样的结果？  **答：**  **（1）能够输出hello**  **（2）内存泄漏** | void Test(void)  {  char \*str = (char \*) malloc(100);  strcpy(str, “hello”);  free(str);  if(str != NULL)  {  strcpy(str, “world”);  printf(str);  }  }  请问运行Test函数会有什么样的结果？  **答：篡改动态内存区的内容，后果难以预料，非常危险。**  **因为free(str);之后，str成为野指针，**  **if(str != NULL)语句不起作用。** |

**五、编写strcpy函数（10分）**

已知strcpy函数的原型是

char \*strcpy(char \*strDest, const char \*strSrc);

其中strDest是目的字符串，strSrc是源字符串。

（1）不调用C++/C的字符串库函数，请编写函数 strcpy

**char \*strcpy(char \*strDest, const char \*strSrc);**

**{**

**assert((strDest!=NULL) && (strSrc !=NULL)); // 2分**

**char \*address = strDest; // 2分**

**while( (\*strDest++ = \* strSrc++) != ‘\0’ ) // 2分**

**NULL ;**

**return address ; // 2分**

**}**

（2）strcpy能把strSrc的内容复制到strDest，为什么还要char \* 类型的返回值？

**答：为了实现链式表达式。 // 2分**

**例如 int length = strlen( strcpy( strDest, “hello world”) );**

**六、编写类String的构造函数、析构函数和赋值函数（25分）**

已知类String的原型为：

class String

{

public:

String(const char \*str = NULL); // 普通构造函数

String(const String &other); // 拷贝构造函数

~ String(void); // 析构函数

String & operate =(const String &other); // 赋值函数

private:

char \*m\_data; // 用于保存字符串

};

请编写String的上述4个函数。

**标准答案：**

// String的析构函数

**String::~String(void) // 3分**

**{**

**delete [] m\_data;**

**// 由于m\_data是内部数据类型，也可以写成 delete m\_data;**

**}**

// String的普通构造函数

**String::String(const char \*str) // 6分**

**{**

**if(str==NULL)**

**{**

**m\_data = new char[1]; // 若能加 NULL 判断则更好**

**\*m\_data = ‘\0’;**

**}**

**else**

**{**

**int length = strlen(str);**

**m\_data = new char[length+1]; // 若能加 NULL 判断则更好**

**strcpy(m\_data, str);**

**}**

**}**

// 拷贝构造函数

**String::String(const String &other) // 3分**

**{**

**int length = strlen(other.m\_data);**

**m\_data = new char[length+1]; // 若能加 NULL 判断则更好**

**strcpy(m\_data, other.m\_data);**

**}**

// 赋值函数

**String & String::operate =(const String &other) // 13分**

**{**

**// (1) 检查自赋值 // 4分**

**if(this == &other)**

**return \*this;**

**// (2) 释放原有的内存资源 // 3分**

**delete [] m\_data;**

**// （3）分配新的内存资源，并复制内容 // 3分**

**int length = strlen(other.m\_data);**

**m\_data = new char[length+1]; // 若能加 NULL 判断则更好**

**strcpy(m\_data, other.m\_data);**

**// （4）返回本对象的引用 // 3分**

**return \*this;**

**}**

# C++/C代码审查表

|  |  |  |
| --- | --- | --- |
| **文件结构** | | |
| 重要性 | 审查项 | 结论 |
|  | 头文件和定义文件的名称是否合理？ |  |
|  | 头文件和定义文件的目录结构是否合理？ |  |
|  | 版权和版本声明是否完整？ |  |
| **重要** | 头文件是否使用了 ifndef/define/endif 预处理块? |  |
|  | 头文件中是否只存放“声明”而不存放“定义” |  |
|  | …… |  |
| **程序的版式** | | |
| 重要性 | 审查项 | 结论 |
|  | 空行是否得体？ |  |
|  | 代码行内的空格是否得体？ |  |
|  | 长行拆分是否得体？ |  |
|  | “{” 和 “}” 是否各占一行并且对齐于同一列？ |  |
| **重要** | 一行代码是否只做一件事？如只定义一个变量，只写一条语句。 |  |
| **重要** | If、for、while、do等语句自占一行，不论执行语句多少都要加“{}”。 |  |
| **重要** | 在定义变量（或参数）时，是否将修饰符 \* 和 ＆ 紧靠变量名？ |  |
|  | 注释是否清晰并且必要？ |  |
| **重要** | 注释是否有错误或者可能导致误解？ |  |
| **重要** | 类结构的public, protected, private顺序是否在所有的程序中保持一致？ |  |
|  | …… |  |
| **命名规则** | | |
| 重要性 | 审查项 | 结论 |
| **重要** | 命名规则是否与所采用的操作系统或开发工具的风格保持一致？ |  |
|  | 标识符是否直观且可以拼读？ |  |
|  | 标识符的长度应当符合“min-length && max-information”原则？ |  |
| **重要** | 程序中是否出现相同的局部变量和全部变量？ |  |
|  | 类名、函数名、变量和参数、常量的书写格式是否遵循一定的规则？ |  |
|  | 静态变量、全局变量、类的成员变量是否加前缀？ |  |
|  | …… |  |
| **表达式与基本语句** | | |
| 重要性 | 审查项 | 结论 |
| **重要** | 如果代码行中的运算符比较多，是否已经用括号清楚地确定表达式的操作顺序？ |  |
|  | 是否编写太复杂或者多用途的复合表达式？ |  |
| **重要** | 是否将复合表达式与“真正的数学表达式”混淆？ |  |
| **重要** | 是否用隐含错误的方式写if语句? 例如  （1）将布尔变量直接与TRUE、FALSE或者1、0进行比较。  （2）将浮点变量用“==”或“！=”与任何数字比较。  （3）将指针变量用“==”或“！=”与NULL比较。 |  |
|  | 如果循环体内存在逻辑判断，并且循环次数很大，是否已经将逻辑判断移到循环体的外面？ |  |
| **重要** | Case语句的结尾是否忘了加break？ |  |
| **重要** | 是否忘记写switch的default分支？ |  |
| **重要** | 使用goto 语句时是否留下隐患? 例如跳过了某些对象的构造、变量的初始化、重要的计算等。 |  |
|  | …… |  |
| **常量** | | |
| 重要性 | 审查项 | 结论 |
|  | 是否使用含义直观的常量来表示那些将在程序中多次出现的数字或字符串？ |  |
|  | 在C++ 程序中，是否用const常量取代宏常量？ |  |
| **重要** | 如果某一常量与其它常量密切相关，是否在定义中包含了这种关系？ |  |
|  | 是否误解了类中的const数据成员？因为const数据成员只在某个对象生存期内是常量，而对于整个类而言却是可变的。 |  |
|  | …… |  |
| **函数设计** | | |
| 重要性 | 审查项 | 结论 |
|  | 参数的书写是否完整？不要贪图省事只写参数的类型而省略参数名字。 |  |
|  | 参数命名、顺序是否合理？ |  |
|  | 参数的个数是否太多？ |  |
|  | 是否使用类型和数目不确定的参数？ |  |
|  | 是否省略了函数返回值的类型？ |  |
|  | 函数名字与返回值类型在语义上是否冲突？ |  |
| **重要** | 是否将正常值和错误标志混在一起返回？正常值应当用输出参数获得，而错误标志用return语句返回。 |  |
| **重要** | 在函数体的“入口处”，是否用assert对参数的有效性进行检查？ |  |
| **重要** | 使用滥用了assert？ 例如混淆非法情况与错误情况，后者是必然存在的并且是一定要作出处理的。 |  |
| **重要** | return语句是否返回指向“栈内存”的“指针”或者“引用”？ |  |
|  | 是否使用const提高函数的健壮性？const可以强制保护函数的参数、返回值，甚至函数的定义体。“Use const whenever you need” |  |
|  | …… |  |
| **内存管理** | | |
| 重要性 | 审查项 | 结论 |
| **重要** | 用malloc或new申请内存之后，是否立即检查指针值是否为NULL？（防止使用指针值为NULL的内存） |  |
| **重要** | 是否忘记为数组和动态内存赋初值？（防止将未被初始化的内存作为右值使用） |  |
| **重要** | 数组或指针的下标是否越界？ |  |
| **重要** | 动态内存的申请与释放是否配对？（防止内存泄漏） |  |
| **重要** | 是否有效地处理了“内存耗尽”问题？ |  |
| **重要** | 是否修改“指向常量的指针”的内容？ |  |
| **重要** | 是否出现野指针？例如  （1）指针变量没有被初始化。  （2）用free或delete释放了内存之后，忘记将指针设置为NULL。 |  |
| **重要** | 是否将malloc/free 和 new/delete 混淆使用？ |  |
| **重要** | malloc语句是否正确无误？例如字节数是否正确？类型转换是否正确？ |  |
| **重要** | 在创建与释放动态对象数组时，new/delete的语句是否正确无误？ |  |
|  | …… |  |
| **C++ 函数的高级特性** | | |
| 重要性 | 审查项 | 结论 |
|  | 重载函数是否有二义性？ |  |
| **重要** | 是否混淆了成员函数的重载、覆盖与隐藏？ |  |
|  | 运算符的重载是否符合制定的编程规范？ |  |
|  | 是否滥用内联函数？例如函数体内的代码比较长，函数体内出现循环。 |  |
| **重要** | 是否用内联函数取代了宏代码？ |  |
|  | …… |  |
| **类的构造函数、析构函数和赋值函数** | | |
| 重要性 | 审查项 | 结论 |
| **重要** | 是否违背编程规范而让C++ 编译器自动为类产生四个缺省的函数：（1）缺省的无参数构造函数；（2）缺省的拷贝构造函数；（3）缺省的析构函数；（4）缺省的赋值函数。 |  |
| **重要** | 构造函数中是否遗漏了某些初始化工作？ |  |
| **重要** | 是否正确地使用构造函数的初始化表？ |  |
| **重要** | 析构函数中是否遗漏了某些清除工作？ |  |
|  | 是否错写、错用了拷贝构造函数和赋值函数？ |  |
| **重要** | 赋值函数一般分四个步骤：（1）检查自赋值；（2）释放原有内存资源；（3）分配新的内存资源，并复制内容；（4）返回 \*this。是否遗漏了重要步骤？ |  |
| **重要** | 是否正确地编写了派生类的构造函数、析构函数、赋值函数？注意事项：  （1）派生类不可能继承基类的构造函数、析构函数、赋值函数。  （2）派生类的构造函数应在其初始化表里调用基类的构造函数。  （3）基类与派生类的析构函数应该为虚（即加virtual关键字）。  （4）在编写派生类的赋值函数时，注意不要忘记对基类的数据成员重新赋值。 |  |
|  | …… |  |
| **类的高级特性** | | |
| 重要性 | 审查项 | 结论 |
| **重要** | 是否违背了继承和组合的规则？  （1）若在逻辑上B是A的“一种”，并且A的所有功能和属性对B而言都有意义，则允许B继承A的功能和属性。  （2）若在逻辑上A是B的“一部分”（a part of），则不允许B从A派生，而是要用A和其它东西组合出B。 |  |
|  | …… |  |
| **其它常见问题** | | |
| 重要性 | 审查项 | 结论 |
| **重要** | 数据类型问题：  （１）变量的数据类型有错误吗？  （２）存在不同数据类型的赋值吗？  （３）存在不同数据类型的比较吗？ |  |
| **重要** | 变量值问题：  （１）变量的初始化或缺省值有错误吗？  （２）变量发生上溢或下溢吗？  （３）变量的精度够吗？ |  |
| **重要** | 逻辑判断问题：  （１）由于精度原因导致比较无效吗？  （２）表达式中的优先级有误吗？  （３）逻辑判断结果颠倒吗？ |  |
| **重要** | 循环问题：  （１）循环终止条件不正确吗？  （２）无法正常终止（死循环）吗？  （３）错误地修改循环变量吗？  （４）存在误差累积吗？ |  |
| **重要** | 错误处理问题：  （１）忘记进行错误处理吗？  （２）错误处理程序块一直没有机会被运行？  （３）错误处理程序块本身就有毛病吗？如报告的错误与实际错误不一致，处理方式不正确等等。  （４）错误处理程序块是“马后炮”吗？如在被它被调用之前软件已经出错。 |  |
| **重要** | 文件I/O问题：  （１）对不存在的或者错误的文件进行操作吗？  （２）文件以不正确的方式打开吗？  （３）文件结束判断不正确吗？  （４）没有正确地关闭文件吗？ |  |

1. 在web页面中不论显示器的分辨率的大小是800×600，还是1024×768使得页面都显示在屏幕中间。
2. 请利用正则表达式验证一个字符串是否为数字串

（请用csharp）。

1. 请用HTML在客户端画出一个5行4列的表格，并用Javascript访问表格第1行的2列的内容
2. 请用javascript脚本实现弹出无边框窗口显示a.htm页
3. 运用ASP.NET的输出缓冲来存储动态页面
4. 本地计算机上有SQL 2000登录密码式 password 请用CSharp 创建与SQL的连接并把Northwind数据库中的sysusers表中的数据读出放入Table数据集。
5. 请用csharp设计一个算法，实现将数字转化为字符。

如（0－A，1－B，26－AA）。

1. Session的作用是什么，当不能使用Session会话时最可能出现的原因是什么？
2. Application 和 Session 两种会话有什么不同。Application会话自身具有那些属性？
3. 请简述一下你对WebService的了解。
4. .net的机制是什么?
5. 请说出ASP，ASP.NET的区别
6. 说出下面几个函数的区别:

private void test(string str){…}

private void test(ref string str){…}

private void test(out string str){…}

如果一个页面与数据

1、描述一下你对dot net 的理解（越详细越好，最好有图形描述）

2、Java Script 与 Java 的区别？

3、Java Script 类型与dot net 类型的区别

4、请列举你所知道的文件上传实现方式

5、dot net 组件如何与JavaScript脚本交互；dot net 组件如何调用JavaScript函数；JavaScript如何调用dot net 公共属性和事件？

1. ASP.net 可以在哪些系统下运行
2. Aspx文件可以用一种以上的语言吗？
3. Asp.net可以有多个Form吗？
4. Dataset和ADO的记录集有什么区别 有什么特点，有什么不足之处

5. ADO和ADO.net 有什么不同？ADO.net 是ADO的后续版本吗？

* 1. Dataset,Datatable,Datareder,DataADapter有什么联系？
  2. C#支持多继承吗？假设有一个人类，有名字和年龄2个属性，用c#完成这个类,C#比vb.net的效率更高吗?
  3. 写一个函数,当参数N很大的时候,计算下列算式的结果:1-2+3-4+5-6+7-8+…n

9、asp.net的命名空间有哪些?

**10、用程序打印出哈希表**

1. 你平时会看哪些方面的书籍？喜欢浏览哪些网站？
2. 当你遇到问题时通常如何解决？
3. 谈谈你对.Net与Java的看法
4. 你对公司要求加班有什么样的看法？
5. 简述你对XML Web Service的原理的认识？
6. 说出你所了解的数据库访问组件(例如ADO，至少4种)
7. 对单表进行模糊查询，并对其中两列分别进行升序和降序排列(在一个Select语句中)
8. 根据上题中的表编写一个带输入参数的存储过程，输入参数的数据类型为varchar。

例如：输入参数为“20060912”，返回表中所有日期字段大于该日期的记录

9．用VB或VB.NET或ASP.NET编程，从一个表中选出两个varchar列(其中一列数据不能有重复，并且长度等于4)，并将两列连接成一个字符串，将这些字符串动态加载到TreeView1所有节点的Text属性中

一、写出相应的sql语句：

①创建一个数据库testDataBase；

②创建一个表testTable，有两个字段A和B，类型分别为int，长度为20的可变类型；

③插入一列（10，“数据库”）；

④查询testTable表，返回所有信息；

⑤查询testTable表，返回信息的行数；

⑥删除testTable表，并记录在日志文件里；

⑦删除testTable表，不记录在日志文件里；

⑧删除testDataBase数据库；

二、用C#或VB或Delph中的一种语言描述下列算法：

给定一个字符串数组，对字符串数组按照由小到大的顺序进行排序，并把重复的字符删去。

1. 用C#或者JAVA写一个MAIN函数
2. 面向对象的几种方式，你在实际应用中用到了哪些，请描述
3. 写出asp.net或者jsp写出动态生成网页的过程,并说出其原理
4. 写出n-tie的结构和每一层的作用
5. 设计数据库需要考虑的问题
6. 什么是重构,以及它的表现方式
7. 接口的定义,它在内存中如何实现
8. 面向对象的思想主要包括什么？
9. 什么叫应用程序域？什么是受管制的代码？什么是强类型系统？什么是装箱和拆箱？什么是重载？CTS、CLS和CLR分别作何解释？
10. C#中的接口和类有什么异同？
11. ASP.net的身份验证方式有哪些？分别是什么原理？
12. 什么是SOAP,有哪些应用。
13. 如何部署一个ASP.net页面。
14. 如何理解.net中的垃圾回收机制。
15. 常用的调用webservice方法有哪些？
16. 什么是code-Behind技术？

10、在什么情况下会用到虚方法？它与接口有什么不同？

1. .net中读写XML的类都归属于哪些命名空间？
2. 解释一下UDDI、WSDL的意义及其作用?
3. 什么是SOAP,有哪些应用?

4．计算水费：(w1<w2<w3)  
 用水量在w1(吨)之内的部分按p1(元/吨)收取，超过w1并在w2之内的部分按p2收取，超出w2部分按p3收取。

请写出表达式求出用水量为sum时应收取的水费(total)？

提示：利用函数IFF(exp1,exp2,exp3), 当表达式exp1的值为true时该函数返回exp2的 值，否则返回exp3的值；该函数可以嵌套使用。total=?

5．用.net写出将字符串加载到xml文档的语句。

6．说明通过get和post提交页面的区别?

1. .net中读写XML的类都归属于哪些命名空间？
2. 解释一下UDDI、WSDL的意义及其作用?
3. 什么是SOAP,有哪些应用?

4．计算水费：(w1<w2<w3)  
 用水量在w1(吨)之内的部分按p1(元/吨)收取，超过w1并在w2之内的部分按p2收取，超出w2部分按p3收取。

请写出表达式求出用水量为sum时应收取的水费(total)？

提示：利用函数IFF(exp1,exp2,exp3), 当表达式exp1的值为true时该函数返回exp2的 值，否则返回exp3的值；该函数可以嵌套使用。total=?

5．用.net写出将字符串加载到xml文档的语句。

6．说明通过get和post提交页面的区别?

1.  填空:

(1)面向对象的语言具有\_\_\_\_\_\_\_\_性、\_\_\_\_\_\_\_\_\_性、\_\_\_\_\_\_\_\_性。

        (2)能用foreach遍历访问的对象需要实现 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_接口或声明\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_方法的类型。

        (3)列举ADO.net中的五个主要对象\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_、\_\_\_\_\_\_\_\_\_\_\_\_\_、\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_、\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_、\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_。

2. 不定项选择：

(1) 以下叙述正确的是：

      A. 接口中可以有虚方法。     B. 一个类可以实现多个接口。  
      C. 接口不能被实例化。       D. 接口中可以包含已实现的方法。

  (2) 从数据库读取记录，你可能用到的方法有：

      A. ExecuteNonQuery            B. ExecuteScalar

      C. Fill                        D. ExecuteReader

3. 简述 private、 protected、 public、 internal 修饰符的访问权限。

4. 写出一条Sql语句： 取出表A中第31到第40记录（SQLServer, 以自动增长的ID作为主键,  注意：ID可能不是连续的。）

5 .列举ASP.NET 页面之间传递值的几种方式。

6. 写出程序的输出结果

class Class1  {  
       private string str = "Class1.str";  
       private int i = 0;  
       static void StringConvert(string str)  {  
           str = "string being converted.";  
       }  
       static void StringConvert(Class1 c)  {  
           c.str = "string being converted.";  
       }  
       static void Add(int i)  {  
           i++;  
       }  
       static void AddWithRef(ref int i)  {  
           i++;  
       }  
       static void Main()  {  
           int i1 = 10;  
           int i2 = 20;  
           string str = "str";  
           Class1 c = new Class1();  
           Add(i1);  
           AddWithRef(ref i2);  
           Add(c.i);  
           StringConvert(str);           
           StringConvert(c);  
           Console.WriteLine(i1);  
           Console.WriteLine(i2);  
           Console.WriteLine(c.i);  
           Console.WriteLine(str);  
           Console.WriteLine(c.str);  
       }    
    }

7.写出程序的输出结果

public abstract class A    
{  
        public A()    
       {  
            Console.WriteLine('A');  
        }  
        public virtual void Fun()    
       {  
            Console.WriteLine("A.Fun()");  
        }  
}

public class B: A    
{  
        public B()   
       {  
            Console.WriteLine('B');  
        }

        public new void Fun()    
       {  
            Console.WriteLine("B.Fun()");  
        }

        public static void Main()    
       {  
           A a = new B();  
           a.Fun();  
        }  
}

8.      写出程序的输出结果：  
public class A    
{  
        public virtual void Fun1(int i)   
       {  
            Console.WriteLine(i);  
        }

        public void Fun2(A a)     
       {  
            a.Fun1(1);  
            Fun1(5);  
        }  
}

public class B : A    
{  
        public override void Fun1(int i)      
       {  
            base.Fun1 (i + 1);  
        }

        public static void Main()     
       {  
            B b = new B();  
            A a = new A();  
            a.Fun2(b);  
            b.Fun2(a);            
        }  
}

9. 一列数的规则如下: 1、1、2、3、5、8、13、21、34......

   求第30位数是多少， 用递归算法实现。(C#语言)

10.  程序设计: 猫大叫一声，所有的老鼠都开始逃跑，主人被惊醒。(C#语言)

要求：  1.要有联动性，老鼠和主人的行为是被动的。

2.考虑可扩展性，猫的叫声可能引起其他联动效应。

一.填空题

1.c#中的三元运算符是\_\_?:\_\_\_?

2.当整数a赋值给一个object对象时，整数a将会被\_\_装箱(封装)\_\_\_?

3.类成员有\_\_\_\_\_种可访问形式？  this.;new Class().Method;

4.public static const int A=1;这段代码有错误么？是什么？  const不能用static修饰

5.float f=-123.567F;

  int i=(int)f;

i的值现在是\_\_\_\_\_?   123

6.利用operator声明且仅声明了“==”，有什么错误么?

7.委托声明的关键字是\_\_\_\_\_\_? delagete

8.用sealed修饰的类有什么特点？密封，不能继承

9.在Asp.net中所有的自定义用户控件都必须继承自\_\_\_\_\_\_\_\_?Control

10.在.Net中所有可序列化的类都被标记为\_\_\_\_\_?

11.在.Net托管代码中我们不用担心内存漏洞，这是因为有了\_\_\_\_\_\_?gc

12.下面的代码中有什么错误吗？\_\_\_\_\_\_\_

     using System;

     class A

     {

          public virtual void F(){

              Console.WriteLine("A.F");

           }

      }

      abstract class B:A

       {

           public abstract override void F(); // new public abstract void F();

       }

13.当类T只声明了私有实例构造函数时，则在T的程序文本外部，\_\_\_可以\_\_\_（可以 or 不可以）从T派生出新的类，不可以\_\_\_\_（可以 or 不可以）直接创建T的任何实例。

14.下面这段代码有错误么？

    switch (i){

    case():

        CaseZero();

        break;

    case 1:

        CaseOne();

        break;

    case 2:

        dufault;  //wrong

        CaseTwo();

        break;

   }

15.在.Net中，类System.Web.UI.Page 可以被继承么？可以

二.简答题

1.在c#中using和new这两个关键字有什么意义，请写出你所知道的意义？using 指令 和语句 new 创建实例 new 隐藏基类中方法

4.谈谈类和结构的区别？类是引用类型、结构是值类型

5.一个长度为10000的字符串，通过随机从a-z中抽取10000个字符组成。请用c＃语言编写主要程序来实现。

6.对于这样的一个枚举类型：

      enum Color:byte{

          Red,

          Green,

          Blue,

          Orange

       }

     string[] ss=Enum.GetNames(typeof(Color));

     byte[]   bb=Enum.GetValues(typeof(Color));

试写一段程序显示出枚举类型中定义的所有符号名称以及它们对应的数值。

7.您了解设计模式么？请列出您所知道的设计模式的名称。 //

8.请在SQL Server中设计表来保存一个树状结构的组织结构图（假设结构图中只有名称这一项内容需要保存），如果我想查询某一职位下的所有职位，用一个存储过程来实现，你有什么思路？

9.什么叫做SQL注入，如何防止？请举例说明。

10.下面这段代码输出什么？为什么？

     int i=5;

     int j=5;

     if (Object.ReferenceEquals(i,j))

          Console.WriteLine("Equal");

     else

          Console.WriteLine("Not Equal");

//不相等，因为比较的是对象

1.写一个实现对一段字符串翻转的方法，附加一些条件，如其中包括“，”、“.”，对其设计测试用 例 。

2.对一支纸杯设计测试用例（可以是广义的杯，不一定是某一支特定功能的杯）

开发语言概念题

3.什么是反射？

4.用Singleton如何写设计模式

5.C#中的垃圾回收机制是怎样的？

6.什么是Application Pool？

7.链表和数组的区别，各有什么优缺点.

8.Remoting在客户端服务器怎么实现 ?

9.什么是友元函数？

10.用标准C如何实现多态？

11.什么是虚函数？

12.什么是抽象函数？

13.什么是内存泄漏，怎样最简单的方法判断被存泄漏 ?

1、在C＃中，string str = null 与 string str = “” 请尽量使用文字或图象说明其中的区别。

２、请详述在dotnet中类(class)与结构(struct)的异同：

3、根据委托(delegate)的知识，请完成以下用户控件中代码片段的填写：

namespace test

{

public delegate void OnDBOperate();

public class UserControlBase : System.Windows.Forms.UserControl

{

public event OnDBOperate OnNew;

privatevoidtoolBar\_ButtonClick(objectsender,System.Windows.Forms.ToolBarButtonClickEventArgs e)

{

if(e.Button.Equals(BtnNew))

{

//请在以下补齐代码用来调用OnDBOperate委托签名的OnNew事件。

}

}

}

4、分析以下代码，完成填空

string strTmp = "abcdefg某某某";

int i= System.Text.Encoding.Default.GetBytes(strTmp).Length;

int j= strTmp.Length;

以上代码执行完后，i= j=

5、SQLSERVER服务器中，给定表 table1 中有两个字段 ID、LastUpdateDate，ID表示更新的事务号， LastUpdateDate表示更新时的服务器时间，请使用一句SQL语句获得最后更新的事务号。

6、根据线程安全的相关知识，分析以下代码，当调用test方法时i>10时是否会引起死锁?并简要说明理由。

public void test(int i)

{

lock(this)

{

if (i>10)

{

i--;

test(i);

}

}

}

7、分析以下代码。

public static void test(string ConnectString)

{

System.Data.OleDb.OleDbConnection conn = new System.Data.OleDb.OleDbConnection();

conn.ConnectionString = ConnectString;

try

{

conn.Open();

…….

}catch(Exception Ex)

{

MessageBox.Show(Ex.ToString());

}finally

{

if (!conn.State.Equals(ConnectionState.Closed))

conn.Close();

}

请问

1)以上代码可以正确使用连接池吗？

2)以上代码所使用的异常处理方法，是否所有在test方法内的异常都可以被捕捉并显示出来？

8、简要谈一下您对微软.NET 构架下remoting和webservice两项技术的理解以及实际中的应用。

9、描绘开发WEB项目时所遵从的基本瀑布模型。

10、IIS日志文件在服务器上的默认保存位置。

11、Tag（标签）是一种日志分类方式，以关键字的形式标识在博客中，用来快速的导向到用户想查找的热点文章中。

一、选择题

　　1. SQL语言是（ ）的语言，容易学习 。

　　A.过程化 B. 非过程化

　　C.格式化 D. 导航式

2. SQL语言的数据操纵语句包括SELECT、INSERT、UPDATE、DELETE等。其中最重要的，也是使用最频繁的语句是（ ） 。

　　A. SELECT B. INSERT

　　C. UPDATE D. DELETE

3. 在视图上不能完成的操作是（ ） 。

　　A. 更新视图 B. 查询

　　C. 在视图上定义新的表 D. 在视图上定义新的视图

4. SQL语言集数据查询、数据操纵、数据定义和数据控制功能于一体，其中，CREATE、DROP、ALTER语句是实现哪种功能（ ）。

　　A. 数据查询 B. 数据操纵

　　C. 数据定义 D. 数据控制

5. SQL语言中，删除一个视图的命令是（ ）。

　　A.DELETE　　B.DROP　　C.CLEAR　　D.REMOVE  
　　6. 在SQL语言中的视图VIEW是数据库的（ ） 。

　　A. 外模式 B. 模式 C. 内模式 D. 存储模式

7. 下列的SQL语句中，（ ）不是数据定义语句。

　　A. CREATE TABLE B. DROP VIEW

　　 C. CREATE VIEW D. GRANT

8. 若要撤销数据库中已经存在的表S，可用（ ）。

　　A. DELETE TABLE S B. DELETE S

　　C. DROP TABLE S D. DROP S

9. 若要在基本表S中增加一列CN（课程名），可用（ ）。

　 A.ADD TABLE S（CN CHAR（８））

　　 B.ADD TABLE S ALTER（CN CHAR（８））

　　 C.ALTER TABLE S ADD（CN CHAR（８））

　　 D.ALTER TABLE S （ADD CN CHAR（８））

10. 学生关系模式 S（ S＃，Sname，Sex，Age），S的属性分别表示学生的学号、姓名、性别、年龄。要在表S中删除一个属性“年龄”，可选用的SQL语句是（ ）。

　　A. DELETE Age from S 　　 B. ALTER TABLE S DROP Age

　　C. UPDATE S Age 　　D. ALTER TABLE S ‘Age’

11. 有关系S（S＃，SNAME，SAGE），C（C＃，CNAME），SC（S＃，C＃，GRADE）。其中S＃是学生号，SNAME是学生姓名，SAGE是学生年龄， C＃是课程号，CNAME是课程名称。要查询选修“ACCESS”课的年龄不小于20的全体学生姓名的SQL语句是SELECT SNAME FROM S，C，SC WHERE子句。这里的WHERE子句的内容是（ ）。

　 　A. S.S# = SC.S# and C.C# = SC.C# and SAGE>=20 and CNAME=‘ACCESS’

　　 B. S.S# = SC.S# and C.C# = SC.C# and SAGE in>=20 and CNAME in ‘ACCESS’

　　 C. SAGE in>=20 and CNAME in ‘ACCESS’

　　 D. SAGE>=20 and CNAME=’ ACCESS’

12. 设关系数据库中一个表S的结构为S（SN，CN，grade），其中SN为学生名，CN为课程名，二者均为字符型；grade为成绩，数值型，取值范围0－100。若要把“张二的化学成绩80分”插入S中，则可用（ ）。

　　A. ADD

　　 INTO S

　　 VALUES（’张二’，’化学’，’80’）

　　B. INSERT

　　 INTO S

　　 VALUES（’张二’，’化学’，’80’）

　　C. ADD

　　 INTO S

　　 VALUES（’张二’，’化学’，80）

　　D. INSERT

　　 INTO S

　　 VALUES（’张二’，’化学’，80）

13. 设关系数据库中一个表S的结构为：S（SN，CN，grade），其中SN为学生名，CN为课程名，二者均为字符型；grade为成绩，数值型，取值范围0－100。若要更正王二的化学成绩为85分，则可用（ ） 。

　　A. UPDATE S

　　 SET grade＝85

　　 WHERE SN＝’王二’ AND CN＝’化学’

　　B. UPDATE S

　　 SET grade＝’85’

　　 WHERE SN＝’王二’ AND CN＝’化学’

　　C. UPDATE grade＝85

　　 WHERE SN＝’王二’ AND CN＝’化学’

　　D. UPDATE grade＝’85’

　　 WHERE SN＝’王二’ AND CN＝’化学’

14. 在SQL语言中，子查询是（ ） 。

　　A. 返回单表中数据子集的查询语言

　　B. 选取多表中字段子集的查询语句

　　C. 选取单表中字段子集的查询语句

　　D. 嵌入到另一个查询语句之中的查询语句

15. SQL是一种（ ）语言。

　　A. 高级算法 B. 人工智能

　　C. 关系数据库 D. 函数型

16. 有关系S（S＃，SNAME，SEX），C（C＃，CNAME），SC（S＃，C＃，GRADE）。其中S＃是学生号，SNAME是学生姓名，SEX是性别， C＃是课程号，CNAME是课程名称。要查询选修“数据库”课的全体男生姓名的SQL语句是SELECT SNAME FROM S，C，SC WHERE子句。这里的WHERE子句的内容是（ ）。

　　A.S.S# = SC.S# and C.C# = SC.C# and SEX=’男’ and CNAME=’数据库’

　　B.S.S# = SC.S# and C.C# = SC.C# and SEX in’男’and CNAME in’数据库’

　　C.SEX ’男’ and CNAME ’ 数据库’

　　D.S.SEX=’男’ and CNAME=’ 数据库’

17. 若用如下的SQL语句创建了一个表SC：

　　CREATE TABLE SC （S# CHAR（6） NOT NULL，C# CHAR（3） NOT NULL，SCORE INTEGER，NOTE CHAR（20））；向SC表插入如下行时，（ ）行可以被插入 。

　　A.（’201009’，’111’，60，必修）　　B.（’200823’，’101’，NULL，NULL）

　　C.（NULL，’103’，80，’选修’）　D.（’201132’，NULL，86，’ ’）

18. 假设学生关系S（S＃，SNAME，SEX），课程关系C（C＃，CNAME），学生选课关系SC（S＃，C＃，GRADE）。要查询选修“Computer”课的男生姓名，将涉及到关系（ ）。

　　A. S B. S，SC C. C，SC D. S，C，SC

二、简答题

1. 什么是基本表？什么是视图？两者的区别和联系是什么？

2. 所有的视图是否都可以更新？为什么？

3. 哪类视图是可以更新的，哪类视图是不可更新的？ 各举一例说明。

4. 试述某个你熟悉的实际系统中对视图更新的规定。

.**net笔试题**

姓名：                      日期：

1.      填空: (1)面向对象的语言具有\_\_\_\_\_\_\_\_性、\_\_\_\_\_\_\_\_\_性、\_\_\_\_\_\_\_\_性。

        (2)能用foreach遍历访问的对象需要实现 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_接口或声明\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_方法的类型。

        (3)列举ADO.net中的五个主要对象\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_、\_\_\_\_\_\_\_\_\_\_\_\_\_、\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_、\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_、\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_。

2. 不定项选择：

(1) 以下叙述正确的是：

      A. 接口中可以有虚方法。     B. 一个类可以实现多个接口。  
      C. 接口不能被实例化。       D. 接口中可以包含已实现的方法。

   (2) 从数据库读取记录，你可能用到的方法有：

      A. ExecuteNonQuery            B. ExecuteScalar

      C. Fill                        D. ExecuteReader

3. 简述 private、 protected、 public、 internal 修饰符的访问权限。

4. 写出一条Sql语句： 取出表A中第31到第40记录（SQLServer, 以自动增长的ID作为主键,  注意：ID可能不是连续的。）

5 .列举ASP.NET 页面之间传递值的几种方式。

6. 写出程序的输出结果

class Class1  {  
       private string str = "Class1.str";  
       private int i = 0;  
       static void StringConvert(string str)  {  
           str = "string being converted.";  
       }  
       static void StringConvert(Class1 c)  {  
           c.str = "string being converted.";  
       }  
       static void Add(int i)  {  
           i++;  
       }  
       static void AddWithRef(ref int i)  {  
           i++;  
       }  
       static void Main()  {  
           int i1 = 10;  
           int i2 = 20;  
           string str = "str";  
           Class1 c = new Class1();  
           Add(i1);  
           AddWithRef(ref i2);  
           Add(c.i);  
           StringConvert(str);           
           StringConvert(c);  
           Console.WriteLine(i1);  
           Console.WriteLine(i2);  
           Console.WriteLine(c.i);  
           Console.WriteLine(str);  
           Console.WriteLine(c.str);  
       }    
    }

7.写出程序的输出结果

public abstract class A    
{  
        public A()    
       {  
            Console.WriteLine('A');  
        }  
        public virtual void Fun()    
       {  
            Console.WriteLine("A.Fun()");  
        }  
}

public class B: A    
{  
        public B()   
       {  
            Console.WriteLine('B');  
        }

        public new void Fun()    
       {  
            Console.WriteLine("B.Fun()");  
        }

        public static void Main()    
       {  
           A a = new B();  
           a.Fun();  
        }  
}

8.      写出程序的输出结果：  
public class A    
{  
        public virtual void Fun1(int i)   
       {  
            Console.WriteLine(i);  
        }

        public void Fun2(A a)     
       {  
            a.Fun1(1);  
            Fun1(5);  
        }  
}

public class B : A    
{  
        public override void Fun1(int i)      
       {  
            base.Fun1 (i + 1);  
        }

        public static void Main()     
       {  
            B b = new B();  
            A a = new A();  
            a.Fun2(b);  
            b.Fun2(a);            
        }  
}

9. 一列数的规则如下: 1、1、2、3、5、8、13、21、34......

   求第30位数是多少， 用递归算法实现。(C#语言)

10.  程序设计: 猫大叫一声，所有的老鼠都开始逃跑，主人被惊醒。(C#语言)

要求：  1.要有联动性，老鼠和主人的行为是被动的。

2.考虑可扩展性，猫的叫声可能引起其他联动效应。

参考答案:  
1. (1)  继承性、封装性、多态性。（考基本概念）  
   (2)  IEnumerable 、 GetEnumerator (对foreach机制的理解，本来不想出这题的，凑分)  
   (3)  ... (送分题, 对ADO.net的了解)  
评分标准：一空1分，满分10分。

2. (1) B、C (考对接口的理解)  (2)  B、C、D (考查对ADO.net的熟练程度)  
  评分标准: 一题5分，不选或者错选均不得分。漏选得2分。满分10分。

3. . private :   私有成员, 在类的内部才可以访问。  
     protected : 保护成员，该类内部和继承类中可以访问。  
     public :    公共成员，完全公开，没有访问限制。  
     internal:   在同一命名空间内可以访问。  
评分标准：答对1题2分，2题5分，3题7分。全对10分。 (送分题)

4.  解1:  select top 10 \* from A where id not in (select top 30 id from A)  
    解2:  select top 10 \* from A where id > (select max(id) from (select top 30 id from A )as A)  
    评分标准： 写对即10分。(答案不唯一，datagrid 分页可能需要用到)  
   
5.  1.使用QueryString,  如....?id=1; response. Redirect()....  
    2.使用Session变量  
    3.使用Server.Transfer  
    ....等等  
   评分标准： 答对1点得3分， 两点7分， 3点10分。

6.  (考查值引用和对象引用)  
10  
21  
0  
str  
string being converted.  
评分标准：答对一点得2分，满分10分。

7.  A  
    B  
A.Fun()  
评分标准： 写出A.B 得5分，写出A.Fun()得5分，满分10分。  
(考查在继承类中构造函数, 以及new 方法, )

8.  2  
    5  
    1  
    6

评分标准： 答对一点得2分，两点得5分，3点得7分。全对得10分。

(一些人做这题，头都晕了.... ^\_^ )

9.

    public class MainClass  
    {  
        public static void Main()     
        {  
            Console.WriteLine(Foo(30));  
        }  
        public static int Foo(int i)  
        {  
            if (i <= 0)  
                return 0;  
            else if(i > 0 && i <= 2)  
                return 1;  
            else return Foo(i -1) + Foo(i - 2);  
        }  
    }  
评分标准： 写出return Foo(i -1) + Foo(i - 2); 得5分。

 写出if(i > 0 && i <= 2) return 1; 得5分。

           方法参数过多需要扣分（扣除分数 = 参数个数 － 1）

           不用递归算法扣5分

           (递归算法在树结构建立等方面比较常用)

10

 要点：1. 联动效果，运行代码只要执行Cat.Cryed()方法。2. 对老鼠和主人进行抽象  
评分标准: <1>.构造出Cat、Mouse、Master三个类，并能使程序运行(2分)  
            <2>从Mouse和Master中提取抽象（5分）  
            <3>联动效应，只要执行Cat.Cryed()就可以使老鼠逃跑，主人惊醒。(3分)

    public interface Observer  
    {  
        void Response();    //观察者的响应，如是老鼠见到猫的反映  
    }  
    public interface Subject  
    {  
        void AimAt(Observer obs);  //针对哪些观察者，这里指猫的要扑捉的对象---老鼠  
    }  
    public class Mouse : Observer  
    {  
        private string name;  
        public Mouse(string name, Subject subj)  
        {             
            this.name = name;  
            subj.AimAt(this);  
        }  
          
        public void Response()  
        {  
            Console.WriteLine(name + " attempt to escape!");  
        }  
    }  
    public class Master : Observer  
    {     
        public Master(Subject subj)  
        {             
            subj.AimAt(this);  
        }  
          
        public void Response()  
        {  
            Console.WriteLine("Host waken!");  
        }    
    }  
   
    public class Cat : Subject  
    {  
        private ArrayList observers;  
        public Cat()  
        {     
            this.observers = new ArrayList();  
        }  
        public void AimAt(Observer obs)  
        {  
            this.observers.Add(obs);  
        }  
        public void Cry()  
        {  
            Console.WriteLine("Cat cryed!");  
            foreach (Observer obs in this.observers)  
            {  
                obs.Response();  
            }  
        }  
    }  
    class MainClass  
    {         
        static void Main(string[] args)  
        {  
            Cat cat = new Cat();  
            Mouse mouse1 = new Mouse("mouse1", cat);  
            Mouse mouse2 = new Mouse("mouse2", cat);  
            Master master = new Master(cat);  
            cat.Cry();  
        }  
    }

//---------------------------------------------------------------------------------------------

设计方法二: 使用event -- delegate设计..  
    public delegate void SubEventHandler();  
    public abstract class Subject  
    {  
        public event SubEventHandler SubEvent;  
        protected void FireAway()  
        {  
            if (this.SubEvent != null)  
                this.SubEvent();  
        }     
    }  
    public class Cat : Subject  
    {    
        public void Cry()  
        {  
            Console.WriteLine("cat cryed.");  
            this.FireAway();  
        }  
    }  
    public abstract class Observer  
    {  
        public Observer(Subject sub)  
        {  
            sub.SubEvent += new SubEventHandler(Response);  
        }  
        public abstract void Response();      
    }  
    public class Mouse : Observer  
    {  
        private string name;  
        public Mouse(string name, Subject sub) : base(sub)  
        {     
            this.name = name;  
        }  
        public override void Response()  
        {  
            Console.WriteLine(name + " attempt to escape!");  
        }  
    }  
    public class Master : Observer  
    {  
        public Master(Subject sub) : base(sub){}  
        public override void Response()  
        {  
            Console.WriteLine("host waken");  
        }  
    }  
    class Class1  
    {  
        static void Main(string[] args)  
        {  
            Cat cat = new Cat();  
            Mouse mouse1 = new Mouse("mouse1", cat);  
            Mouse mouse2 = new Mouse("mouse2", cat);  
            Master master = new Master(cat);  
            cat.Cry();  
        }

    }

第二十二，接口是否可继承接口? 抽象类是否可实现(implements)接口? 抽象类是否可继承实体类(concrete class)?

第二十三，启动一个线程是用run()还是start()?

第二十四，构造器Constructor是否可被override?

第二十五，是否可以继承String类?

第二十六，当一个线程进入一个对象的一个synchronized方法后，其它线程是否可进入此对象的其它方法?

第二十七，try {}里有一个return语句，那么紧跟在这个try后的finally {}里的code会不会被执行，什么时候被执行，在return前还是后?

第二十八，编程题: 用最有效率的方法算出2乘以8等於几?

第二十九，两个对象值相同(x.equals(y) == true)，但却可有不同的hash code，这句话对不对?

第三十，当一个对象被当作参数传递到一个方法后，此方法可改变这个对象的属性，并可返回变化后的结果，那么这里到底是值传递还是引用传递?

第三十一，swtich是否能作用在byte上，是否能作用在long上，是否能作用在String上?

第三十二，编程题: 写一个Singleton出来。

数据库方面：

1.存储过程和函数的区别

2.事务是什么？

3.游标的作用？如何知道游标已经到了最后？

4.触发器分为事前触发和事后触发，这两种触发有和区别。语句级触发和行级触发有何区别。

1。用C＃实现以下功能

   a 产生一个int数组，长度为100，并向其中随机插入1-100，并且不能重复。

   b 对上面生成的数组排序，需要支持升序、降序两种顺序

2。请说明在.net中常用的几种页面间传递参数的方法，并说出他们的优缺点。

3。请说明.net中的错误处理机制，并举例

4。请说出强名的含义

5。请列出c＃中几种循环的方法，并指出他们的不同

6。请指出.net中所有类型的基类

7。请指出GAC的含义

8。SQL SREVER中，向一个表中插入了新数据，如何快捷的得到自增量字段的当前值

您在什么情况下会用到虚方法？它与接口有什么不同？

Q：Override与重载有什么区别？

Q：值类型与引用类型有什么区别？

Q：怎样理解静态变量？

Q：向服务器发送请求有几种方式？

Q：DataReader与Dataset有什么区别？

Q：如果在一个B/S结构的系统中需要传递变量值，但是又不能使用Session、Cookie、Application，您有几种方法进行处理？

Q：用.net做B/S结构的系统，您是用几层结构来开发，每一层之间的关系以及为什么要这样分层？

Q：软件开发过程一般有几个阶段？每个阶段的作用？

Q：微软推出了一系列的Application Block，请举出您所知道的Application Block并说明其作用？

Q：请列举一些您用到过的设计模式以及在什么情况下使用该模式？

Q：您对WebService的体会？

Q：您对编程的兴趣如何？工作中遇到不懂的问题是怎样去解决的？您一般怎样去提高自己的编程水平？

Q：您离职的原因是什么？

Q：通过超链接怎样传递中文参数？

Q：请编程遍历页面上所有TextBox控件并给它赋值为string.Empty？

Q：请编程实现一个冒泡排序算法？

1、override与重载的区别

2、.net的错误处理机制是什么

3、C＃中接口和类的异同

4、DataReader和DataSet的异同

1.有哪几种方法可以实现一个类存取另外一个类的成员函数及属性,并请举列来加以说明和分析.

2.如果需记录类的实例个数,该如何实现,请写一个简单的类于以证明.

3.A类是B类的基类,并且都有自己的构造,析构函数,请举例证明B类从实例化到消亡过程中构造,析构函数的执行过程.

4.需要实现对一个字符串的处理,首先将该字符串首尾的空格去掉,如果字符串中间还有连续空格的话,仅保留一个空格,即允许字符串中间有多个空格,但连续的空格数不可超过一个.

一.填空题

二.简答题

1.在c#中using和new这两个关键字有什么意义，请写出你所知道的意义？using 指令 和语句 new 创建实例 new 隐藏基类中方法

4.谈谈类和结构的区别？类是引用类型、结构是值类型

5.一个长度为10000的字符串，通过随机从a-z中抽取10000个字符组成。请用c＃语言编写主要程序来实现。

6.对于这样的一个枚举类型：

      enum Color:byte{

          Red,

          Green,

          Blue,

          Orange

       }

     string[] ss=Enum.GetNames(typeof(Color));

     byte[]   bb=Enum.GetValues(typeof(Color));

试写一段程序显示出枚举类型中定义的所有符号名称以及它们对应的数值。

7.您了解设计模式么？请列出您所知道的设计模式的名称。 //

8.请在SQL Server中设计表来保存一个树状结构的组织结构图（假设结构图中只有名称这一项内容需要保存），如果我想查询某一职位下的所有职位，用一个存储过程来实现，你有什么思路？

9.什么叫做SQL注入，如何防止？请举例说明。

10.下面这段代码输出什么？为什么？

     int i=5;

     int j=5;

     if (Object.ReferenceEquals(i,j))

          Console.WriteLine("Equal");

     else

          Console.WriteLine("Not Equal");

//不相等，因为比较的是对象

1.写一个实现对一段字符串翻转的方法，附加一些条件，如其中包括“，”、“.”，对其设计测试用 例 。

2.对一支纸杯设计测试用例（可以是广义的杯，不一定是某一支特定功能的杯）

开发语言概念题

3.什么是反射？

4.用Singleton如何写设计模式

5.C#中的垃圾回收机制是怎样的？

6.什么是Application Pool？

7.链表和数组的区别，各有什么优缺点.

8.Remoting在客户端服务器怎么实现 ?

9.什么是友元函数？

10.用标准C如何实现多态？

11.什么是虚函数？

12.什么是抽象函数？

13.什么是内存泄漏，怎样最简单的方法判断被存泄漏 ?

英语题

14.用英文介绍一下使用C#/C++做的项目，主要功能

15.如果要与美国开电话会议，会如何与美国的工程师沟通

16.如果老板认为你的技术落后，你会怎么回答

数据库知识题

17.使用什么工具来调用存储过程

18.SQL Server的两种索引是何形式？索引的作用？索引的优缺点？

19.触发器的作用

其它知识题及问题

20.什么是Web Service？

21.什么是XML？

22.Socket怎么实现？

1. 创建用户时，需要赋予新用户什么权限才能使它联上数据库。  
2. 如何在tablespace里增加数据文件？  
3. 如何变动数据文件的大小？

4. 哪个VIEW用来检查数据文件的大小？  
5. 哪个VIEW用来判断tablespace的剩余空间  
6. 如何判断谁往表里增加了一条纪录？  
7. 如何重构索引？  
8. 解释什么是Partitioning（分区）以及它的优点。  
9. 你刚刚编译了一个PL/SQL Package但是有错误报道，如何显示出错信息？  
10. 如何搜集表的各种状态数据？  
11. 如何启动SESSION级别的TRACE  
12. IMPORT和SQL\*LOADER 这2个工具的不同点  
13。用于网络连接的2个文件？

# 北京爱恩科技发展有限公司笔试考题

# Name :\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

# Date :\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**Time Start :\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**Time End :\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

# VISUAL BASIC SECTION

*Q1 : Grade yourself on your Visual Basic Skill ?*

*1 – poor …….. 10- Advance player*

Ans:

*Q2 : Declare Function Multiply\_Add Lib “test” (ByRef A as integer ,VeRef B as integer , ByVal c as integer ) as integer*

*Q2.1: Explain what is Declare Function used for ?*

Ans :

*Q2.2 : Explain the difference between ByRef and ByVal ?*

Ans :

Q3 : Is Visual Basic a COMPILER or an INTERPERTER ?

Ans

# C++ SECTION

*Q4 :*

int iVal1 = 0, iVal2 = 0;

int \* ipVal;

ipVal = &iVal1; What is the Value of ipVal = \*ipVal =

ipVal = (int\*)iVal1; What is the Value of ipVal = \*ipVal =

iVal2 = (int)&iVal1; What is the Value of ipVal = \*ipVal =

*Q5:*

void SwapAB(int A, int B)

{

A=B;

B=A;

}

int main()

{

int A,B;

A=1;

B=2;

SwapAB(A,B);

return 0; // A= ?, B= ?

}

*What is the value of A and B ?*

What is the difference between STRUCTURE and OBJECT ORIENTED PROGRAMMING ?

*What is the difference between C and C++ ?*

*What is an interpreter ?*

.

# Q6. Step through the following C code and answer the questions

void FillMem(unsigned char \*pData, unsigned char len)

{

while (len--) \*pData++ = len;

}

void main()

{

unsigned char buffer1[8];

unsigned char buffer2[8];

unsigned char \* pData;

unsigned char cnt1, cnt2;

for (cnt1 = 0; cnt1 < 8; cnt1++)

buffer1[cnt1] = 0xFF;

for (cnt1 = 0; cnt1 < 8; cnt1++)

buffer2[cnt1] = 0xFF;

for (cnt1 = 0, cnt2 = 4; cnt1 < 4; cnt1++, cnt2--)

buffer1[cnt1] = cnt1;

buffer2[cnt2] = cnt2;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Q What is the content of buffer1 & buffer2 ?

buffer1[0] = ?? buffer2[0] = ??

buffer1[1] = ?? buffer2[1] = ??

buffer1[2] = ?? buffer2[2] = ??

buffer1[3] = ?? buffer2[3] = ??

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

for (cnt1 = 0; cnt1 < 8; cnt1++)

buffer1[cnt1] = 0xFF;

FillMem(&buffer1[0], 4);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Q. What is the content of buffer1 ?

buffer1[0] = ??

buffer1[1] = ??

buffer1[2] = ??

buffer1[3] = ??

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

for (cnt1 = 0; cnt1 < 8; cnt1++)

buffer1[cnt1] = 0xFF;

pData = &buffer1[4];

pData = pData - 2;

for (cnt1 = 0; cnt1 < 2; cnt1++)

\*pData++ = cnt1;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Q. What is the content of buffer1 ?

buffer1[0] = ??

buffer1[1] = ??

buffer1[2] = ??

buffer1[3] = ??

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

}

# Visual Basic / C++ Section:

You can use C/C++ or Visual Basic to do the following

Please write a simple function to access the momory card 8x8 bytes

The input parameter for memory location is between 0-63

If user want to write ‘1’ at the memory location 9, then this function will write ‘1’ at cell (1,1)

Eg.

MemAccess(9,1); // MemAccess(int Position, int Value);

MemAccess(2,5);

Result

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 0 |  |  |  |  |  |  |  |  |
| 1 |  |  |  |  |  |  |  |  |
| 2 |  |  |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |  |  |
| 4 |  |  |  |  |  |  |  |  |
| 5 |  |  |  |  |  |  |  |  |
| 6 |  |  |  |  |  |  |  |  |
| 7 |  |  |  |  |  |  |  |  |

Note: you can use array type to define this memory card.

Eg.

*Int MemoryCard[8][8]; // Array definition in C*

{

}

# BACK TO FOUNDATION

1 BYTE = \_\_\_\_\_\_\_\_\_BITS (based on 8 bits processor)

1 INTEGER = \_\_\_\_\_\_\_BYTES (based on 8 bits processor)

1 LONG = \_\_\_\_\_\_\_\_\_BYTES (based on 8 bits processor)

# 8051 ASSEMBLY LANGUAGE PROGRAMMING

*Add a comment on each line !*

MOV DPTR,#atable

MOV A,R7

MOVC A, @A+DPTR

CJNE A,#EEPDELIVERYSTATE,\_JUMP1

RET

\_JUMP1:

MOV R6,A

MOV A,R7

INC A

MOVC A, @A+DPTR

MOV DPL,A

MOV DPH,R6

POP ACC

POP ACC

CLR A

JMP @A+DPTR

*Explain what is the function ?*

**JAVA SECTION:**

Q 1. Construct a Java program to do the following

**(if you do this question, you do not have to answer Q2, Q3 and Q4):**

byte b =

byte pos = ;

bu.IsBitOn(b, pos);

b = 4;

bu.IsBitOn(b, pos);

Result:

> Byte = 5, Bit at position 0 is on.

> Byte = 4, Bit at position 0 is off

Eg.

5 = 00000101

7 6 5 4 3 2 1 0

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |

Bit at position 0 is on

Q 2. Explain why Java program can be easily run on different platforms/operating systems.

Q 3. Describe the structure of a Class in Java

Q 4. Explain *Inheritance* and *Overloaded Function* in Java

1. 你平时会看哪些方面的书籍？喜欢浏览哪些网站？
2. 当你遇到问题时通常如何解决？
3. 谈谈你对.Net与Java的看法
4. 你对公司要求加班有什么样的看法？
5. 简述你对XML Web Service的原理的认识？
6. 说出你所了解的数据库访问组件(例如ADO，至少4种)
7. 对单表进行模糊查询，并对其中两列分别进行升序和降序排列(在一个Select语句中)
8. 根据上题中的表编写一个带输入参数的存储过程，输入参数的数据类型为varchar。

例如：输入参数为“20060912”，返回表中所有日期字段大于该日期的记录

9．用VB或VB.NET或ASP.NET编程，从一个表中选出两个varchar列(其中一列数据不能有重复，并且长度等于4)，并将两列连接成一个字符串，将这些字符串动态加载到TreeView1所有节点的Text属性中

**光声世纪笔试题目**

**1．表 g 和x或c,连接关键字是 :**

**g\_q(date), g\_z(date), g\_c(char) ,g\_k(char)**

**c\_q(date), c\_z(date), c\_c(char) ,c\_k(char)**

**条件是 日期g\_q为2004年1月1日**

**筛选内容**

**g\_q ,g\_z ,g\_c ,g\_k ,c\_k**

**2．将表 day\_eng 中的字段plant\_code尾标识改变 “受”为“售” 例如**

**“2201受” 改为“2201售”**

**条件是日期字段是DateTime为2004年6月**

**3.VS.net 中支持 web.control有哪几个，分页的原理是什么？优缺点又是什么？如果不依赖web control 数据量大时如何实现分页？**

**4．随机数组排序**

**1．9．8．2．5．5-4．9-7．8-3.2-2-2. . . . . . . . .**

**排序结果是**

**1．2．5．8．9．5-2．5-4．8-2. 8-3. 9-5. 9-7. 2-2-2**

**用你熟悉的语言写一个程序要适合上边所有的排序**

**（面视题）**

**5．谈谈ASP 和ASP.NET 的区别**

**6．如何理解跳槽现象？和上司意见相左你怎么处理？**

**7．你如何处理工作中遇到的问题（技术问题或工作问题）**

**（上机题）**

**8．上机写一个DataGrid 或 DataList的 增删改查，作好用MVC 结构写，时限一小时。**

广州运通

1.MFC中最主要根基类（），窗口类的基类（）

2.基于MFC的WIN32可执行应用程序，有且仅有一个从（）类派生的类的对象。

3.所有COM组件都必须实现的一个接口是：

4.一个COM组件，如果要支持自动化，必须实现的接口。

5.MFC中的类，从（）类派生的类都可以有消息映射，请列举一个（或多个）只能收到WM\_COMMAND消息的类：（）。

6.在多线程应用中，为使线程间的某些操作同步，一般使用\_\_\_\_同步对象。请列举在应用此对象时要用的API函数（）。

7.STL有哪些摸板类（）

8.使用Windows API设置一个窗口的文本，是哪一个API函数（）。

9.线程有哪些类型，如何创建？如何终止？

10.如何定义一个自定义消息，并在程序中响应它？

11.从一个DLL中导出函数的方法有哪几种？

12.有组件A，它的头文件包括如下内容：

DEFINE\_GUID(DIID\_IENoteBook, 0x5D32DE27, 0x1A02,0x49AC,0xAF,0x2B,0x9A,0x18,0x95,0x5D, 0xCB,0x45);

DEFINE\_GUID(CLSID\_Document, 0xCC6D602E, 0xE64D, 0x42B9, 0xA6, 0x54,0x57, 0x90,0x69, 0xD1, 0x44, 0xAC);

IENoteBook:public IDispatch

{

//其它略

virtual HRESULT STDMETHOD(ALLTYPE AddHtmlPage())=0;

}

请根据以上的接口声明，写一段代码，得到一个IENoteBook的接口， 再调用他的方法AddHtmlPage(),然后再释放此接口。

13.如果使用WinSock接口来编写网络应用程序，请分别写出客户端和服务端的程序流程（以调用哪些借口函数表示）。

笔试：

1. servlet的生命周期；jsp与servlet的区别。
2. cookie与session的区别；session的运行机制（工作原理）。
3. web.xml的作用。
4. math.round(11.5)和math.round(-11.5)的结果各是多少？
5. 写sql语句，主要考察group by…..having的用法。

面试题：

1. Oracle
   1. 如何创建序列？是否可以回滚？
   2. 存储过程是否可以嵌套？如果可以，最多可以嵌套几层？
   3. 大对象怎么存？大对象可以检索吗？
2. Sql Server

A、自定义函数怎么写？有哪几种？能否返回？

B、image类型？

C、有哪几种触发器？

1. ASP.NET里命名空间的定义;
2. .net中类与对象的关系;
3. ASP.NET中页面传值的方法;
4. IspostBack的作用,当其值为TRUE或FALSE时的作用;
5. 写出求某字段最大值的SQL语句.

**C/VC++工程师笔试试题**

**姓名： 申请职位： 日期：**

**1.用你熟悉的语言描述并构造一个基于字符串节点的二叉树，并说明如何实现通过关键词对节点进行模糊查询。**

**2.在分析某个C++源文件的过程中，有这样一行变量定义式：**

**int t\_from=20,t\_to=12,t\_length=120,i\_count,t\_usetime;**

**现在你并不知道这行一共定义了多少个变量，以及哪些变量进行了初始值设定，要求从该语句中解析出所有以“t\_”开头的变量名，并且如果该变量赋予了初始值的话也需要提取出来，请构造一个方法来实现该要求。**

**3.如果现在交给你一个标准的Windows动态库mylib.dll，已知它采用C++语言编程实现，提供了若干个功能接口，按照你的经验和判断，你认为是否有方法通过工具或者编程的方式获得这些功能接口及其参数信息，如果有，请对该方法进行描述。**

**4.在一个VC编写的对话框应用程序中，对话框的OnInitDialog事件函数里面最后有这么一段程序：**

**int scx = ::GetSystemMetrics(SM\_CXSCREEN);**

**int scy = ::GetSystemMetrics(SM\_CYSCREEN);**

**CRect myrect;**

**GetWindowRect(&myrect);**

**int fromx = (scx - myrect.Width())/2;**

**int fromy = (scy - myrect.Height())/2;**

**SetWindowPos(NULL,fromx,fromy,myrect.Width(),myrect.Height(),SWP\_NOZORDER);**

**请描述其作用，你觉得这段程序在方式、效率上是否能进行提高，如果能，请说明方法。**

**5.就你所熟悉的方法描述Windows平台下的音频数据采集过程，如果可以，请列出关键代码。**

**.net 程 序 员 初 步 测 试 题**

**可以使用任何你所熟悉的编程语言来实现以下算法题:**

1. 写一个函数将输入的任意正整数转换为它所对应的二进制编码并输出。
2. 有两个数值变量分别为a与b　试写算法，实现a与b值的交换，在程序中只能出现a,b两个变量。
3. 有一组无序排列的整数数组如{1,5,7,9,2} 通过算法实现对该数组按从大到小的顺序进行排序，输出排序后的结果同时输入原始数组的数组索引下标，如输出排序结果为{9,7,5,2,1}，输出原始数组索引下标为{3,2,1,4,0}。
4. 已知有一长度为100的无序随机整型数组，且数值范围是[1,100]，写一算法，判断数组中是否有存在重复值，要求，不得嵌套循环，不得使用递归。
5. 用算法实现Round函数效果，这里仅要求将所输入的任意数值型变量保留两位小数后输出（四舍五入），以代码实现，不得使用Printf(“%.2f”,&d)或是.NET中的Round(double value, int digits)函数。

**SQL题**

1. 写出一条Sql语句： 选取表A中 第31条至第40条 这10条记录 （注：在SQLServer, 表A是以整形的无序“TableID”字段作为主关键字，如TableID=1,2,3,7,10,20……100……。）

1.面向对象的思想主要包括什么？

2.什么是ASP.net中的用户控件

3.什么叫应用程序域？什么是受管制的代码？什么是强类型系统？什么是装箱和拆箱？什么是重载？CTS、CLS和CLR分别作何解释？

4.列举一下你所了解的XML技术及其应用

5.值类型和引用类型的区别？写出C#的样例代码。

6.ADO.net中常用的对象有哪些？分别描述一下。

7.如何理解委托？

8.C#中的接口和类有什么异同。

9.。net中读写数据库需要用到哪些类？他们的作用

10.UDP连接和TCP连接的异同。

11.ASP.net的身份验证方式有哪些？分别是什么原理？

12.进程和线程分别怎么理解？

13.什么是code-Behind技术。

14.活动目录的作用。

15..net中读写XML的类都归属于哪些命名空间？

16.解释一下UDDI、WSDL的意义及其作用。

17.什么是SOAP,有哪些应用。

18.如何部署一个ASP.net页面。

19.如何理解.net中的垃圾回收机制。

20.常用的调用webservice方法有哪些？

1.在一个项目中要求采用UDP方式在4567端口上提供网络服务，其功能是将从客户端收到的字符串信息进行反向处理（例如abc变为cba）后发还给客户端。针对该要求，采用你所熟悉的方法（不要求基于Windows操作系统）来实现这一要求，阐述关键代码并简要说明即可，如果可能，希望采用多线程/进程处理方式。

2.是否了解windows平台上的“完成端口”技术，是则请简要描述其原理。

3.就你所最熟悉得方法阐述如何实现将一幅位图指定透明色后进行透明绘制，给出关键代码。

4.现有两张数据表（MySQL数据库）如下：

x\_pt\_info x\_user\_info

字段 类型 长度 小数位 字段 类型 长度 小数位

user\_id int 11 user\_id int 11

pt\_group int 11 user\_name varchar 20

pt\_number decimal 10 2 user\_work int 1

描述一下这个SQL语句的作用：

SELECT b.user\_id as user\_id,b.user\_name as user\_name,round(sum(a.pt\_number),2) as ptnum from x\_pt\_info a,x\_user\_info b where

a.user\_id=b.user\_id and a.pt\_group=1 and b.user\_work=2 group by a.user\_id order by ptnum desc

另外，就你的经验，你觉得该语句是否有改进的余地，如有，请阐述。

5.在某个项目中，测试组以电子邮件的方式给你发送测试报告，电子邮件将采用标题进行类别标识，其格式举例如下：“[IM][网络故障]关于客户端访问连接的异常中断”，第一部分是项目名称，第二部分是类别标识，第三部分是问题标题，由于你处于一个大型项目中，这样的邮件每天都会收到上百封，现在你觉得每次通过收取邮件后手工进行分类太过繁琐，希望开发一个工具用来自动进行分类存储并可以随时进行查询，同时还能对任何一封邮件作出批注（这样的批注主要是给自己查看），以及能进行回复。针对这样的想法，你已经找到了一个OCX控件可以完成邮件的收取（该接口可获得邮件的标题并将正文部分以HTML文件方式存放为传入的指定文件），可以实现邮件的发送（该接口可以输入标题、以指定HTML文件为正文、发送地址并自动发送），而你需要做的是将邮件分门别类的存储并实现查询处理功能。你将怎样来规划并实现这个工具软件？

提出几个参考建议，根据你的经验和能力尽量作答：

a)尽你所能构想尽量多的邮件查询方式方法，你总是希望能更简单更快捷的从数千封邮件内查询到你需要找的那一封

b)不排除使用数据库，如果你考虑采用数据库，希望阐述数据库选型及关键表的主要结构设计

c)当然，你应该在这个工具的容错性能上进行考虑，阐述一下你认为需要注意的容错点以及相应的处理措施

d)你在收发信件的接口调用上是怎么考虑的，需要用定时器吗？是否要考虑多线程处理？更进一步，如果你拿到这个控件的源代码，你觉得是否会考虑修改其接口？

e)你是否会考虑将这个工具软件做成一个系统服务？

一、三层架构的分析

二、get 与 post的区别

三、数据库查询优化：

1、多态性，多种数据库兼容；

2、支持翻页，支持查询总数，页码显示；

3、能处理100万以上数据量；

四、数据库的高级查询：

|  |
| --- |
| course |
| id |
| name |
| location |
| time |

|  |
| --- |
| student |
| id |
| name |
| age |

|  |
| --- |
| xk |
| sid |
| cid |
| score |

1、读出没有修的课程；

2、读出所修人数超过100人的课程；

3、列出所有课程的名称以及所修人数；

4、怎样提高查询速度；

北京泡泡网网络有限公司

1.两个表，写查询语句，根据两个字段一个是升序，一个将序。

2.根据第一题，每页面显示10条记录，在第25页时怎样显示

3.编程（C#）：\*\*\*\*\*

\*\*\*\*

\*\*\*

\*

4.STRING和STRINGBUILDER区别

5.看程序写结果

6.一个文本，在他获得焦点的时候弹一个“OK”的对话筐（JAVASCRIPT）

剩下2题都是编程。

清华同方C++的笔试题（部分）

1改错题

2、3题写程序的输出结果

这是第2题：

以下三条输出语句分别输出什么？

char str1[] = "abc";

char str2[] = "abc";

const char str3[] = "abc";

const char str4[] = "abc";

const char\* str5 = "abc";

const char\* str6 = "abc";

cout << boolalpha << ( str1==str2 ) << endl; // 输出什么？

cout << boolalpha << ( str3==str4 ) << endl; // 输出什么？

cout << boolalpha << ( str5==str6 ) << endl; // 输出什么？

第3题是类与子类的输出结果，忘记了

4：DLL两种加载方式，显式加载和隐式，问.exe文件用隐式加载使用DLL时，DLL文件放在哪里？

5：MFC文档类视图的结构及主要作用？

6：头文件ifdef/define/endif的作用？

7：#include<filename.h>与@include"filename.h"的区别？

8：写内存拷贝函数 原型给出 void\*mymemcpy(void\*dest,void\*src,int count)

9:给出今天2006年8月31日是周四，要求把今年所有的周三的日子找出来，编程实现

10：编程实现sum(n)=1!-2！+3！-4！+......+n!

11:n个人围成一圈，从一开始报数，报道3的出队，最后剩谁？用数据结构实现

12：忘了。

13：智力测试

14：英文翻译

1.在c#中using和new这两个关键字有什么意义，请写出你所知道的意义？using 指令 和语句 new 创建实例 new 隐藏基类中方法

4.谈谈类和结构的区别？类是引用类型、结构是值类型

5.一个长度为10000的字符串，通过随机从a-z中抽取10000个字符组成。请用c＃语言编写主要程序来实现。

6.对于这样的一个枚举类型：

      enum Color:byte{

          Red,

          Green,

          Blue,

          Orange

       }

     string[] ss=Enum.GetNames(typeof(Color));

     byte[]   bb=Enum.GetValues(typeof(Color));

试写一段程序显示出枚举类型中定义的所有符号名称以及它们对应的数值。

7.您了解设计模式么？请列出您所知道的设计模式的名称。 //

8.请在SQL Server中设计表来保存一个树状结构的组织结构图（假设结构图中只有名称这一项内容需要保存），如果我想查询某一职位下的所有职位，用一个存储过程来实现，你有什么思路？

9.什么叫做SQL注入，如何防止？请举例说明。

10.下面这段代码输出什么？为什么？

     int i=5;

     int j=5;

     if (Object.ReferenceEquals(i,j))

          Console.WriteLine("Equal");

     else

          Console.WriteLine("Not Equal");

**Name:**

1、x86,win32,vc++6,默认属性的控制台工程：

class C1

{

public:

virtual int sum(int i1, int i2)

{

return (i1 + i2);

}

private:

double m\_var1;

char m\_var2;

};

char sz[] = "abcd";

问题：sizeof(C1) = ?

sizeof(sz) = ?

2、C++中的虚继承是解决什么问题的？请举例说明。

3、请写出下面的输出：

class B

{

public:

virtual void Print(void)

{

printf(“B::Print\n”);

}

virtual void Reprint(void)

{

printf(“B:Reprint\n”);

}

void Algo(void)

{

Print();

Reprint();

}

};

class D : public B

{

public:

virtual void Print(void)

{

printf(“D::Print\n”);

}

};

void main()

{

B \*p = new D();

p->Print();

p->Algo();

}

4、请指出下面程序中不合理的地方。

Class A

{

public:

A(void) : m\_nVar2(m\_nVar1), m\_nVar1(0), m\_cVar3(0)

{

m\_cVar3 =3;

}

protected:

int m\_nVar1;

int m\_nVar2;

const int m\_cVar3;

};

5、在WIN32平台下，进程和线程句柄的引用计数的初始值是多少？其句柄是否为可等待对象？内核对象和用户对象的显著区别是什么？

6、WaitForSingleObject的超时参数为0是何含意？Sleep的超时参数为0又是何含意？

7、请列举您对设计模式的理解？并用自然语言举例说明其中一个例子，另外请使用C++写出一个Singleton的例子。

8、TCP建立需要进行了几次握手？正常传输过程中的捎带确认是什么含意？TCP的发送窗口、拥塞窗口与接受窗口的意义何在？为什么UDP没有这些感念和机制？音视频等实时性要求比较高的数据应该用哪种传输协议作承载？你都考虑了哪些方面？

9、如果往一个箱子里倾倒沙子，每一秒钟往箱子中注入的是前面倾注的总和，在10秒钟时恰好将箱子填满，请问第7秒钟时注入的沙子是箱子容量的多少？

10、请随意列举您曾经读过的认为比较优秀的书籍（计算机编程、软件过程、管理、励志不限）。

1、EJB有哪几中BEAN？

2、画Struts流程图。

3、画EJB容器图。

1、UML用例图的画法、活动图的画法；

2、算内存容量大小（根据内存地址）

3、二叉树的排序（前序、后序、中序）

4、数据库建表

5、基本的SELECT语句；

6、通讯协议的相关知识（TCP/IP、HTTP等）

7、排序的最优算法；

8、英文完形填空（技术文档）

1、list 、set、map 有什么区别？

2、抽象类和接口的区别？

3、连接ORACLE数据库

4、什么是序列化？

5、多线程终止？

6、读出文件，显示行号？

7、Runtime Exception和Exception区别？

1. GC是什么?
2. 当一线程处于等待状态时,用什么方法可以将其唤醒？
3. 当一个线程处于new状态时，通过什么方法使其进入runnable状态？
4. 2乘以8的最快计算方法？
5. checked异常与unchecked异常的区别？
6. 抽象类与接口的区别？
7. 编程题：

Class father{fathe() {}

Class sun extends father

{ son() {…..}

Public static void main()

{ son a=new father();

}

}

此程序是否正确，为什么？

1. 说明Friendly,private,protect,public四者间的区别?
2. status Bean与statusless Bean的区别
3. string与stringBuffer的区别？
4. SQL Server DB中，是由何种机制来保持数据的一致性的。
5. SQL Server中触发器编程部分的增加，填充。
6. What does the ThreadStatic attribute do?

(线程静态成员的作用是什么)

1. Does C# support multiple inheritance?

(C#支持多重继承吗)

1. C# provides, by default a parameterless constructor. If I write a constructor that takes a string as a parameter, but want to keep the parameterless constructor. How many constructors should I write?

(C#提供默认构造函数(不带参数),如果我写了一个带有一个string类型参数的构造函数,但是又想保留不带参数的构造函数,那么我需要写多少个构造函数)

1. What’s the difference between an interface and abstract class?

(接口与抽象类有什么区别)

1. What’s the difference between deep copy and shallow copy?

(深拷贝与浅拷贝有什么区别)

1. If an exception is thrown inside a catch block, will the finally block be still be called?

(如果catch块捕获了一个异常,那么finally块还会被调用吗)

1. a)Can a struct inherit from another struct?

(结构体能继承结构体吗)

b)Can a struct inherit from another class?

(结构体能继承类吗)

c)Can a struct be the base of another class?

(结构体可以作为一个类的基类吗)

d)Can a struct implement an interface?

(结构体可以实现一个接口吗)

e)What’s the difference between struct and class?

(结构体与类有什么区别)

8．What’s the difference between Debug and Trace class?

(Debug类与Trace类有什么区别)

1. What’s the final result?

(下面这段代码的最终运行结果是什么)

double expectedValue = 1/2;

if(expectedValue > 0)

{

expectedValue = expectedValue + 0.5;

}

Console.WriteLine(expectedValue);

1. What is the role of the DataReader class in ADO.NET connections?

(DataReader在ADO.NET连接中起到什么样的角色)

面试题参考

1. 用C#或者JAVA写一个MAIN函数
2. 面向对象的几种方式，你在实际应用中用到了哪些，请描述
3. 写出asp.net或者jsp写出动态生成网页的过程,并说出其原理
4. 写出n-tie的结构和每一层的作用
5. 设计数据库需要考虑的问题
6. 什么是重构,以及它的表现方式
7. 接口的定义,它在内存中如何实现

鼎天科技集团试题

上机：

第一题：1 1 2 3 5 8······

使用递归算法，求出第30位。

第二题：

使用用户控件，实现显示数据库数据

笔试题：

1. asp和asp.net的区别？
2. 求表的31条记录到40条记录（编号不连续）
3. 用户控件和自定义控件的使用？
4. 程序的输出结果。

9 asp.net如何验证数据正确性？以"E\_mail的@"为例

答：在RegularExpressionValidator验证控件中加入正则表达式：\w+([-+.']\w+)\*@\w+([-.]\w+)\*\.\w+([-.]\w+)\*

10 什么是webservice，什么情况下使用,如何使用？

答：

现在我将列举三种情况，在这三种情况下，你将会发现使用Webservice会带来极大的好处。此后，我还会举出不应该使用Webservice的一些情况。

跨越防火墙的通信

如果你的应用程序有成千上万的用户，而且他们都分布在世界各地，那么客户端和服务器之间的通信将是一个棘手的问题。那是因为客户端和服务器之间通常都会有防火墙或者代理服务器。在这种情况下，你想使用DCOM就不是那么简单了，而且，通常你也不愿意把你的客户端程序发布到如此庞大数量的每一个用户手中。于是，你最终选择了用浏览器作为客户端，写下一堆ASP页面，把应用程序的中间层暴露给最终用户。结果呢？运气好的话，只是开发难度大了一些，运气不好的话，就会得到一个根本无法维护的应用程序。

想象一下你应该怎么在你的应用程序里面加入一个新的页面：你必须先建立好用户界面(Web页面)，以及在这个页面后面，包含相应商业逻辑的中间层组件。这还不够，你还要再建立至少一个ASP页面，用来接受用户输入的信息，调用中间层组件，把结果格式化为HTML形式，最后还要把"结果页"送回浏览器。要是客户端代码不再如此依赖于HTML表单，客户端的编程不就简单多了吗？还有，建立ASP页面的那一步可以省略掉吗？

当然。如果你的中间层组件是Webservice的话，你完全可以从用户界面直接调用中间层组件，从而省掉建立ASP页面的那一步。要调用Webservice，你可以直接使用MicrosoftSOAPToolkit或.NET这样的SOAP客户端，也可以使用你自己开发的SOAP客户端，然后把它和你的应用程序连接起来。这样做，不仅可以缩短开发周期，还可以减少代码的复杂度，并增强整个应用程序的可维护性。同时，你的应用程序也不再需要在每次调用中间层组件时，都跳转到相应的"结果页"了。

以我的经验来看，在一个用户界面和中间层有较多交互的应用程序中，使用Webservice这种结构，可以轻松的节省花在用户界面编程上的20%的开发时间。这样做还有另一个好处，就是你将得到一个由Webservice组成的中间层，这一层是完全可以在应用程序集成或其他场合下被重用的。最后，通过Webservice把你的应用程序的逻辑和数据暴露出来，还可以让其它平台上的客户重用你的应用程序。

应用程序集成

企业级的应用程序开发者都知道，企业里经常都要把用不同语言写成的在不同平台上运行的各种程序集成起来，而这种集成将花费很大的开发的力量。你的应用程序经常都需要从运行在古老的IBM主机上的程序中获取数据；或者再把数据发送到主机或UNIX应用程序中去。即使是在同一个平台上，不同的软件厂商生产的各种软件也常常需要集成起来。通过Webservice，应用程序可以用标准的方法把功能和数据暴露出来，供其它的应用程序使用。

例如，你有一个订单登录程序，用于登录从客户来的新订单，包括客户信息、发货地址、数量、价格和付款方式等信息。同时，你还有一个订单执行程序，用于实际货物发送的管理。这两个程序是来自不同软件厂商的。一份新订单进来之后，订单登录程序需要通知订单执行程序发送货物。通过在订单执行程序上面增加一层Webservice，订单执行程序可以把"AddOrder"函数暴露出来。这样，每当有新订单到来时，订单登录程序就可以调用这个函数来发送货物了。进而通过Webservice集成应用程序

B2B的集成

用Webservice集成应用程序，可以使你公司内部的商务处理更加自动化。但当交易跨越了你的供应商和客户，突破了公司的界线时又会怎么样呢？跨公司的商务交易集成通常叫做B2B集成。

Webservice是B2B集成成功的关键。通过Webservice，你的公司可以把关键的商务应用暴露给指定的供应商和客户。例如，把你的电子下单系统和电子发票系统暴露出来，你的客户就可以以电子的方式向你发送购货订单，而你的供应商则可以以电子的方式把原料采购的发票发送给你。当然，这并不是一个新的概念：电子文档交换(EDI)早就是这样了。Webservice和EDI之间的主要区别在于，Webservice的实现要比EDI简单得多，而且Webservice是运行在Internet上的，在世界任何地方都可轻易实现，这样其运行成本就相对较低。不过，Webservice并不像EDI那样，是文档交换或B2B集成的一套完整的解决方案。Webservice只是B2B集成的一个关键部分，还需要许多其它的部分才能完成这个集成。

用Webservice来实现B2B集成的最大好处在于可以轻易实现互操作性。只要把你的商务逻辑暴露出来，成为Webservice，你就可以让任何指定的合作伙伴轻松的调用你的商务逻辑，而不管他们的系统在什么平台上运行，使用的是什么开发语言。这样就大大减少了花在B2B集成的上的时间和成本。这样的低成本让许多原本无法承受EDI的投资成本的中小企业也能实现B2B集成。

软件重用

软件重用是一个很大的主题，它有很多的形式和程度。最基本的形式是源代码模块或者类一级的重用。另一种形式是二进制形式的组件重用。当前，像表格控件或用户界面控件这样的可重用软件组件在市场上都占有很大的份额。但这类软件的重用都有一个很严重的限制：重用仅限于代码，而数据不能被重用。原因在于你可以很轻易的发布组件甚至源代码，但要发布数据就没那么容易了，除非那些数据都是不会经常变化的静态数据。

而Webservice允许你在重用代码的同时，重用代码后面的数据。使用Webservice，你不再像以前那样，要先从第三方购买、安装软件组件，再从你的应用程序中调用这些组件。你只需要直接调用远端的Webservice就可以了。举个例子，你想在你的应用程序中确认用户输入的邮件地址，那么，你只需把这个地址直接发送给相应的Webservice，这个Webservice就会帮你查阅街道地址、城市、省区和邮政编码等信息，确认这个地址的确在相应的邮政编码区域。Webservice的提供商可以按时间或使用次数来对这项服务进行收费。这样的服务要通过组件重用来实现是不现实的，因为那样的话你必须下载并安装好包含街道地址、城市、省区和邮政编码等信息的数据库，而且这个数据库还是不能实时更新的。

另一种软件重用的情况是把好几个应用程序的功能集成起来。例如，你想要建立一个局域网上的门户站点应用，让用户既可以查询他们的联邦快递包裹，察看股市行情，又可以管理他们的日程安排，还可以在线购买电影票。现在Web上有很多应用程序供应商，都在其应用中实现了上面的这些功能。一旦他们把这些功能都通过Webservice暴露出来，你就可以非常轻易地把所有这些功能都集成到你的门户站点中，为用户提供一个统一的、友好的界面。

用Webservice来集成各种应用中的功能，为用户提供一个统一的界面

许多应用程序都会利用Webservice，把当前基于组件的应用程序结构扩展为组件和Webservice的混合结构。你也可以在应用程序中使用第三方的Webservice提供的功能。你还可以把你自己的应用程序的功能通过Webservice提供给别人。所有这些情况下，你都可以重用代码和代码后面的数据。总之，Webservice将是软件重用的一种非常有力的形式。

什么时候不应该使用WebService

一个对Webservice的完整介绍还应该包括什么时候不该用Webservice。经过前面的介绍，我们知道了Webservice在通过Web进行互操作或远程调用的时候是最有用的。不过，还有许多情况，Webservice根本不能给你带来任何好处。

单机应用程序

目前，我们还有很多桌面应用程序是供商用和个人使用的。其中一些只需要与运行在本机上的其他程序通信。在这种情况下，我们最好就不要再用Webservice，只要用本地的API就可以了。COM非常适合于在这种情况下工作，因为它既小又快。运行在一台服务器上的服务器软件也是这样：最好直接用COM或其他本地的API来进行应用程序间的调用。当然Webservice也能用在这些情况下，但那样不仅消耗太大，而且不会给你带来任何好处。

局域网上的同构应用程序

在许多应用中，你所有的程序都是用VB或VC开发的，都在Windows平台下使用COM，都运行在同一个局域网上。例如，你有两个服务器应用程序需要相互通信，或者你有一个Win32或WinForm的客户程序要连接到局域网上的另一个服务器程序。在这些程序里使用DCOM会比SOAP/HTTP有效的多。类似的，如果你的一个.NET程序要连接到LAN上的另一个.NET程序，那么你应该使用.NETremoting。有趣的是，在.NETremoting中，你也可以指定使用SOAP/HTTP来进行Webservice调用。不过最好还是直接通过TCP进行RPC调用，那样会有效得多。总之，只要你从应用程序结构的角度看来，有别的方法比Webservice更有效，更可行，那就不要再用Webservice。

总结

Webservice是创建可互操作的分布式应用程序的新平台。Webservice的主要目标是跨平台的可互操作性。为了达到这一目标，Webservice是完全基于XML、XSD等独立于平台、独立于软件供应商的标准的。

Webservice在应用程序跨平台和跨网络进行通信的时候是非常有用的。Webservice适用于应用程序集成、B2B集成、代码和数据重用，以及通过Web进行客户端和服务器的通信的场合。

当然，Webservice也不是万能的，你不能到处滥用Webservice。在有些情况下，Webservice会降低应用程序的性能，而不会带来任何好处。例如，一台机器或一个局域网里面运行的同构应用程序就不应该用Webservice进行通信。

11 用gridview如何把其中一行添加下拉列表框并绑定数据库？如何分页和动态排序？

答：

先生成模版，在EditItemTemplate里放一个下拉筐进去，选择数据源即可 。

把gridview的属性Allowpaging ＝ true 就能分页，AllowSorting＝true就能排序

或者用gridview.Sort方法排序。

12 说明一下软件开发流程？

答：分析（需要，概要，详细），开发（编程，单元测试），测试（集成测试），维护

13 如果做到编码规范？

答：方法，类，变量尽量写有意义的单词。注释有写清楚，但不要罗唆

14 try catch finally中 catch和finally的作用

答：catch扑获异常，finally不管代码是否出现异常都执行

15 web页面代码的执行顺序？

答：从上到下

16 用户控件和自定义控件的异同点?

答：用户控件(UserControl): 扩展名为\*.ascx,跟\*.aspx在结构上相似，是指页面中

加载的功能块,只是用户控件不能单独作为页面运行,必须嵌入到\*.aspx页面或其它用

户控件中使用 。

自定义控件，跟HtmlControl或WebControl相似，编译后可以添加引用到工具栏里

面，直接用鼠标拖动使用。

看情况使用，如果功能比较多，会用自定义控件。

17 当发生一个http代码错误时可以显示一个自定义的错误页面给用户,怎么花最小代价完成这个目标?

答：

18 一个人拿了100元去商店买25元钱的东西,店主没有零钱找,于是去旁边的小摊贩换了100的零钱,找给了那个人75,那个人走了之后,小摊贩拿着100元钱去找店主,说是假的,店主看了看钱,发现真的是假的,于是赔给了小摊贩一个真的100块钱,问店主一共亏了多少钱,为什么?

答：25。进75元，出100元。

19 .net中哪项技术实现跨多语言

A、CLR B、CTS C、CLS D、CTT

答：CLR

20 在B/S模式中，你是使用几层架构实现的，说说各层之间的关系和这样的好处？

答：MVC（模型－视图－控制器）

视图(View)代表用户交互界面.

模型(Model)：就是业务流程/状态的处理以及业务规则的制定,业务模型还有一个很重要的模型那就是数据模型.

控制(Controller)可以理解为从用户接收请求, 将模型与视图匹配在一起，共同完成用户的请求.

此套试题总分105，考试时间为1小时，考试分数为70分为合格。

**C++/MFC试题**

**一．填空题(26分)**

1． WIN32平台下，sizeof(short) = \_\_\_\_，sizeof(int) = \_\_\_\_，sizeof(long) = \_\_\_\_。(3分)

2．请给出如下程序的结果(2分)

int a = 3;

int b = a << 3;

a = \_\_\_\_，b = \_\_\_\_。

3．请给出如下程序的结果(2分)

int aaa = 0x01;

htonl(aaa) = \_\_\_\_。

4．请给出如下程序的结果(2分)

#define MAX\_NUM 100+200

int nTemp = MAX\_NUM\*10;

则Temp = \_\_\_\_。

5．请给出如下程序的结果(3 分)

char szTemp[1000] = "";

int nLen1 = sizeof(szTemp);

int nLen2 = strlen(szTemp);

strcpy(szTemp, "abc");

int nLen3 = sizeof(szTemp);

int nLen4 = strlen(szTemp);

int nTemp[100];

int \*pTemp = nTemp;

int nLen5 = sizeof(pTemp);

char szResult[200] = "";

sprintf(szResult, "%d,%d,%d,%d,%02d.", nLen1, nLen2, nLen3, nLen4, nLen5);

则szResult = \_\_\_\_。

6．MFC中，大部分类是从哪个类继承而来（CCmdTarget、CObject、CWinApp、CWnd）？(2分)\_\_\_\_

7．内存是进程范围or线程范围；\_\_\_\_

CPU调度时，针对进程or线程；\_\_\_\_

函数调用堆栈，针对进程or线程。\_\_\_\_(3分)

8．调用函数bbb后，输出是什么(4分)

void ccc(int x)

{

char szTemp[10] = "";

x = 2;

sprintf(szTemp, "%d,", x);

afxDump << szTemp;

if(x = 3)

{

int x = 4;

sprintf(szTemp, "%d,", x);

afxDump << szTemp;

}

sprintf(szTemp, "%d,", x);

afxDump << szTemp;

}

void bbb()

{

char szTemp[10] = "";

int x = 7;

ccc(x);

sprintf(szTemp, "%d,", x);

afxDump << szTemp;

}

**二．改错题(总共15分,每题5分)。**

1．下面代码有何错误

void func1()

{

int \*pa = NULL;

func2(pa);

delete pa;

}

void func2(int \*pb)

{

pb = new int(5);

}

2．下面代码有何错误

void func2(int \*value)

{

\*value = 2;

}

void func1()

{

int \*p = 0;

func2(p);

}

3．

int func1(int& b)

{

return 0;

}

void func2()

{

int bbb = 3;

func1(&bbb);

func1(bbb);

}

func2中有何错误，func1的参数b的类型是什么。

**三．简答题(64分)**

1.请简述C、C++、VC、MFC在概念上的区别(4分)

2．请写一个函数重载的简单例子(4分)

3.用什么函数开启新进程、线程。(4分)

4.SendMessage和PostMessage有什么区别(4分)

5.WaitForSingleObject有何作用；m\_pThrd的类型是CWinThread\*时，WaitForSingleObject(m\_pThrd->m\_hThread, INFINITE);有何作用。(4分)

6. \_\_stdcall、\_\_cdecl、\_\_pascal在什么方面有所不同。(4分)

7．请把下述代码加上异常处理。(6分)

int MyWriteFile(CString strFileName, CString strText)

{

int nRet = 0;

CFile myFile;

myFile.Open(strFileName, CFile::modeWrite|CFile::shareExclusive|CFile::modeCreate, NULL);

int nLen = strText.GetLength();

myFile.Write((char\*)(LPCSTR)strText, nLen);

myFile.Close();

return nRet;

}

8．请解释“func”为何种类型，这种类型的作用什么，变量ttt 的值是多少？(6分)

typedef int (\*func)(int, int\*);

int xxx(int a, int \*p)

{

return a + \*p;

}

int dowork(func aaa, int bbb, int \*ccc)

{

return aaa(bbb, ccc);

}

int sss = 4;

int ttt = dowork(&xxx, 3, &sss);

9．请问下述代码中: int operator+(…）起什么作用？this是什么？ccc 的值最终为多少？(6分)

class Fruit

{

public:

Fruit()

{

weight = 2;

}

Fruit(int w)

{

weight = w;

}

int operator+(Fruit f)

{

return this->weight \* f.weight;

}

private:

int weight;

};

Fruit aaa;

Fruit bbb(4);

int ccc = aaa + bbb;

10.请解释下面代码采用了何种C++特性（C语言不具备），作用是什么？(6分)

template<typename T>

T sum(T a, T b)

{

return (a + b);

}

11．请解释aaa.h中下面代码的功能(5分)

#if !defined(AFX\_MYSUDU\_H\_\_9B952BEA\_A051\_4026\_B4E5\_0598A39D2DA4\_\_INCLUDED\_)

#define AFX\_MYSUDU\_H\_\_9B952BEA\_A051\_4026\_B4E5\_0598A39D2DA4\_\_INCLUDED\_

**... ...**

#endif

12．CMemoryState主要功能是什么(5分)

13．请阅读下述代码，写出程序执行的结果（6分）

#include <iostream>

using namespace std;

class CBase

{

public:

virtual void print()

{

cout<< "base" << endl;

}

void DoPrint()

{

print();

}

};

class CChild1: public CBase

{

public:

virtual void print()

{

cout<< "child1" << endl;

}

};

class CChild2: public CBase

{

public:

virtual void print()

{

cout<< "child2" << endl;

}

};

void DoPrint(CBase \*base)

{

base->DoPrint();

}

void main()

{

CBase\* base = new CBase();

CChild1\* child1 = new CChild1();

CChild2\* child2 = new CChild2();

DoPrint(child1);

DoPrint(child2);

DoPrint(base);

delete base;

base = child1;

base->print();

delete child1;

delete child2;

}

1. 问答  
   1、实模式与保护模式。为什么要设计这两种模式？好处在什么地方？分别写出各自寻址的  
   过程。  
     
   2、请阅读以下一段程序，并给出答案。  
   class A  
   {  
   public：  
     A(){ doSth() }  
     virtual void doSth(){ printf("I am A");}  
   }  
     
   class B：public A  
   {  
   public：  
     
     virtual void doSth(){ printf("I am B");}  
   }  
     
   B b;  
   执行结果是什么？为什么？  
     
   3、在STL的应用中 map<int,int>这种key-value的应用很多，如果key的类型是GUID，该如  
   何处理？  
     
   4、一个内存变量a=5，有5个线程需要对其进行操作，其中3个对a进行加1操作，2个对a进  
   行减1操作，为了保证能够得到正常结果6，需要使用什么方法？（列出越多越好）  
     
   5、描述并比较以下对象：事件，信标，临界区，互斥对象。  
     
   6、cdecl、stdcall、fastcall是什么？哪种可以实现个数不定的入口参数，为什么？

1. 程序设计（以下题目请写出实现代码）  
   1、有一段文本，统计其中的单词数。例如：  
   As a technology , "HailStorm" is so new that it is still only known by its   
     
     
   code name.  
   注意：单词间的间隔不一定是一个空格  
     
   2、国际象棋有8×8格，每个格子可放一个棋子。皇后的规则是可以横、竖、斜移动。在一  
   个棋盘放置8个皇后，并使它们互相无法威胁到彼此。  
     
   3、输入二个64位的十进制数，计算相乘之后的乘积

已知strcpy函数的原型是：  
        char \* strcpy(char \* strDest,const char \* strSrc);  
    1.不调用库函数，实现strcpy函数。  
    2.解释为什么要返回char \*。

**三**

1. How do you code an infinite loop in C?   
  
2. Volatile:   
  
a) What does the keyword volatile mean? Give an example   
  
b) Can a parameter be both const and volatile? Give an example   
  
c) Can a pointer be volatile? Give an example   
  
3. What are the values of a, b, and c after the following instructions:   
  
int a=5, b=7, c;   
  
c = a+++b;   
  
4, What do the following declarations mean?   
  
a) const int a;   
  
b) int const a;   
  
c) const int \*a;   
  
d) int \* const a;   
  
e) int const \* a const;   
  
5. Which of the following statements describe the use of the keyword   
  
static?   
  
a) Within the body of a function: A static variable maintains its value   
  
between function revocations   
  
  
  
  
  
b) Within a module: A static variable is accessible by all functions   
  
within that module   
  
c) Within a module: A static function can only be called by other   
  
functions within that module   
  
6. Embedded systems always require the user to manipulate bits in   
  
registers or variables. Given an integer variable a, write two code fragments.   
  
The first should set bit 5 of a. The second shnuld clear bit 5 of a. In both   
  
cases, the remaining bits should be unmodified.   
  
7. What does the following function return?   
  
char foo(void)   
  
{   
  
unsigned int a = 6;   
  
iht b = -20;   
  
char c;   
  
(a+b > 6) ? (c=1): (c=0);   
  
return c;   
  
}   
  
8. What values are printed when the following C program is executed?   
  
int i = 8;   
  
void main(void)   
  
(   
  
  
  
  
  
  
  
9. What will be the output of the following C code?   
  
main()   
  
{   
  
int k, num= 30;   
  
k =(num > 5 ? (num <=10 ? 100:200): 500);   
  
printf("%d", k);   
  
}   
  
10. What will the following C code do?   
  
int \*ptr;   
  
ptr =(int \*)Ox67a9;   
  
\*ptr = Oxaa55;   
  
11. What will be the output of the follow C code?   
  
#define product(x) (x\*x)   
  
main()   
  
{   
  
int i = 3, j, k;   
  
j = product(i++);   
  
k = product(++i);   
  
printf("%d %d",j,k);   
  
}   
  
12. Simplify the following Boolean expression   
  
!((i ==12) || (j > 15))

struct Node {  
int value;  
Node\* next;  
};  
1.1 Get the value of the Nth node from last node in the linked list.  
  
PARAM HEAD: the first element in the linked list:  
PARAM n: the number of the node counted reversely  
RETURN: the value of the node, or -1 if not exists  
  
int GetValue(Node\* HEAD, int n)  
{  
}  
  
1.2 Delete a node WITHOUT using the HEAD pointer.  
PARAM p: A pointer pointed to a node in the middle of the linked list.  
RETURN: void  
  
void Delete(Node\* p)  
{  
}  
  
1.3 Insert a new node before p WITHOUT using the HEAD pointer  
PARAM p: A pointer pointed to a node in the middle of the linked list.  
PARAM value: new Node value  
RETURN: void  
  
void Insert(Node\* p, int value)  
{  
}  
  
Question 2:  
  
Please write a String class with following features:

**四:**  
1. Default constructors with no parameters passed.  
2. Constructor with parameter const char\* sourceString passed.  
3. Destructor.  
4. Copy constructor.  
5. Assignment operator.  
6. Operator overloading operator+= which appends another String instance into current String instance.  
7. A method returning length of the String.  
8. A conversion operator which converts the current String instance into raw C-style string of type const char\*.  
  
**五**  
1. Part of the signature of the String class is:  
class String {  
....  
};  
  
Other part of signature and implementation is completed by you.   
  
2. Please make the implementation as simple as possible. Only help functions and classes from standard C&C++ may be used to aid your implementation.  
  
3. If possible please suggest further improvement of the String class.  
  
Question 3:  
  
Given a link list, detect whether it's circular using only one loop.   
  
Tips: Below implementation is allowed  
for( ... )  
{  
...  
}  
  
The following implementations is NOT allowed  
...  
for( ... )  
{  
...  
for( ... ) {...}  
}  
...  
  
or  
...  
for( p = list->head, q = list->head; p != NULL && q != NULL; p = p->next )  
{  
...  
}  
...  
for( ... )  
{  
...  
}

**六**

(1)  
　　  
　　int Calc(unsigned int x)  
　　{  
　　 int count=0;  
　　 while(x)  
　　 {  
　　 printf("x=%i\n",x);  
　　 count++;  
　　 x=x&(x-1);  
　　 }  
　　  
　　 return count;  
　　  
　　  
　　  
　　}  
　　问Calc(9999)的值是多少。  
　　  
　　(2)检查错误  
　　  
　　int CopyStringCount(const char\* Str)  
　　{  
　　 int nCount = 0;  
　　 char\* pBuffer;  
　　  
　　 pBuffer = new char[\_MAX\_PATH];  
　　  
　　  
　　 strcpy(pBuffer,Str);  
　　  
　　 for(;\*pBuffer!='\0'; pBuffer++)  
　　 if(\*pBuffer == '\') nCount ++;  
　　   
　　// delete [] pBuffer;  
　　  
　　 return nCount;  
　　}  
　　  
　　(3)写出结果  
　　  
　　void foo(int p1[])  
　　{  
　　 \*p1 += 5;  
　　}  
　　  
　　void bar(int p2[])  
　　{  
　　 p2[1] = 15;  
　　}  
　　  
　　void main()  
　　{  
　　 int a[]={3,4,5};  
　　 int b[]={3,4,5};  
　　 int \*p2;  
　　 p2=&a[1];  
　　 bar(p2);  
　　 printf("%i %i %i\n",a[0],a[1],a[2]);  
　　 p2=&b[0];  
　　 p2++;  
　　 foo(p2);  
　　 bar(p2);  
　　 printf("%i %i %i\n",b[0],b[1],b[2]);  
　　}  
　　  
　　(4)  
　　  
　　有一5节车厢的过山车，每节能座两人，现有Luair,Jack,Gwen,Tom,Mark,Paul,6人去乘  
　　车，有以下条件  
　　1,Luair和别人同乘  
　　2,Mark 不合别人同乘，而且Mark的前一节车厢是空的  
　　3,Tom 不和Gwen 与 Paul 中的任何一人同乘  
　　4,Gwen乘3，或者4节  
　　  
　　   
　　  
　　  
　　-----------------------------------------------------------------------------  
　　-------------  
　　

**七**  
　　1写出下列算法的时间复杂度。   
　　(1)冒泡排序；   
　　(2)选择排序；   
　　(3)插入排序；   
　　(4)快速排序；   
　　(5)堆排序；   
　　(6)归并排序；   
　　  
　　2写出下列程序在X86上的运行结果。   
　　  
　　struct mybitfields   
　　{   
　　unsigned short a : 4;   
　　unsigned short b : 5;   
　　unsigned short c : 7;   
　　}test   
　　  
　　void main(void)   
　　{   
　　int i;   
　　test.a=2;   
　　test.b=3;   
　　test.c=0;   
　　  
　　i=\*((short \*)&test);   
　　printf("%d\n",i);   
　　}   
　　  
　　3写出下列程序的运行结果。   
　　  
　　unsigned int i=3;   
　　cout<  
　　4写出下列程序所有可能的运行结果。   
　　  
　　int a;   
　　int b;   
　　int c;   
　　  
　　void F1()   
　　{   
　　b=a\*2;   
　　a=b;   
　　}   
　　  
　　void F2()   
　　{   
　　c=a+1;   
　　a=c;   
　　}   
　　  
　　main()   
　　{   
　　a=5;   
　　//Start F1,F2 in parallel   
　　F1(); F2();   
　　printf("a=%d\n",a);   
　　}   
　　  
　　5考察了一个CharPrev()函数的作用。   
　　  
　　6对 16 Bits colors的处理，要求：   
　　（1）Byte转换为RGB时，保留高5、6bits；   
　　（2）RGB转换为Byte时，第2、3位置零。   
　　  
　　7一个链表的操作，注意代码的健壮和安全性。要求：   
　　（1）增加一个元素；   
　　（2）获得头元素；   
　　（3）弹出头元素（获得值并删除）。   
　　  
　　8一个给定的数值由左边开始升位到右边第N位，如   
　　0010<<1 == 0100   
　　或者   
　　0001 0011<<4 == 0011 0000   
　　请用C或者C++或者其他X86上能运行的程序实现。   
　　-----------------------------------------------------------------------------  
　　-----------------------------------  
　　附加题（只有在完成以上题目后，才获准回答）   
　　In C++, what does "explicit" mean? what does "protected" mean?   
　　  
　　explicit  
　　C++ Specific   
　　  
　　This keyword is a declaration specifier that can only be applied to   
　　in-class constructor declarations. Constructors declared explicit will not   
　　be considered for implicit conversions. For example:  
　　  
　　class X {   
　　public:   
　　explicit X(int); //legal   
　　explicit X(double) { //legal // ... }   
　　};  
　　 explicit X::X(int) {} //illegal  
　　An explicit constructor cannot take part in implicit conversions. It can   
　　only be used to explicitly construct an object. For example, with the class   
　　declared above:  
　　  
　　void f(X) {}   
　　void g(int I)   
　　{   
　　f(i); // will cause error   
　　}   
　　void h()   
　　{   
　　X x1(1); // legal   
　　}  
　　The function call f(i) fails because there is no available implicit   
　　conversion from int to X.  
　　  
　　Note It is meaningless to apply explicit to constructors with multiple   
　　arguments, since such constructors cannot take part in implicit conversions.  
　　  
　　END C++ Specific  
　　  
　　protected  
　　C++ Specific —>  
　　  
　　protected: [member-list]  
　　  
　　protected base-class  
　　  
　　When preceding a list of class members, the protected keyword specifies   
　　that those members are accessible only from member functions and friends of   
　　the class and its derived classes. This applies to all members declared up   
　　to the next access specifier or the end of the class.  
　　  
　　When preceding the name of a base class, the protected keyword specifies   
　　that the public and protected members of the base class are protected   
　　members of the derived class.  
　　  
　　Default access of members in a class is private. Default access of members   
　　in a structure or union is public.  
　　  
　　Default access of a base class is private for classes and public for   
　　structures. Unions cannot have base classes.  
　　  
　　For related information, see public, private, friend, and Table of Member   
　　Access Privileges.  
　　  
　　END C++ Specific  
　　  
　　Example  
　　  
　　// Example of the protected keyword   
　　class BaseClass {   
　　protected: int protectFunc();   
　　};   
　　class DerivedClass : public BaseClass   
　　{ public:   
　　int useProtect() { protectFunc(); } // protectFunc accessible from   
　　derived class   
　　};   
　　void main()   
　　{   
　　BaseClass aBase;   
　　DerivedClass aDerived;   
　　aBase.protectFunc(); // Error: protectFunc not accessible   
　　aDerived.protectFunc(); // Error: protectFunc not accessible in derived   
　　class   
　　}  
　　  
　　  
　　  
　　  
　　  
　　  
　　How do you code an infinite loop in C?   
　　2. Volatile:   
　　 a) What does the keyword volatile mean? Give an example   
　　 b) Can a parameter be both const and volatile? Give an example   
　　 c) Can a pointer be volatile? Give an example   
　　3. What are the values of a, b, and c after the following instructions:   
　　 int a=5, b=7, c;   
　　 c = a+++b;   
　　4, What do the following declarations mean?   
　　 a) const int a;   
　　 b) int const a;   
　　 c) const int \*a;   
　　 d) int \* const a;   
　　 e) int const \* a const;   
　　5. Which of the following statements describe the use of the keyword   
　　static?   
　　a) Within the body of a function: A static variable maintains its value   
　　 between function revocations   
　　 b) Within a module: A static variable is accessible by all functions   
　　 within that module   
　　 c) Within a module: A static function can only be called by other   
　　 functions within that module   
　　6. Embedded systems always require the user to manipulate bits in   
　　registers or variables. Given an integer variable a, write two code   
　　fragments.  
　　  
　　The first should set bit 5 of a. The second shnuld clear bit 5 of a. In   
　　both   
　　  
　　cases, the remaining bits should be unmodified.   
　　7. What does the following function return?   
　　 char foo(void)   
　　 {   
　　 unsigned int a = 6;   
　　 iht b = -20;   
　　 char c;   
　　 (a+b > 6) ? (c=1): (c=0);   
　　 return c;   
　　 }   
　　8. What values are printed when the following C program is executed?   
　　 int i = 8;   
　　 void main(void)   
　　 (   
　　  
　　9. What will be the output of the following C code?   
　　 main()   
　　 {   
　　 int k, num= 30;   
　　 k =(num > 5 ? (num <=10 ? 100:200): 500);   
　　 printf("%d", k);   
　　 }   
　　10. What will the following C code do?   
　　 int \*ptr;   
　　 ptr =(int \*)Ox67a9;   
　　 \*ptr = Oxaa55;   
　　11. What will be the output of the follow C code?   
　　 #define product(x) (x\*x)   
　　 main()   
　　 {   
　　 int i = 3, j, k;   
　　 j = product(i++);   
　　 k = product(++i);   
　　 printf("%d %d",j,k);   
　　 }   
　　12. Simplify the following Boolean expression   
　　 !((i ==12) || (j > 15))   
　　13. How many flip-flop circuits are needed to divide by 16?   
　　14. Provides 3 properties that make an OS, a RTOS?   
　　15. What is pre-emption?   
　　16. Assume the BC register value is 8538H, and the DE register   
　　value is 62A5H.Find the value of register BC after the following   
　　assembly operations:   
　　 MOV A,C   
　　 SUB E   
　　 MOV C,A   
　　 MOV A,B   
　　 SBB D   
　　 MOV B,A   
　　17.In the Assembly code shown below   
　　 LOOP: MVI C,78H   
　　 DCR C   
　　 JNZ LOOP   
　　 HLT   
　　How many times is the DCR C Operation executed?   
　　  
　　18.Describe the most efficient way(in term of execution time   
　　and code size) to divide a number by 4 in assembly language   
　　  
　　19.what value is stored in m in the following assembly language code   
　　fragment if n=7?   
　　 LDAA #n   
　　LABEL1: CMPA #5   
　　 BHI L3   
　　 BEQ L2   
　　 DECA   
　　 BRA L1   
　　LABEL2: CLRA   
　　LABEL3: STAA #m   
　　20. What is the state of a process if a resource is not   
　　available?   
　　#define a 365\*24\*60\*60  
　　21. Using the #define statement, how would you   
　　declare a manifest constant that returns the   
　　number of seconds in a year? Disregard leap   
　　years in your answer.   
　　22. Interrupts are an important part of embedded   
　　systems. Consequently, many compiler vendors   
　　offer an extension to standard C to support interrupts.   
　　Typically, the keyword is \_\_interrupt. The following   
　　code uses \_\_interrupt to define an interrupt service   
　　routine (ISR). Point out problems in the code.   
　　  
　　\_\_interrupt double compute\_area (double radius)   
　　{   
　　 double area = PI \* radius \* radius;   
　　 printf("\nArea = %f", area);   
　　 return area;   
　　}   
　　  
　　  
　　  
　　  
　　  
　　  
　　

**八**　  
　　1.一株查找二叉树,其结点A、B、C、D、E、F依次存放在一个起始地址为n(假定地址以字  
　　节   
　　为单位顺序编号)的连续区域中,每个节点占4个字节:前两个字节存放结点值,后两个字节  
　　依   
　　次放左指针、右指针.   
　　若该查找二叉树的根结点为E,则它的一种可能的前序遍历为\_\_\_\_ ,相应的层次遍历为\_\_\_  
　　\_   
　　.   
　　在以上两种遍历情况下,结点C的左指针LC的存放地址为\_\_\_\_\_ ,LC的内容为\_\_\_\_\_\_ 结点A  
　　的   
　　左指针RA的内容为\_\_\_\_\_\_\_.   
　　供选择的答案   
　　(1) A. EAFCBD B.EFACDB C.EABCFD D.EACBDF   
　　(2) A. EAFCBD B.EFACDB C.EABCFD D.EACBDF   
　　(3) A.n+4 B.n+10 C.n+12 D.n+13   
　　(4) A.n+9 B.n+8 C.n+12 D.n+13   
　　(5) A.n+4 B.n+8 C.n+12 D.n+16   
　　  
　　2.虚存页面调整算法有多种,\_\_\_\_\_\_ 调度算法不是页面调度算法.   
　　供选择的答案   
　　A.后进先出 B.先进先出 C.最近最少使用 D.随机选择   
　　  
　　3.在软件开发过程中常用图作为描述工具.如DFD就是面向\_\_\_\_\_\_\_分析方法的描述工具.  
　　在   
　　一套分层DFD中,如果某一张图中有N个加工(Process),则这张图允许有\_\_\_\_\_ 张子图.在  
　　一   
　　张DFD图中,任意两个加工之间\_\_\_\_\_ .在画分层DFD时,应保持\_\_\_\_\_ 之间的平衡.DFD中从  
　　系   
　　统的输出流到系统的输出流的一连串连续变换形成一种信息流,这种信息可分为\_\_\_\_\_两  
　　类   
　　.   
　　A.(1)数据结构 (2)数据流 (3)对象 (4)构件   
　　B.(1)0 (2)1 (3)1-N (4)0-N   
　　C.(1)有且仅有一条数据流   
　　(2)至少有一条数据流   
　　(3)可以有0条或多条名字互不相同的数据流   
　　(4)可以有0或多条数据流,但允许其中存若干条名字相同的数据流.   
　　D.(1)父图与其子图 (2)同一父图的所有子图 (3)不同父图的所有子图 (4)同一子图的   
　　所有直接父图.   
　　E.(1)控制流和变换流   
　　(2)变换流和事务流   
　　(3)事务流和事件流   
　　(4)事件流和控制流   
　　  
　　4.用二进制加法器对二一十进制编码的十进制数求和,当和的本位十进制数二一十进制编  
　　码   
　　小于等于1001且向高位无进位时,\_\_\_\_\_ ;当和小于等于1001且向高位存进位时,\_\_\_\_\_;当  
　　和   
　　大于1001时,\_\_\_\_\_   
　　(1)-(3) A:不需进行修改   
　　B:需进行加6修改   
　　C:需进行减6修改   
　　D:进行加6或减6修改,需进一步判别.   
　　  
　　5.www页面访问的大致过程如下:   
　　用户在浏览器中输入要访问的WWW页面的\_\_\_\_地址(http://hostname/directory/file)   
　　浏   
　　览器通过\_\_\_\_ 查询上述输入信息所指的WEB服务器的IP地址;浏览器通过网络与该IP地址  
　　处   
　　的WEB服务器的\_\_\_\_\_\_服务端之间建立一条\_\_\_\_\_\_连接;浏览器依照相关协议发送\_\_\_\_\_命  
　　令   
　　;WEB服务器根据命令取出文档,发送回来;浏览器释放连接,显示该文档.   
　　(1) A.URL B.EMS C.NDS D.DNS   
　　(2)A.NAT B.EMS C.NDS D.DNS   
　　(3)A.HTML B.HTTP C.SMTP D.SNMP   
　　(4)A.RTP B.IP C.TCP D.UDP   
　　(5)A.TCP B.GET C.UDP D.PUT   
　　  
　　6.假设某计算机具有1MB的内存(目前使用的计算机往往具有64MB以上内存),并按字节编  
　　址   
　　,为了能存取该内存各地址的内容,其地址寄存器至少需要二进制\_\_\_\_位.为使4字节组成  
　　的   
　　字段从存储器中一次读出,要求存放存储器中的字边界对齐,一个字节的地址码应\_\_\_\_\_若  
　　存   
　　储器周期为200ns,且每个周期可访问4个字节,则该存储器带宽为\_\_\_\_\_bit/s假如程序员  
　　所   
　　用的地址为\_\_\_\_\_\_,而真正访问内存的地址称为\_\_\_\_\_\_\_   
　　A.(1)10 (2)16 (3) 20 (4)32   
　　B.(1)最低两位00 (2)最低两位为10 (3)最高两位为00 (4)最高两位为10   
　　C.(1)20M (2)40M (3)80M (4)160M   
　　D.(1)有效地址 (2)程序地址 (3)逻辑地址 (4)物理地址   
　　E.(1)指令地址 (2)物理地址 (3)内存地址 (4)数据地址   
　　  
　　7.英语题   
　　Soon,more of the information we receive via the internet could come \_\_\_\_\_in   
　　di   
　　gital wrappers.   
　　Wrappers are made up \_\_\_\_\_\_ softwore code that’s targeted to do specific   
　　thin   
　　gs with the data \_\_\_\_\_within them such as helping to define queries for   
　　search   
　　engines They also keep \_\_\_\_\_from\_\_\_\_\_access to that code.   
　　(1) A.Package B.packaged C.packages D.packaging   
　　(2)A.of B.off C.on D.out   
　　(3)A.close B.closed C.enclose D.enclosed   
　　(4)A.insiders B.money C.outsiders D.warehouse   
　　(5)A.gain B.gained C.gains D.gaining   
　　  
　　二.设计题   
　　1.在VC中怎样获得父窗口的指针(写出代码)   
　　2.怎样创建一个临时文件   
　　3.怎样获得状态栏和工具栏的指针.   
　　4.访问控件存几种方法   
　　  
　　三.填空题:   
　　1.ODBC的数据类型分为\_\_\_\_\_\_\_\_\_和\_\_\_\_\_\_\_\_\_   
　　2.VC访问数据库的方式\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_   
　　3.VC的线路分为\_\_\_\_\_\_\_\_\_和\_\_\_\_\_\_\_\_\_,它是用什么对象表示的\_\_\_\_\_\_\_\_\_   
　　4.下列中a的值是\_\_\_\_\_\_\_\_\_   
　　#define AAA 200   
　　#define BBB AAA+100   
　　int a= BBB\*2   
　　  
　　  
　　  
　　  
　　  
　　试题一：基础知识   
　　1、从供选择的答案中，选出应填入下面叙述中\_?\_内的最确切的解答，把相应编号写在  
　　答   
　　卷的对应栏内。   
　　假设某计算机具有1M 字节的内存（目前使用的计算机往往具有64M字节以上的内存），  
　　并   
　　按字节编址，为了能存取该内存各地址的内容，其地址寄存器至少需要二进制\_A\_位。为  
　　使   
　　4字节组成的字能从存储器中一次读出，要求存放在存储器中的字边界对齐，一个字的地  
　　址   
　　码应\_B\_。若存储周期为200NS，且每个周期可访问4个字节，则该存储器带宽为\_C\_BIT/S  
　　。   
　　假如程序员可用的存储空间为4M字节，则程序员所用的地址为\_D\_，而真正访问内存的地  
　　址   
　　称为\_E\_。   
　　供选择的答案：   
　　A： ①10 ②16 ③20 ④32   
　　B： ①最低两位为00 ②最低两位为10 ③最高两位为00 ④最高两位为10   
　　C： ①20M ②40M ③80M ④160M   
　　D： ①有效地址 ②程序地址 ③逻辑地址 ④物理地址   
　　E： ①指令 ②物理地址 ③内存地址 ④数据地址   
　　  
　　2、从供选择的答案中。选出应填入下面叙述中＿?＿内的最确切的解答，把相应编号写  
　　在   
　　答卷的对应栏内。   
　　给定结点的关键字序列（Ｆ、Ｂ、Ｊ、Ｇ、Ｅ、Ａ、Ｉ、Ｄ、Ｃ、Ｈ），对它按字母的  
　　字   
　　典顺序进行排列，采用不同方法，其最终结果相同。但中间结果是不同的。   
　　Shell排序的第一趟扫描（步长为５）结果应为＿Ａ＿。   
　　冒泡排序（大数下沉）的第一趟起泡的效果是＿B＿3．   
　　快速排序的第一趟结果是＿Ｃ＿。   
　　二路归并排序的第一趟结局是 ＿D＿。   
　　供选择的答案   
　　Ａ：①（B、F、G、J、A、D、I、E、H、C）   
　　②（B、F、G、J、A、E、D、I、C、H）   
　　③（A、B、D、C、E、F、I、J、G、H）   
　　④（C、B、D、A、E、F、I、G、J、H）   
　　Ｂ：①（A、B、D、C、F、E、I、J、H、G）   
　　②（A、B、D、C、E、F、I、H、G、J）   
　　③（B、F、G、E、A、I、D、C、H、J）   
　　④（B、F、G、J、A、E、D、I、C、H）   
　　Ｃ：①（C、B、D、A、F、E、I、J、G、H）   
　　②（C、B、D、A、E、F、I、G、J、H）   
　　③（B、A、D、E、F、G、I、J、H、C）   
　　④（B、C、D、A、E、F、I、J、G、H）   
　　Ｄ：①（B、F、G、J、A、E、D、I、G、H）   
　　②（B、A、D、E、F、G、I、J、H、C）   
　　③（A、B、D、C、E、F、I、J、G、H）   
　　④（A、B、D、C、F、E、J、I、H、C）   
　　  
　　3、从供选择的答案中，选出应填入下面叙述中＿？＿内的最确切的解答．把相应编号写  
　　在   
　　答卷的对应栏内。   
　　进程是操作系统中的一个重要概念。进程是一个具有一定独立功能的程序在某个数据集  
　　合   
　　上的一次＿A2＿。   
　　进程是一个＿B3＿的概念，而程序是一个＿C3＿的概念。   
　　进程的最基本状态有＿D4＿。在一个单处理机中，若有6个用户进程，在非管态的某一时  
　　刻   
　　，处于就绪状态的用户进程最多有＿E5＿个。   
　　供选择的答案   
　　A：①单独操作 ②关联操作 ③运行活动 ④并发活动   
　　B：①静态 ②动态 ③逻辑 ④物理   
　　C：①物理 ②逻辑 ③动态 ④静态   
　　D：①就绪、运行、隐蔽 ②停止、就绪、运行   
　　③运行、就绪、阻塞 ④就绪、撤消、运行   
　　E：①5 ②6 ③1 ④4   
　　  
　　4、软件设计中划分模块的一个准则是＿A2＿。两个模块之间的耦合方式中，＿Ｂ3＿耦  
　　合   
　　的耦合度最高，＿Ｃ4＿耦合的耦合度最低。一个模块内部的内聚种类中＿Ｄ4＿内聚的  
　　内   
　　聚度最高，＿E1＿内聚的内聚度最低。   
　　供选择的答案   
　　A：①低内聚低耦合②低内聚高耦合③高内聚低耦合④高内聚高耦合   
　　B：①数据 ②非直接 ③控制 ④内容   
　　C：①数据 ②非直接 ③控制 ④内容   
　　D：①偶然 ②逻辑 ③功能 ④过程   
　　E：①偶然 ②逻辑 ③功能 ④过程   
　　  
　　5、从供选择的答案中选出应填入下面叙述中＿？＿内的最确切的解答，把相应编号写在  
　　答   
　　卷的对应栏内。   
　　最常用的一种基本数据模型是关系数据模型，它用统一的＿A 1＿结构来表示实体及实体  
　　之   
　　间的联系。关系数据库的数据操作语言（ＤＭＬ）主要包括＿Ｂ2＿两类操作。   
　　关系运算以关系代数为理论基础，关系代数的最基本操作是并、差、笛卡尔积、＿Ｃ4＿  
　　。   
　　用Ｒ∣×∣Ｓ 表示关系 Ｒ和关系 Ｓ的＿Ｄ1＿。   
　　设关系Ｒ和关系Ｓ图示如下：   
　　Ｒ： A B C S: B C D T: A B C D   
　　则关系Ｔ是关系Ｒ和关系Ｓ＿E ＿的结果。   
　　供选择的答案   
　　A：①树 ②网络 ③图 ④二维表   
　　B：①插入和删除 ②检索和更新 ③查询和编辑 ④统计和修改   
　　C：①投影、联接 ②联接、选择 ③选择、投影 ④交、选择   
　　D：①联接 ②笛卡尔积 ③日联接 ④自然联接   
　　E:①自然联接 ②θ联接 ③笛卡尔积 ④并   
　　  
　　  
　　  
　　  
　　试题二：程序设计   
　　1、用你所熟悉的任意一种程序语言，编写一个完整的过程，将一个字符串插入到另一个  
　　字   
　　符串的某个位置后面（例如：将“abc”插入到“abcdef”的第三个字符位置后面，结果  
　　为   
　　“abcabcdef”）。编写程序时，请在必要的地方加以注释(注：不能用该程序语言的内  
　　置   
　　函数或过程)。   
　　  
　　  
　　2、用你所熟悉的任意一种程序语言，编写一个完整的过程，完成从一个给定的完整的文  
　　件   
　　路径（如“C:\My Documents\Software Test 1.00.doc”)中，析取文件名，扩展名和文  
　　件   
　　所处目录的功能，编写程序时，请在必要的地方加以注释(注：不能用该程序语言的内置  
　　函   
　　数或过程)。   
　　  
　　  
　　  
**九**

编写类String 的构造函数,析构函数和赋值函数   
　　已知类String 的原型为   
　　class string   
　　{   
　　public:   
　　string(const char \*str=null);//普通构造函数   
　　string(const string &other);//拷贝构造函数   
　　---string(void);   
　　string &operate=(const string &other);//赋值函数   
　　private:   
　　char \* m-data;//用于保存字符串   
　　};   
　　请编写string 的上述4个函数   
　　三. 有关内存的思考题   
　　1. void getmemory(char \*p)   
　　{ p=(char\*)mallol(100);   
　　}   
　　void test(void)   
　　{   
　　char \* str =null;   
　　getmemory(str);   
　　strcpy(str,”hello,world”);   
　　printf(str);   
　　}   
　　请问运行Test函数会有什么样的结果   
　　2. char\*getmemory(void)   
　　{ char p[]=”hello world”;   
　　return p;   
　　}   
　　void test(void)   
　　{   
　　char \*str=null;   
　　str=Getmemory();   
　　printf(str);   
　　} 请问运行Test 函数会有什么样的结果.

**十**

　C++/C试题   
　　本试题仅用于考查C++/C程序员的基本编程技能。内容限于C++/C常用语法，不涉及数据  
　　结   
　　构、算法以及深奥的语法。考试成绩能反映出考生的编程质量以及对C++/C的理解程度，  
　　但   
　　不能反映考生的智力和软件开发能力。   
　　笔试时间90分钟。请考生认真答题，切勿轻视。   
　　  
　　一、请填写BOOL , float, 指针变量 与“零值”比较的 if 语句。（10分）   
　　提示：这里“零值”可以是0, 0.0 , FALSE或者“空指针”。例如 int 变量 n 与“零  
　　值   
　　”比较的 if 语句为：   
　　if ( n == 0 )   
　　if ( n != 0 )   
　　以此类推。   
　　  
　　请写出 BOOL flag 与“零值”比较的 if 语句：   
　　请写出 float x 与“零值”比较的 if 语句：   
　　请写出 char \*p 与“零值”比较的 if 语句：   
　　  
　　二、以下为Windows NT下的32位C++程序，请计算sizeof的值（10分）   
　　  
　　char str[] = “Hello” ; char \*p = str ;int n = 10;请计算sizeof (str )   
　　= sizeof ( p ) = sizeof ( n ) = void Func (   
　　char str[100]){请计算 sizeof( str ) = }   
　　void \*p = malloc( 100 );请计算sizeof ( p ) =   
　　  
　　三、简答题（25分）   
　　  
　　1、头文件中的 ifndef/define/endif 干什么用？   
　　  
　　  
　　  
　　2、#include 和 #include “filename.h” 有什么区别？   
　　  
　　  
　　  
　　3、const 有什么用途？（请至少说明两种）   
　　  
　　  
　　  
　　4、在C++ 程序中调用被 C编译器编译后的函数，为什么要加 extern “C”声明？   
　　  
　　  
　　  
　　  
　　5、请简述以下两个for循环的优缺点   
　　  
　　// 第一个for (i=0; i ing();} // 第二个if (condition){for (i=0; i for (i=0;   
　　i 优点：缺点： 优点：缺点：   
　　  
　　四、有关内存的思考题（20分）   
　　  
　　void GetMemory(char \*p){p = (char \*)malloc(100);}void Test(void) {char \*str   
　　=   
　　NULL;GetMemory(str); strcpy(str, "hello world");printf(str);}请问运行Test函数  
　　会   
　　有什么样的结果？答： char \*GetMemory(void){ char p[] = "hello world";return   
　　p;   
　　}void Test(void){char \*str = NULL;str = GetMemory(); printf(str);}请问运行Tes  
　　t   
　　函数会有什么样的结果？答：   
　　Void GetMemory2(char \*\*p, int num){\*p = (char \*)malloc(num);}void   
　　Test(void){c   
　　har \*str = NULL;GetMemory(&str, 100);strcpy(str, "hello"); printf(str); }请问  
　　运   
　　行Test函数会有什么样的结果？答： void Test(void){char \*str = (char \*)   
　　malloc(1   
　　00); strcpy(str, “hello”); free(str); if(str != NULL) { strcpy(str, “   
　　world”); printf(str);}}请问运行Test函数会有什么样的结果？答：   
　　  
　　  
　　五、编写strcpy函数（10分）   
　　已知strcpy函数的原型是   
　　char \*strcpy(char \*strDest, const char \*strSrc);   
　　其中strDest是目的字符串，strSrc是源字符串。   
　　  
　　（1）不调用C++/C的字符串库函数，请编写函数 strcpy   
　　  
　　  
　　  
　　  
　　  
　　  
　　  
　　（2）strcpy能把strSrc的内容复制到strDest，为什么还要char \* 类型的返回值？   
　　  
　　  
　　  
　　  
　　六、编写类String的构造函数、析构函数和赋值函数（25分）   
　　已知类String的原型为：   
　　class String   
　　{   
　　public:   
　　String(const char \*str = NULL); // 普通构造函数   
　　String(const String &other); // 拷贝构造函数   
　　~ String(void); // 析构函数   
　　String & operate =(const String &other); // 赋值函数   
　　private:   
　　char \*m\_data; // 用于保存字符串   
　　};   
　　请编写String的上述4个函数。   
　　  
　　  
　　  
　　盛大游戏策划笔试题目！  
　　  
　　  
　　1.游戏策划的关键词   
　　2.游戏运营与研发之间的关系   
　　3.设计策划案目录一个   
　　4.新的赢利方式《除点卡和出卖游戏物品两种之外》   
　　5.游戏人群的特征---白领。女人。小孩。骨灰级玩家   
　　6.1）给一个旧魔法，设计一个新魔法（具体不说了）   
　　2）给《坦克大战》设计一个开房间类的策划方案   
　　3）地图的不平衡，造成法师升级速度快，策划设计一个新地图，使战士的升级速度加   
　　快！

**十一**

　　一. int a[2][3]={1,2,3,4,5};   
　　问：a[1][2]=?   
　　二。char \*p="hello";   
　　printf("%s",p);   
　　p++;   
　　printf("%s",p);   
　　printf("%c",\*p);   
　  
**十二**

　1．进程和线程的差别。   
　　  
　　2．Heap与stack的差别。   
　　  
　　3．Windows下的内存是如何管理的？   
　　  
　　4．介绍.Net和.Net的安全性。   
　　  
　　5．客户端如何访问.Net组件实现Web Service？   
　　  
　　6．C/C++编译器中虚表是如何完成的？   
　　  
　　7．谈谈COM的线程模型。然后讨论进程内/外组件的差别。   
　　  
　　8．谈谈IA32下的分页机制。   
　　  
　　9．给两个变量，如何找出一个带环单链表中是什么地方出现环的？   
　　  
　　10．在IA32中一共有多少种办法从用户态跳到内核态？   
　　  
　　11．如果只想让程序有一个实例运行，不能运行两个。像winamp一样，只能开一个窗口  
　　，怎样实现？   
　　  
　　12．如何截取键盘的响应，让所有的‘a’变成‘b’？   
　　  
　　13．Apartment在COM中有什么用？为什么要引入？   
　　  
　　14．存储过程是什么？有什么用？有什么优点？   
　　  
　　15．Template有什么特点？什么时候用？   
　　  
　　16．谈谈Windows DNA结构的特点和优点

**十三**

1．1000！有几位数，为什么？   
　　  
　　2．F（n）=1 n>8 n<12   
　　  
　　F（n）=2 n<2   
　　  
　　F（n）=3 n=6   
　　  
　　F（n）=4 n=other   
　　  
　　使用+ - \* /和sign（n）函数组合出F（n）函数   
　　  
　　sign（n）=0 n=0   
　　  
　　sign（n）=-1 n<0   
　　  
　　sign（n）=1 n>0   
　　  
　　3．编一个程序求质数的和，例如F（7）=1+3+5+7+11+13 +17=57。

一:

1. new有几种用法   
   第一种:new Class();   
   第二种:覆盖方法   
   public new XXXX(){}   
   第三种:new 约束指定泛型类声明中的任何类型参数都必须有公共的无参数构造函数。   
   2.如何把一个array复制到arrayList里   
   foreach( object o in array )arrayList.Add(o);   
   3.datagrid.datasouse可以连接什么数据源 [dataset,datatable,dataview]   
   dataset,datatable,dataview , IList   
   4.概述反射和序列化   
   反射:程序集包含模块，而模块包含类型，类型又包含成员。反射则提供了封装程序集、模块和类型的对象。您可以使用反射动态地创建类型的实例，将类型绑定到现有对象，或从现有对象中获取类型。然后，可以调用类型的方法或访问其字段和属性   
   序列化:序列化是将对象转换为容易传输的格式的过程。例如，可以序列化一个对象，然后使用 HTTP 通过 Internet 在客户端和服务器之间传输该对象。在另一端，反序列化将从该流重新构造对象。   
   5.概述o/r mapping 的原理   
   利用反射，配置 将类于数据库表映射   
   6.类成员有( )种可访问形式   
   可访问形式？不懂。   
   可访问性：public ,protected ,private,internal   
   7.用sealed修饰的类有什么特点   
   sealed 修饰符用于防止从所修饰的类派生出其它类。如果一个密封类被指定为其他类的基类，则会发生编译时错误。   
   密封类不能同时为抽象类。   
   sealed 修饰符主要用于防止非有意的派生，但是它还能促使某些运行时优化。具体说来，由于密封类永远不会有任何派生类，所以对密封类的实例的虚拟函数成员的调用可以转换为非虚拟调用来处理。   
   8.列举ADO.NET中的五个主要对象，并简单描述   
   connection,command,dataReader,trans,dataset ...   
   9.执行下面代码后：   
   String strTemp =”abcdefg 某某某”；   
   Int i System.Text.Encoding.Default.GetBytes(strTemp).Length;   
   Int j = strTemp.Length;   
   问：i=(14 ) ；j=(11 )   
   i=(14 ) ；j=(11 ) 中文两个字节   
   10.C#中，string str = null 与 string str =””，请尽量用文字说明区别。(要点：说明详细的内存空间分配)   
   string str =”” 分配空间   
   11.详述.NET里class和struct的异同！   
   class:放在 ? struct放在？   
   struct值传递   
   类与结构有很多相似之处：结构可以实现接口，并且可以具有与类相同的成员类型。然而，结构在几个重要方面不同于类：结构为值类型而不是引用类型，并且结构不支持继承。结构的值存储在“在堆栈上”或“内联”。细心的程序员有时可以通过聪明地使用结构来增强性能。   
   12.概述.NET里对 remoting 和 webservice 两项技术的理解和实际中的应用。   
   远程逻辑调用，remoing接口只能用在.net中   
   13.什么是code-behind技术   
   aspx and cs   
   14.概述三层结构体系   
   web/business/dataaccess   
   15.asp.net如何实现MVC模式，举例说明！   
   web/business/dataaccess

二

1. 面向对象的思想主要包括什么？   
   答:这个题范围太广,不知道说什么.   
   2.什么是ASP.net中的用户控件   
   答:用户控件就是.ascx扩展名的东西喽,可以拖到不同的页面中调用,以节省代码.比如登陆可能在多个页面上有,就可以做成用户控件,但是有一个问题就是用户控件拖到不同级别的目录下后里面的图片等的相对路径会变得不准确,需要自已写方法调整.   
   3.什么叫应用程序域？什么是受管制的代码？什么是强类型系统？什么是装箱和拆箱？什么是重载？CTS、CLS和CLR分别作何解释？   
   答:装箱就是把值类型转成引用类型,从MS IL角度看好像是boxing,没记错的话是把值从堆栈转到堆中.拆箱相反,重载就是指一个方法名同,参数个数不同,返回值可以相同的方法.CLR是通用语言运行时,其它的不清楚.   
   4.列举一下你所了解的XML技术及其应用   
   答:XML可是好东西,保存配置,站与站之间的交流,WEB SERVICE都要用它.   
   5.值类型和引用类型的区别？写出C#的样例代码。   
   答:结构是值类型,类是引用类型,所以传结构就是值类型的应用啦,传对象或类就是引用类型的,这个不用多写了吧.   
   6.ADO.net中常用的对象有哪些？分别描述一下。   
   答:connection command sqladapter dataset datatable dataview等等.写不完了.   
   7.如何理解委托？   
   答:据说相当于函数指针,定义了委托就可以在不调用原方法名称的情况下调用那个方法.   
   msdn2005中是这样解释的:   
   委托具有以下特点：   
   委托类似于 C++ 函数指针，但它是类型安全的。   
   委托允许将方法作为参数进行传递。   
   委托可用于定义回调方法。   
   委托可以链接在一起；例如，可以对一个事件调用多个方法。   
   方法不需要与委托签名精确匹配。有关更多信息，请参见协变和逆变。   
   C# 2.0 版引入了匿名方法的概念，此类方法允许将代码块作为参数传递，以代替单独定义的方法。   
   8.C#中的接口和类有什么异同。   
   答:这个异同可多了,要说清楚还真不容易.   
   9.。net中读写数据库需要用到哪些类？他们的作用   
   答:这个类自已可以写的啊,你是指基类吗?那configuration,sqlconnection,sqlcommand等都要用到.   
   10.UDP连接和TCP连接的异同。   
   答:前者只管传,不管数据到不到,无须建立连接.后者保证传输的数据准确,须要连结.

**三**

1. Write a script in PHP and Javascript to accept two textarea's, slot and alternative by the name.  
    + slot can contain many a lines, each line includes many words and is ended by a number  
    + the chars in the slot can only be either alpha-numeric chars or squares or underlines  
    + if a word in the slot is surrounded by underlines, it must start a new line in the alternatives, followed by the detailed definition of the word  
    + the square brackets should come in pairs  
 for example, the following is legal slot and alternative:  
          SLOT                                ALTERNATIVE  
    ----------------------           --------------------------  
    | this is 1          |           | \_this\_ th ee s         |  
    | the is 1           |           |                        |  
    | this's 2           |           |                        |  
    | \_this\_ 2           |           |                        |  
    ----------------------           --------------------------  
  
2. A directory contains many a numbered sub-dir, and other files or directories as well. Each numbered sub-dir contains pictures(only jpg or png) and other files. Some of the pictures are original, while the others are the result of conversion from the original(jpg.jpg or png.png or jpg.png). Write a BASH command/script to list all those orphaned pictures whose original counterparts are missing.  
  
3. There are two tables: students, and scores, in the mysql database (4.0+) exactly like this:  
  
    SN | Name                         SN | Score  
    -------------                    -------------  
     1 | Jerry                         1 | 48  
    -------------                    -------------  
     2 | Lily                          2 | 50  
    -------------                    -------------  
     3 | Roy  
    -------------  
  where the SN is the foreigh key. Write an SQL statement to delete all the names that have no score.  
  
4. Make a function to print an HTML table of r rows and c cols; fill the cells with up to n consecutive numbers (if n is greater than the produce of r and c, discard the rest). Ensure that all rows have the same number of columns.  
  
5. Write some bash commands to list all the php scripts (\*.php) in the current directory and its sub-directory. Try to use as many different methods as possible.  
  
6. Write a function to resize an image to be less than 660x570px. Be sure to keep the original ratio of width to height.  
  
7. Write a script to reverse (interchange 1 with 0) the n-th bit of a m-bit binary number.  
  
8. Use regex to remove all html tags from a file.

**四**

1. 分别用PHP和Javascript来验证两个分别叫slot和alternative的文本框里面的内容：  
    + slot可以包含许多行，每行又包含若干个单词，并且以数字结尾  
    + slot中的每个单词只能由数字、字母、方括号和下划线构成  
    + 如果slot中的某个单词前后各有相同书面的下划线，则这个单词必须在alternatives另起一行，并带有其定义  
    + 方括号必须成对出现  
  例如，下面的slot和alternative是正确的：  
          SLOT                                ALTERNATIVE  
    ----------------------           --------------------------  
    | this is 1          |           | \_this\_ th ee s         |  
    | the is 1           |           |                        |  
    | this's 2           |           |                        |  
    | \_this\_ 2           |           |                        |  
    ----------------------           --------------------------  
  
2. 一个目录里面有若干个以数字命名的子目录和其他子目录。每个以数字命名的子目录里面有若干（JPG或者PNG格式的）图片以及其他文件。这些图片中有的是原始的图片，有的是从原始图片生成的（生成以后的扩展名是png.png或者jpg.jpg或者png.jpg）。使用BASH命令或者脚本删除哪些没有对应原始图片的导出图片。  
  
3. 在一个mysql 4.0+的数据库中有两个表，学生表和成绩表，分别如下所示：  
  
    SN | Name                         SN | Score  
    -------------                    -------------  
     1 | Jerry                         1 | 48  
    -------------                    -------------  
     2 | Lily                          2 | 50  
    -------------                    -------------  
     3 | Roy  
    -------------  
   其中SN是联系两个表的外键。用带有LEFT JOIN的SQL语句删除哪些没有对应成绩的学生。  
4. 用函数打印一个r行c列的HTML表格，并在单元格中依次填入前n个自然数。如果n大于单元格的总数，以单元格总数为准，去掉多余的部分。注意使得每行都有相同的单元格。  
5. 用BASH和perl命令（脚本），列出当前目录及其子目录里面所有扩展名为.php的文件。尽量使用多种不同的方法。  
6. 写一个函数，将一个图片缩放成大小不超过660x570px。注意保持长与宽的比例不变。  
7. 用函数实现将一个m位二进制串中的第n位置反。  
8. 用正则表达式除去一个文件中所有HTML标记。

**五**

\* 1.在PHP中，当前脚本的名称（不包括路径和查询字符串）记录在预定义变量

（1）中；而链接到当前页面的URL记录在预定义变量（2）中。

2.执行程序段<?php echo 8%(-2) ?>将输出（3）。

3.在HTTP 1.0中，状态码 401 的含义是（4）；如果返回“找不到文件”的提示，则可用 header 函数，其语句为（5）。

4.数组函数 arsort 的作用是（6）；语句 error\_reporting(2047)的作用是（7）。

5.PEAR中的数据库连接字符串格式是（8）。

6.写出一个正则表达式，过虑网页上的所有JS/VBS脚本（即把script标记及其内容都去掉）：（9）。

7.以Apache模块的方式安装PHP，在文件http.conf中首先要用语句（10）动态装载PHP模块，然后再用语句（11）使得Apache把所有扩展名为php的文件都作为PHP脚本处理。

8.语句 include 和 require 都能把另外一个文件包含到当前文件中，它们的区别是（12）；为了避免多次包含同一文件，可以用语句（13）来代替它们。

9.类的属性可以序列化后保存到 session 中，从而以后可以恢复整个类，这要用到的函数是（14）。

10.一个函数的参数不能是对变量的引用，除非在php.ini中把（15）设为on.

11.SQL 中LEFT JOIN的含义是（16）。如果 tbl\_user记录了学生的姓名(name)和学号(ID)，tbl\_score记录了学生（有的学生考试以后被开除了，没有其记录）的学号(ID)和考试成绩(score)以及考试科目(subject)，要想打印出各个学生姓名及对应的的各科总成绩，则可以用SQL语句（17）。  
12.在PHP中，heredoc是一种特殊的字符串，它的结束标志必须（18）。

**六**

1. 写一个函数，能够遍历一个文件夹下的所有文件和子文件夹。14.简述论坛中无限分类的实现原理。15.设计一个网页，使得打开它时弹出一个全屏的窗口，该窗口中有一个文本框和一个按钮。用户在文本框中输入信息后点击按钮就可以把窗口关闭，而输入的信息却在主网页中显示。
2. 1、用[**PHP**](javascript:;)打印出前一天的时间格式是2006-5-10 22:21:21  
   2、echo(),print(),print\_r()的区别  
   3、能够使[**HTML**](javascript:;)和PHP分离开使用的模板  
   4、如何实现PHP、JSP交互？  
   5、使用哪些工具进行版本控制？  
   6、如何实现字符串翻转？  
   7、优化[**MYSQL**](javascript:;)数据库的方法。  
   8、谈谈事务处理  
   9、apache+mysql+php实现最大负载的方法  
   10、实现中文字串截取无乱码的方法

**七**

Q1. 做一个提供不同数据给不同用户的个人主页，什么URL/CGI技术能够鉴别每一个用户? 这些技术的优点和缺点是什么？

Q2. Assuming we had a continuous feed of data that was being piped into

site, and it needed to distributed to up to 10 different machines in

real time. Describe at least one method you would use?

**八**

1.用来打开与关闭数据库连接的是哪个ADO对象？

A.Command B.Recourdest

C.Connection D.Field

2.若要将网页重新导向，而且要保留所有内置对象的值，那么必须使用哪个方法？

A.Execute B.URLEncode C.Transfer D.MapPath

3.若要将字符串进行编码，使它不会被浏览器解释为HTML语法，可以使用哪个方法？

A.HTMLEncodeing B.URLEencode C.MapEncode D.ASPEncode

4.若要找出父目录的实际路径，可以使用下列哪种语法？

A.Server.MapPath（“/”） B.Server.MapPath（“./”）

C.Server.MapPath（“../”） D.Server.MapPath（“../”）

5.用来读取、插入、删除或更新表记录的是哪个ADO对象？

A.Command B.Recourdest

C.Connection D.Field

6.若要复制文件夹，可以使用下列哪个方法？（复选）

A.FileSystemObject服务器组件的CopyFolder方法

B.FileSystemObject服务器组件的Copy方法

C.Folder服务器组件的CopyFolder方法

D.Folder服务器组件的Copy方法

7.objFSO.OpenTextFile（“a.txt”，1，True）的第三个参数意义是什么？

A.以只读的方式打开文本文件 B.以附加到文件后面的方式打开文本文件

C.若打开的文件不存在，就建立 D.若打开的文件不存在，仍不建立

8.objFSO.OpenAsTextStream（3，0）的第一个参数的意义是什么？

A.以只读的方式打开文本文件 B.以附加到文件后面的方式打开文本文件

C.文本文件的格式为Unicode D.文本文件的格式为ASCII

9.若要从文本文件读取字符，可以使用哪个方法？

A.Read B.ReadChar C.ReadLine D.ReadAII

10.若要在文本文件写入空行，可以使用哪个方法？

A.Write B.WriteChar C.WriteLine D.WriteBlankLines

11.下列语句哪一个是错误的？（复选）

A.Cookie的优点之一是存放在浏览器端，不会占用服务器端的空间。

B.服务器端文件存取即使在数据量很大时，也不会影响到存取效率。

C.数据库适合记录大量数据，可做读取、插入、删除、更新与查询。

D.打开数据库连接所花费的时间比查询少。

12.若要移动到表的最后一条记录，可以使用哪个方法？

A.Move B.Move Next

C.MoveFirst D.MoveLast

13.我们可以使用哪个方法筛选表的记录？

A.Sort B.Filter

C.Open D.Execute

14.Access数据库可能由一个或多个表所构成，对不对？

A.是 B.否

15.下列哪一个代表表的最后一条记录？

A.EOF B.FOF

C.BOF D.ROF

**九**

1.在记事本中,使用HTML语言编辑一个以表格布局,内有文字,图片和超链接的网页..

2.对思考题中的所建立的网页添加CSS控制其显视效果.

**十**

Web&JDBC 考试上机题

问题描述:

学生在每个学期都会选择一些课程，每门通过考试后获得相应的分数。学生的信息通常包括学号，姓名，出生日期，住址，联系电话，email。而课程信息包含课程编号，课程名称，课程描述。

所有程序必须在exam.java.advanced包下。

要求:

1. 按如下类图写出相应数据库建表sql脚本。 其中Student和Score是1对多的关系，Scroe和Course是多对1的关系。

Student

-id: String

-name: String

-birthday: Date

-address: String

-phone: String

-email: String

Score

-student: Student

-course: Course

-grade: float

Course

-id: String

-name: String

-description: String

2. 使用jdbc编写class EnrollmentImpl实现接口Enrollment:

public interface Enrollment{

public void createStudent(Student student) throws Exception;

pubic void createCourse(Course course) throws Exception;

public void enroll(Student student, Course course) throws Exception;

public void exam(Student, Course course, float grade) throws Exception;

}

3. 用jsp实现用户图形界面，servlet作为控制器， 调用EnrollmentImpl实现接口Enrollment的任一方法。

4. 用struts架构完成与3同样的功能。

**北承题目(C++)**

1. 说出你对remoting 和webservice的理解和应用
2. Asp.net的身份验证方式有哪些，原理是什么？
3. 怎样实现无对话框的文件下载
4. 如何实现一个自定义类的序列化
5. Hashtable 添加内容的方式有哪几种，有什么区别？
6. 试用客户端脚本实现“全选“的功能，即把页面的所有Checkbox都选中
7. 不通过HtmlInputFile之类的Htmlcontrol,webControls控件对象怎样实现文件的上传
8. 写出程序运行的结果

Public class Base

Public virtual string Hello() {return “Base”;}

Public class Sub:Base

Public override string Hello() {return “Sub”;}

1. Base b = new Base(); b.Hello;
2. Sub s = new Sub(); s.Hello;
3. Base b = new Sub (); b.Hello;
4. Sub s = new Base(); s.Hello;

一．ASP.NET题目：

1. 接口和抽象类的区别？
2. 解释DataSet(ds) 和 ds as DataSet 的含义？
3. WebService是什么意思？什么情况下用WebService？
4. ASP.NET中内置对象是什么？
5. OLEDBConnection 和SQLConnection 有什么区别？
6. 采用怎样的方法保证数据的完整性？

二．数据库题目：

1．查询记录（注：题目是公司给的，我没记，不是很难这题）

2．修改记录

3．删除记录

1. application与session的区别

答：application公有的，所以可以用来做聊天室；session私有的，在一次会话后消失

1. 用javascript写一个脚本，以确认用户输入

答：if(confirm()){}else{}

1. viewstate的用法以及button的causevalide的作用

答：Viewstate是保存状态的一种机制，EnableViewState属性设置为false即可禁用。

控制按钮是否导致激发验证。

1. 数据库设计
2. 说出一到两个你写过的自定义控件以及他的作用，并说明他使用的基类和方法

答：自定义控件一般从以下几个基类(此处不包含数据控件)

* 1. Control类(所有服务器控件的基类,算是比较底层的类,如果控件功能比较简单,要求不多,可直接继承此类.)
  2. WebControl类(标准控件的基类,继承此类,你可以继承其丰富的公共属性,若标准控件中的控件没有你需要的控件,你可以继承此类)
  3. CompositeControl 类(2.0新增的类,此类继承自WebControl类,如果你需要制作复合控件,请从继承此类开始)
  4. 直接从内置控件继承

1. XML的XSLT的用法

答：XSLT是把一种XML文档转换成另一种文档的语言。XPath是在XML文档中定位信息的语言。在转换过程中，XSLT使用XPath来定位源文档中匹配一个或多个预定义模板的部分。当一个匹配被找到后，XSLT将转换这一匹配的部分到结果文档中。源文档中无法匹配模板的部分将在结果文档中不做任何修改。

1. 数据流有什么要素

答：数据流是一个按照时间递增顺序排列的无穷序列,可以表示为： I = α1， α2 ，…，αt的形式,αt 是时刻t 出现的序列元素。数据流与一般的数据的区别在于它的到达是快速的，无界的，时变的和不可预测的，从而不可能将原始数据流中的数据完全存储。数据流模型能够应用到各种数据类型如电话记录，web文档，网络流量管理等

要素：

1. 检查是否每个加工都至少有一个输入数据流，一个输出数据流
2. 检查是否每一个图形符号都有相应的标记
3. 加工没有标记无法对应子图
4. 数据流没有标记无法表明数据流的含义
5. 无法对应数据字典（每一个图形符号都将在数据字典中进行描述）
6. 关注于数据的流动，而不是值的流动
7. 检查：流动的应该是“变量”而不是变量的“值”
8. 检查父图和子图的平衡
9. 分解的深度与层次要适当

考试时间为30分钟

还剩： 秒

在线应聘

赵永强你好！请认真填写下面的试题！ (每题5分)

(1).在对SQL Server 数据库操作时应选用（A ）。

A) SQL Server .NET Framework 数据提供程序；

B) OLE DB .NET Framework 数据提供程序；

C) ODBC .NET Framework 数据提供程序；

D) Oracle .NET Framework数据提供程序；

(2).下列选项中，（C ）是引用类型。

A) enum类型

B) struct类型

C) string类型

D) int类型

(3).关于ASP.NET中的代码隐藏文件的描述正确的是（c）

A) Web窗体页的程序的逻辑由代码组成，这些代码的创建用于与窗体交互。编程逻辑唯一与用户界面不同的文件中。该文件称作为“代码隐藏”文件，如果用C＃创建，该文件将具有“.ascx.cs”扩展名。

B) 项目中所有Web窗体页的代码隐藏文件都被编译成.EXE文件。

C) 项目中所有的Web窗体页的代码隐藏文件都被编译成项目动态链接库（.dll）文件。

D) 以上都不正确。

(4).C#的数据类型有（B ）

A) 值类型和调用类型；

B) 值类型和引用类型；

C) 引用类型和关系类型；

D) 关系类型和调用类型；

(5).下列描述错误的是（ D）

A) 类不可以多重继承而接口可以；

B) 抽象类自身可以定义成员而接口不可以；

C) 抽象类和接口都不能被实例化；

D) 一个类可以有多个基类和多个基接口；

(6).下列关于构造函数的描述正确的是（C ）

A) 构造函数可以声明返回类型；

B) 构造函数不可以用private修饰；

C) 构造函数必须与类名相同；

D) 构造函数不能带参数；

(7).以下是一些C#中的枚举型的定义，其中错误的用法有（ c） 在2005中都没问题啊？？？

A) public enum var1{ Mike = 100, Nike = 102, Jike }

B) public enum var1{ Mike = 100, Nike, Jike }

C) public enum var1{ Mike=-1 , Nike, Jike }

D) public enum var1{ Mike , Nike , Jike }

(8).接口是一种引用类型，在接口中可以声明（A），但不可以声明公有的域或私有的成员变量

A) 方法、属性、索引器和事件；

B) 方法、属性信息、属性；

C) 索引器和字段；

D) 事件和字段；

(9).在ADO.NET中，对于Command对象的ExecuteNonQuery()方法和ExecuteReader()方法，下面叙述错误的是（ C）。

A) insert、update、delete等操作的Sql语句主要用ExecuteNonQuery()方法来执行；

B) ExecuteNonQuery()方法返回执行Sql语句所影响的行数；

C) Select操作的Sql语句只能由ExecuteReader()方法来执行；

D) ExecuteReader()方法返回一个DataReder对象；

(10).下列ASP.NET语句（B ）正确地创建了一个与SQL Server 2000数据库的连接。

A) SqlConnection con1 = new Connection(“Data Source = localhost; Integrated Security = SSPI; Initial Catalog = myDB”);

B) SqlConnection con1 = new SqlConnection(“Data Source = localhost; Integrated Security = SSPI; Initial Catalog = myDB”);

C) SqlConnection con1 = new SqlConnection(Data Source = localhost; Integrated Security = SSPI; Initial Catalog = myDB);

D) SqlConnection con1 = new OleDbConnection(“Data Source = localhost; Integrated Security = SSPI; Initial Catalog = myDB”);

(11) int[][] myArray3=new int[3][]{new int[3]{5,6,2},new int[5]{6,9,7,8,3},new int[2]{3,2}}; myArray3[2][1]的值是（B）。 ？？？

A) 1.9

B) 2.2

C) 3.6

(12).在ADO.NET中执行一个存储过程时，如果要设置输出参数则必须同时设置参数的方向和（D ），必要时还要设置参数尺寸。

A) 大小；

B) 上限；

C) 初始值；

D) 类型；

(13).如果设treeView1=new TreeView()，则treeView1.Nodes.Add("根节点")返回的是一个 ( A )类型的值。

A) TreeNode；

B) int；

C) string；

D) TreeView；

(14).下面关于XML的描述错误的是（ c）。

A) XML提供一种描述结构化数据的方法；

B) XML 是一种简单、与平台无关并被广泛采用的标准；

C) XML文档可承载各种信息；

(15).“访问范围限定于此程序或那些由它所属的类派生的类型”是对以下哪个成员可访问性含义的正确描述？(d )

A) public

B) protected

C) internal

D) protected internal

(16).class Class1 {

private static int count = 0;

static Class1(){

count++;

}

public Class1(){

count++;

}

}

Class1 o1 = new Class1();

Class1 o2 = new Class1();

请问，o1.Count的值是多少？(c ) 题目有问题，对象怎么能访问静态变量

A) 1

B) 2

C) 3

D) 4

(17).以下关于ref和out的描述哪些项是正确的？(多选) ( ACD )

A) 使用ref参数，传递到ref参数的参数必须最先初始化。

B) 使用out参数，传递到out参数的参数必须最先初始化。

C) 使用ref参数，必须将参数作为ref参数显式传递到方法。

D) 使用out参数，必须将参数作为out参数显式传递到方法。

(18).您要创建一个ASP.NET应用程序在DataGrid控件中显示一个经过排序的列表。产品数据被存放于一个名为PubBase的Microsoft SQL Server 数据库。每个产品的主键是ProductID，Numeric型并且每个产品有一个字母描述字段，名为ProductName。您使用一个SqlDataAdapter对象和一个SqlCommand对象通过调用一个存储过程从数据库中获取产品数据。您将SqlCommand对象的CommandType属性设置为CommandType.StoredProcedure，并将它的CommandText属性设置为procProductList。您成功的获取了一个DataTable对象，其中是已经按ProductID降序排列的产品列表。您打算显示以相反的字母顺序排列的ProductName，请问该怎么做？ ( c )

A) 将SqlCommand对象的CommandType属性修改为CommandType.Text，将CommandText属性修改为”SELECT \* FROM procProductList ORDER BY ProductName DESC”。然后将这个DataTable对象绑定到DataGrid控件。

B) 创建一个基于这个DataTable对象的新的DataView并将这个DataView的Sort属性设置为“ProductName DESC”。然后将这个DataView对象绑定到DataGrid控件。

C) 将DataGrid控件的AllowSorting属性设置为True，并将DataGridColumn的SortExpression属性设置为 “ProductName DESC”.以显示ProductName。然后将这个DataTable对象绑定到DataGrid控件。

D) 将DataTable对象的DisplayExpression属性设置为 “ORDER BY ProductName DESC”.。然后将这个DataTable对象绑定到DataGrid控件。

(19)在ADO.NET中，对于Command对象的ExecuteNonQuery()方法和ExecuteReader()方法，下面叙述错误的是（ C）。

A) insert、update、delete等操作的Sql语句主要用ExecuteNonQuery()方法来执行；

B) ExecuteNonQuery()方法返回执行Sql语句所影响的行数；

C) Select操作的Sql语句只能由ExecuteReader()方法来执行；

D) ExecuteReader()方法返回一个DataReder对象；

(20)ASP.NET中，在Web窗体页上注册一个用户控件，指定该控件的名称为”Mike”，正确的注册指令为(D )

A) <%@Register TagPrefix = “Mike” TagName = “Space2” Src = “myX.ascx”%>

B) <%@Register TagPrefix = “Space2” TagName = “Mike” Src = “myX.ascx”%>

C) <%@Register TagPrefix = “SpaceX” TagName = “Space2” Src = “Mike”%>

D) 以上皆非

正确的是：<%@ Register Src="Mike.ascx" TagName="Mike" TagPrefix="uc1" %>

版权所有 ? 2001-2005 虹天电子商务有限公司

经营许可证：粤B2-20040293

1.C/S和B/S的区别，优缺点有哪些？

答：C/S是客户端/服务器端，C/S的程序通常也叫胖客户端，也就是一个程序的大部分功能，都在客户端实现，而服务器端只实现一小部分功能。通过这点不难看出，C/S的程序大部分在客户端实现，对于服务器端的压力相对小一些，服务器端可以节省一些。而且C/S的程序用窗口来做，个人认为开发效率上快一点。但C/S的程序一大弊端就是，必须要在客户端安装并部署程序才能运行，也就是用户必须得到客户端程序才可以运行。所以得考虑不同系统之间的安装与配置。

B/S是浏览器/服务器端，B/S的程序通常也叫瘦客户端，与C/S相反。B/S的程序大部分功能都要在服务器端实现，客户端只用来做辅助的控制功能。因为B/S的程序直接部署安装在服务器上。用户只需要有浏览器，并知道网址就可以使用程序。客户端不必做任何配置和安装，即可使用应用程序。而B/S也是有缺点的，首先B/S大部分功能都要在服务器端实现，对于服务器的要求就很高。服务器同时要应付用户访问与程序处理，所以服务器的配置必须能够应付。另外就是B/S的程序是浏览器中执行，所以B/S的程序操作客户端电脑的硬件，一般就得安装ActiveX等插件在客户端才可以操作。因此B/S的程序对于操作硬件，优势没有C/S的程序要好。

2.ADO.NET的常用对象

答：Connection，Command，DataAdapter，DataSet，DataReader，configuration

3.MVC指的是什么？

答：MVC将程序功能分成三部分考虑：视图(VIEW，即可见的用户界面部分)、模型（Model，程序的数据模型和业务逻辑部分），控制器（Controller，根据用户输入通知模型和视图进行相应更改）。

4.三层架构？

答：业务层，逻辑层，可视层

5.为什么要用ISPORTBACK？

答：页面第一次加载时需要绑定一些数据，如果每次加载都绑定的话，会给服务器很大压力 ISPORTBACK第一次加载时返回的是false，以后每次返回的都是true，这样可以设置 一些只需要在页面一次加载时就绑定好的数据。

6.用户自定义控件？

答：自定义控件，跟HtmlControl或WebControl相似，编译后可以添加引用到工具栏里

7.WEBSERVERS调用主要有哪两种方法？

答：SOAP（简单对象访问协议），WSDL（Web服务描述语言）

8.怎样在ASP.NET中改变DATAGRID的自动换行？

答：在他的RowStyle属性的Font属性的Wrap＝true

9.怎样布置一个ASP.NET的界面？

答：VS 2005和VS 2003都有发布机制。2003可以发布然后再复制部署。

VS2005基本上可以直接部署到对应位置。

10.谈谈你对ASP和ASP.NET的区别？

答：ASP.Net摆脱了以前ASP使用脚本语言来编程的缺点，理论上可以使用任何编程语言包括C++ , VB , JS等等，当然，最合适的编程语言还是MS为.Net Frmaework专门推出的C(读c sharp)，它可以看作是VC和Java的混合体吧，尽管MS自己讲C#内核中更多的象VC，但实际上我还是认为它和Java更象一些吧。首先它是面向对象的编程语言，而不是一种脚本，所以它具有面向对象编程语言的一切特性，比如封装性、继承性、多态性等等，这就解决了刚才谈到的ASP的那些弱点。封装性使得代码逻辑清晰，易于管理，并且应用到ASP.Net上就可以使业务逻辑和Html页面分离，这样无论页面原型如何改变，业务逻辑代码都不必做任何改动；继承性和多态性使得代码的可重用性大大提高，你可以通过继承已有的对象最大限度保护你以前的投资。并且C#和C++、Java一样提供了完善的调试/纠错体系。

1.用JS或VB脚本编写一个程序

要求：1.当输入为2个以上字符时，提示报错信息，跳转

2.输入时间yyyy-mm-dd

答：1) function fun()

{

var text = document.getElementById('TextBox1').value;

if(text.length == 2)

{

alert('输入有误');

window.location='login.aspx';

}

}

2) function ChkData(str)

{

var r = str.match(/^(\d{1,4})(-|\/)(\d{1,2})\2(\d{1,2})$/);

if(r==null)

{

alert('格式不正确');

return;

}

}

2.事务处理，存储过程混合使用，题目太大，但是考的就是这个方面的

答：在许多大型、关键的应用程序中，计算机每秒钟都在执行大量的任务。更为经常的不是这些任务本身，而是将这些任务结合在一起完成一个业务要求，称为事务。当某一个任务失败时，就恢复到指定的恢复点，这个叫回滚。存储过程是编译好的存储在数据库的操作

1. 什么情况下用到虚方法，他接口的区别？

答：需要在派生类里重写该方法时在基类中标记为虚方法，接口中包含方法、属性、索引器、事件。

1. ArrayList,Arra和string[]三着的区别？

答：Array 的容量是固定的，而 ArrayList 或 List 的容量可根据需要自动扩充。如果更改了 Capacity 属性的值，则可以自动进行内存重新分配和元素复制。

ArrayList 和 List 提供添加、插入或移除某一范围元素的方法。在 Array 中，您只能一次获取或设置一个元素的值。

使用 Synchronized 方法很容易创建 ArrayList 或 List 的同步版本。Array 将实现同步的任务留给了用户。

ArrayList 和 List 提供将只读和固定大小包装返回到集合的方法；而 Array 不提供。

另一方面，Array 提供了 ArrayList 和 List 所缺少的某些灵活性。例如：

可以设置 Array 的下限，但 ArrayList 或 List 的下限始终为零。

Array 可以具有多个维度，而 ArrayList 或 List 始终只是一维的。

特定类型（不包括 Object）的 Array 的性能优于 ArrayList，这是因为 ArrayList 的元素属于 Object 类型，所以在存储或检索值类型时通常发生装箱和取消装箱操作。不过，在不需要重新分配时（即最初的容量十分接近列表的最大容量），List 的性能与同类型的数组十分相近。

需要数组的大多数情况都可以改为使用 ArrayList 或 List；它们更容易使用，并且一般与相同类型的数组具有相近的性能。

Array 位于 System 命名空间中；ArrayList 位于 System.Collections 命名空间中；List 位于System.Collections.Generic 命名空间中。

string[]只是字符串数组

1. ADO.net常用的对象有哪些？DataSet和DataReader的区别

答：Connection,Command,DataAdapter,DataSet,DataReader,Configuration；DateSet是数据在内存中的缓存，dataset是保存数据的数据结构，断开模式，而DataReader不承担保存数据的责任，需要自己手动关闭连接数据库，它只负责从数据源读取数据到本地而已，是只读向前的，它不是数据结构，而是网络通讯组件的高层封装。

1. Web控件和HTML控件能调用客户段吗？若能，都用到什么技术。说Web控件的优点。

答：可以，query就是获取客户端信息的。如果是脚本的话，用Attributes添加脚本。方便，快捷

1. ASP.net的身份验证有那些？都用了什么原理？

答：

|  |  |
| --- | --- |
| **术语** | **定义** |
| [Windows 身份验证提供程序](ms-help://MS.MSDNQTR.v80.chs/MS.MSDN.v80/MS.VisualStudio.v80.chs/dv_aspnetcon/html/0346bc00-a4c6-491b-a103-ad2100f71606.htm) | 提供有关如何将 Windows 身份验证与 Microsoft Internet 信息服务 (IIS) 身份验证结合使用来确保 ASP.NET 应用程序安全的信息。 |
| [Forms 身份验证提供程序](ms-help://MS.MSDNQTR.v80.chs/MS.MSDN.v80/MS.VisualStudio.v80.chs/dv_aspnetcon/html/77e21ba2-bad1-4967-a8ec-74942dea7e47.htm) | 提供有关如何使用您自己的代码创建应用程序特定的登录窗体并执行身份验证的信息。使用 Forms 身份验证的一种简便方法是使用 ASP.NET 成员资格和 ASP.NET 登录控件，它们一起提供了一种只需少量或无需代码就可以收集、验证和管理用户凭据的方法。有关更多信息，请参见[使用成员资格管理用户](ms-help://MS.MSDNQTR.v80.chs/MS.MSDN.v80/MS.VisualStudio.v80.chs/dv_aspnetcon/html/824c3a24-f0af-427c-a652-0d2d1e9397cd.htm)和 [ASP.NET 登录控件概述](ms-help://MS.MSDNQTR.v80.chs/MS.MSDN.v80/MS.VisualStudio.v80.chs/dv_aspnetcon/html/ac032230-6469-4b03-b68d-03ef2643a24d.htm)。 |
| [Passport 身份验证提供程序](ms-help://MS.MSDNQTR.v80.chs/MS.MSDN.v80/MS.VisualStudio.v80.chs/dv_aspnetcon/html/18eb5bbf-5fb0-44a6-826b-46304847b6bc.htm) | 提供有关由 Microsoft 提供的集中身份验证服务的信息，该服务为成员站点提供单一登录和核心配置文件服务。 |

1. 在.net中页面传递参数除了用到Session,Application和cookie还有几种方式？

答：url,<input type=hidden>,数据库

1. 有张表student有字段id,name,sex,birth

(1)找出Name相同的学生（用一句SQL语句）

答：select \* from student

group by id,name,sex,birth

having count(name)>1

（2）用一句SQL语句把学生Sex为男的改为女，女的改为男

答：update test1

set sex =case sex when '男' then '女' else '男' end from student

第8题忘了

1.在什么情况下使用虚方法?虚方法和接口的差别?

答：需要在派生类里重写该方法时在基类中标记为虚方法，接口中包含方法、属性、索引器、事件。

2.ArraryList,Array string[]的差别是什么?

答：Array 的容量是固定的，而 ArrayList 或 List 的容量可根据需要自动扩充。如果更改了 Capacity 属性的值，则可以自动进行内存重新分配和元素复制。

ArrayList 和 List 提供添加、插入或移除某一范围元素的方法。在 Array 中，您只能一次获取或设置一个元素的值。

使用 Synchronized 方法很容易创建 ArrayList 或 List 的同步版本。Array 将实现同步的任务留给了用户。

ArrayList 和 List 提供将只读和固定大小包装返回到集合的方法；而 Array 不提供。

另一方面，Array 提供了 ArrayList 和 List 所缺少的某些灵活性。例如：

可以设置 Array 的下限，但 ArrayList 或 List 的下限始终为零。

Array 可以具有多个维度，而 ArrayList 或 List 始终只是一维的。

特定类型（不包括 Object）的 Array 的性能优于 ArrayList，这是因为 ArrayList 的元素属于 Object 类型，所以在存储或检索值类型时通常发生装箱和取消装箱操作。不过，在不需要重新分配时（即最初的容量十分接近列表的最大容量），List 的性能与同类型的数组十分相近。

需要数组的大多数情况都可以改为使用 ArrayList 或 List；它们更容易使用，并且一般与相同类型的数组具有相近的性能。

Array 位于 System 命名空间中；ArrayList 位于 System.Collections 命名空间中；List 位于System.Collections.Generic 命名空间中。

string[]只是字符串数组

3.Html ,Web的服务器端的控件能不能在客户端调用,如果能,怎样调用?

答：可以，query就是获取客户端信息的。如果是脚本的话，用Attributes添加脚本。方便，快捷

4. 身份验证有那几种方法?原理是什么?

5.DataReader和DataSet有什么区别?

答：dataset是保存数据的数据结构，断开模式，而DataReader不承担保存数据的责任，需要自己手动关闭连接数据库，它只负责从数据源读取数据到本地而已，是只读向前的，它不是数据结构，而是网络通讯组件的高层封装。

还有一些关于JavaScript方面的一些题目,我不是很清楚的我记不得

就这些了还有些就忘了

1. 类的三种特性。
2. 接头和抽象类的特征和区别。
3. 排序。
4. Dateset 和SQL datereder 区别以及各自用法。
5. 线程的含义，线程中的strat和把起放入进程池有什么区别。
6. 装箱，坼箱各自的含义。
7. web user control 和 cusrom control 区别和含义
8. cookie application session viovstate 各种页面传值方式的用法。
9. 如何给服务器端控件增加客户端脚本。

1.抽象类和接口有什么异同?10分

答：接口用于规范，抽象类用于共性。接口中只能声明方法，属性，事件，索引器。而抽象类中可以有方法的实现，也可以定义非静态的类变量。抽象类是类，所以只能被单继承，但是接口却可以一次实现多个。抽象类可以提供某些方法的部分实现，接口不可以.抽象类的实例是它的子类给出的。接口的实例是实现接口的类给出的。再抽象类中加入一个方法，那么它的子类就同时有了这个方法。而在接口中加入新的方法，那么实现它的类就要重新编写（这就是为什么说接口是一个类的规范了）。接口成员被定义为公共的，但抽象类的成员也可以是私有的、受保护的、内部的或受保护的内部成员（其中受保护的内部成员只能在应用程序的代码或派生类中访问）。此外接口不能包含字段、构造函数、析构函数、静态成员或常量。

2.结构和类有什么异同?10 分

答：类是引用类型，可以继承类、接口和被继承，有默认的构造函数，有析构函数，可以使用abstract和sealed，有protected修饰符，必须使用new初始化。

结构是值类型，只能继承接口，不能被继承，没有默认的构造函数，可以创建，没有析构函数，不可以用abstract和sealed，没有protected修饰符，可以不用new初始化。

如何选择使用结构还是类：

1. 堆栈的空间有限，对于大量的逻辑的对象，创建类要比创建结构好一些
2. 结构表示如点、矩形和颜色这样的轻量对象，例如，如果声明一个含有 1000 个点对象的数组，则将为引用每个对象分配附加的内存。在此情况下，结构的成本较低。
3. 在表现抽象和多级别的对象层次时，类是最好的选择
4. 大多数情况下该类型只是一些数据时，结构是最佳的选择

3.有根粗细不一的绳子有A,B两端,从A端烧到B端的时间是一个小时,从B端烧到A端也是一个小时 .但从中间烧到A端或是B端不是半个小时.

给你两跟绳子和一个打火机 怎么样得到45分钟?15分

答：两根绳子同时点，只不过，一根两头一起点，一根只点一个头，当两头点的烧玩时，花掉30分钟，再把另一根绳子的另一头也点燃，这样它烧玩花掉的时间就是15分钟，两个时间加起来就是45分钟。

4.有一个数据库中的两个表(具体的记不太清楚)

1).再建两个表 10分

2).通过建的两个表查询 10分

3).两表查询 10分

4).两表复杂查询 15分

还有两个10分的题目我真的记不起来了

1 你知道哪几种WEB服务器？ 你最熟悉哪种？

答：IIS,IIS。

2 微软的WEB应用服务器是什么？配置步骤（具体步骤）

答：通常将 IIS（Internet 信息服务）用作 Web 服务器，来运行 ASP.NET 应用程序。

启动 IIS 管理器

创建虚拟目录

配置虚拟目录

或：启动 IIS 管理器

创建本地网站

配置本地网站

3 如何优化WEB服务器？

4 你熟悉哪些数据库？ 你最熟悉哪个？

答：SQL

5 怎么还原一个数据库？ 简单操作步骤？

答：先创建一个同名数据库，右键点击他，在所有任务中选择还原数据库

6 查询一张表，姓名为张三，根据姓名张三 查询他的信息

答：select \* from 表 where name=’ 张三’

7 什么是网络协议？Windows有什么网络协议？

答：网络协议是网络上所有设备（网络服务器、计算机及交换机、路由器、防火墙等）之间通信规则的集合，它定义了通信时信息必须采用的格式和这些格式的意义。TCP/IP协议

8 在局域网中，某台电脑有问题不能上网，上网慢 这是什么原因，如何解决？

答：首先看网络连接是否是好的，网线有问题没，IP地址是否和其他人冲突，中毒，网速受限制，电脑坏了，网络提供商

9 你认为作为一个人力资源软件的技术支持人员 应该具备哪些素质跟能力？

答：两年以上管理软件（eHR、OA、进销存或ERP）相关经验； 熟悉计算机软硬件和网络知识，熟悉SQL SERVER数据库，掌握一般编程知识； 耐心细致、思维清晰、任劳任怨，具备良好的沟通能力和应变能力

10 自我评价跟职业规划？

客户服务工程师

电子邮箱： gzjdware@pub.guangzhou.gd.cn

发布日期： 2006-11-21 工作地点： 广州市

招聘人数： 2 工作年限： 二年以上

外语要求： 英语 薪水范围： 面议

学 历： 大专 接受简历语言： 中文或英文

职位描述：两年以上管理软件（eHR、OA、进销存或ERP）相关经验； 熟悉计算机软硬件和网络知识，熟悉SQL SERVER数据库，掌握一般编程知识； 耐心细致、思维清晰、任劳任怨，具备良好的沟通能力和应变能力，有培训经验者优先。

联系地址：广州天河工业园软件园区建中路66号佳都商务科技大厦5楼

电话：020-85542558 找张先生 就ok了！

选择题：

(1)string str1 = "",(2)string str1 = null,string (3)str1 = string.Length = 0;

上面三个式子哪个运行更快？（）

(2)

|  |  |
| --- | --- |
|  | A) 类不可以多重继承而接口可以； |
|  | B) 抽象类自身可以定义成员而接口不可以； |
|  | C) 抽象类和接口都不能被实例化； |
|  | D) 一个类可以有多个基类和多个基接口； |

问答题：

1. 将一组无序的数字按从大到小排列?(提示：用冒泡排序做)
2. 数据库查询的基本结构是怎样的？
3. 值类型和引用类型的区别？
4. 一个程序运行要经过5个步骤？是哪5个？分别说出其作用？ 2.如何理解委托？ 3. Override与重载有什么区别
5. asp.net页面在加载的时，有许多事件。我们能在不同时间在不同的事件写代码，试写出asp.net页面启动时发生的事件（按顺序）及事件的用法。（5个）
6. public void sort (int[] list) 冒泡排序。
7. 用递归算法写出treenode的各个节点。
8. 表users中的字段id,name.

要求用数据库脚本实现以下功能，以ip为升序排列，并分页，每页10行。

5. 接口和抽象类各自的含义用法及区别。

**广州御银科技股份有限公司试卷（C++）**

**1、请说明 strcoyp()函数和memcopy()函数的区别**

**2、内部静态整形变量，整形自动变量，整形存储器默认初始值各是什么？**

**3、使用内联函数应该注意什么？**

**4、什么是抽象类，什么时候用到？**

**5、C++中继承分几类，继承方式有哪几种？**

**6、什么是匈牙利表示法？**

**7、用MFC TRACE宏编写一个调试案子**

**8、VC++ 编译器 warning level 有几个级别？**

**9、写出VC++6。0系统下的默认快捷键（不少于十一个）**

**10、描述TCP/IP协议的层次结构**

**11、简述 Windows NT 下的多进程，多线程的概念**

**12、子进程可以继承父进程的资源有哪些？**

**13、子进程可以继承父进程资源的前提条件是什么？**

**14、Windows NT 下进程之间的通讯形式有哪些？**

**15、现有一个从串口读取单个ASCIII字符函数 getchar(),请编写一个从串口读取字符串的函数**

**16、简述一下瀑布模型下的软件开发的基本流程**

1. 数据库建表！增删改查
2. 什么是存储过程，有什么作用，为什么要用存储过程。
3. 值类型和引用类型的区别。
4. Page.ispostback的作用。
5. B/S结构中，除了appliction,session,cookies 还有什么方法传递数据。
6. 什么是静态变量。

.**net笔试题**  
姓名：                      日期：

1.      填空: (1)面向对象的语言具有\_\_\_\_\_\_\_\_性、\_\_\_\_\_\_\_\_\_性、\_\_\_\_\_\_\_\_性。

        (2)能用foreach遍历访问的对象需要实现 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_接口或声明\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_方法的类型。

        (3)列举ADO.net中的五个主要对象\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_、\_\_\_\_\_\_\_\_\_\_\_\_\_、\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_、\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_、\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_。

2. 不定项选择：

(1) 以下叙述正确的是：

      A. 接口中可以有虚方法。     B. 一个类可以实现多个接口。  
      C. 接口不能被实例化。       D. 接口中可以包含已实现的方法。

  (2) 从数据库读取记录，你可能用到的方法有：

      A. ExecuteNonQuery            B. ExecuteScalar

      C. Fill                        D. ExecuteReader

3. 简述 private、 protected、 public、 internal 修饰符的访问权限。

4. 写出一条Sql语句： 取出表A中第31到第40记录（SQLServer, 以自动增长的ID作为主键,  注意：ID可能不是连续的。）

5 .列举ASP.NET 页面之间传递值的几种方式。

6. 写出程序的输出结果

class Class1  {  
       private string str = "Class1.str";  
       private int i = 0;  
       static void StringConvert(string str)  {  
           str = "string being converted.";  
       }  
       static void StringConvert(Class1 c)  {  
           c.str = "string being converted.";  
       }  
       static void Add(int i)  {  
           i++;  
       }  
       static void AddWithRef(ref int i)  {  
           i++;  
       }  
       static void Main()  {  
           int i1 = 10;  
           int i2 = 20;  
           string str = "str";  
           Class1 c = new Class1();  
           Add(i1);  
           AddWithRef(ref i2);  
           Add(c.i);  
           StringConvert(str);           
           StringConvert(c);  
           Console.WriteLine(i1);  
           Console.WriteLine(i2);  
           Console.WriteLine(c.i);  
           Console.WriteLine(str);  
           Console.WriteLine(c.str);  
       }    
    }

7.写出程序的输出结果

public abstract class A    
{  
        public A()    
       {  
            Console.WriteLine('A');  
        }  
        public virtual void Fun()    
       {  
            Console.WriteLine("A.Fun()");  
        }  
}

public class B: A    
{  
        public B()   
       {  
            Console.WriteLine('B');  
        }

        public new void Fun()    
       {  
            Console.WriteLine("B.Fun()");  
        }

        public static void Main()    
       {  
           A a = new B();  
           a.Fun();  
        }  
}

8.      写出程序的输出结果：  
public class A    
{  
        public virtual void Fun1(int i)   
       {  
            Console.WriteLine(i);  
        }

        public void Fun2(A a)     
       {  
            a.Fun1(1);  
            Fun1(5);  
        }  
}

public class B : A    
{  
        public override void Fun1(int i)      
       {  
            base.Fun1 (i + 1);  
        }

        public static void Main()     
       {  
            B b = new B();  
            A a = new A();  
            a.Fun2(b);  
            b.Fun2(a);            
        }  
}

9. 一列数的规则如下: 1、1、2、3、5、8、13、21、34......

   求第30位数是多少， 用递归算法实现。(C#语言)

10.  程序设计: 猫大叫一声，所有的老鼠都开始逃跑，主人被惊醒。(C#语言)

要求：  1.要有联动性，老鼠和主人的行为是被动的。

2.考虑可扩展性，猫的叫声可能引起其他联动效应。

参考答案:  
1. (1)  继承性、封装性、多态性。（考基本概念）  
   (2)  IEnumerable 、 GetEnumerator (对foreach机制的理解，本来不想出这题的，凑分)  
   (3)  ... (送分题, 对ADO.net的了解)  
评分标准：一空1分，满分10分。

2. (1) B、C (考对接口的理解)  (2)  B、C、D (考查对ADO.net的熟练程度)  
  评分标准: 一题5分，不选或者错选均不得分。漏选得2分。满分10分。

3. . private :   私有成员, 在类的内部才可以访问。  
     protected : 保护成员，该类内部和继承类中可以访问。  
     public :    公共成员，完全公开，没有访问限制。  
     internal:   在同一命名空间内可以访问。  
评分标准：答对1题2分，2题5分，3题7分。全对10分。 (送分题)

4.  解1:  select top 10 \* from A where id not in (select top 30 id from A)  
    解2:  select top 10 \* from A where id > (select max(id) from (select top 30 id from A )as A)  
    评分标准： 写对即10分。(答案不唯一，datagrid 分页可能需要用到)  
   
5.  1.使用QueryString,  如....?id=1; response. Redirect()....  
    2.使用Session变量  
    3.使用Server.Transfer  
    ....等等  
   评分标准： 答对1点得3分， 两点7分， 3点10分。

6.  (考查值引用和对象引用)  
10  
21  
0  
str  
string being converted.  
评分标准：答对一点得2分，满分10分。

7.  A  
    B  
A.Fun()  
评分标准： 写出A.B 得5分，写出A.Fun()得5分，满分10分。  
(考查在继承类中构造函数, 以及new 方法, )

8.  2  
    5  
    1  
    6

评分标准： 答对一点得2分，两点得5分，3点得7分。全对得10分。

(一些人做这题，头都晕了.... ^\_^ )

9.

    public class MainClass  
    {  
        public static void Main()     
        {  
            Console.WriteLine(Foo(30));  
        }  
        public static int Foo(int i)  
        {  
            if (i <= 0)  
                return 0;  
            else if(i > 0 && i <= 2)  
                return 1;  
            else return Foo(i -1) + Foo(i - 2);  
        }  
    }  
评分标准： 写出return Foo(i -1) + Foo(i - 2); 得5分。

 写出if(i > 0 && i <= 2) return 1; 得5分。

           方法参数过多需要扣分（扣除分数 = 参数个数 － 1）

           不用递归算法扣5分

           (递归算法在树结构建立等方面比较常用)

10

 要点：1. 联动效果，运行代码只要执行Cat.Cryed()方法。2. 对老鼠和主人进行抽象  
评分标准: <1>.构造出Cat、Mouse、Master三个类，并能使程序运行(2分)  
            <2>从Mouse和Master中提取抽象（5分）  
            <3>联动效应，只要执行Cat.Cryed()就可以使老鼠逃跑，主人惊醒。(3分)

    public interface Observer  
    {  
        void Response();    //观察者的响应，如是老鼠见到猫的反映  
    }  
    public interface Subject  
    {  
        void AimAt(Observer obs);  //针对哪些观察者，这里指猫的要扑捉的对象---老鼠  
    }  
    public class Mouse : Observer  
    {  
        private string name;  
        public Mouse(string name, Subject subj)  
        {             
            this.name = name;  
            subj.AimAt(this);  
        }  
          
        public void Response()  
        {  
            Console.WriteLine(name + " attempt to escape!");  
        }  
    }  
    public class Master : Observer  
    {     
        public Master(Subject subj)  
        {             
            subj.AimAt(this);  
        }  
          
        public void Response()  
        {  
            Console.WriteLine("Host waken!");  
        }    
    }  
   
    public class Cat : Subject  
    {  
        private ArrayList observers;  
        public Cat()  
        {     
            this.observers = new ArrayList();  
        }  
        public void AimAt(Observer obs)  
        {  
            this.observers.Add(obs);  
        }  
        public void Cry()  
        {  
            Console.WriteLine("Cat cryed!");  
            foreach (Observer obs in this.observers)  
            {  
                obs.Response();  
            }  
        }  
    }  
    class MainClass  
    {         
        static void Main(string[] args)  
        {  
            Cat cat = new Cat();  
            Mouse mouse1 = new Mouse("mouse1", cat);  
            Mouse mouse2 = new Mouse("mouse2", cat);  
            Master master = new Master(cat);  
            cat.Cry();  
        }  
    }

//---------------------------------------------------------------------------------------------

设计方法二: 使用event -- delegate设计..  
    public delegate void SubEventHandler();  
    public abstract class Subject  
    {  
        public event SubEventHandler SubEvent;  
        protected void FireAway()  
        {  
            if (this.SubEvent != null)  
                this.SubEvent();  
        }     
    }  
    public class Cat : Subject  
    {    
        public void Cry()  
        {  
            Console.WriteLine("cat cryed.");  
            this.FireAway();  
        }  
    }  
    public abstract class Observer  
    {  
        public Observer(Subject sub)  
        {  
            sub.SubEvent += new SubEventHandler(Response);  
        }  
        public abstract void Response();      
    }  
    public class Mouse : Observer  
    {  
        private string name;  
        public Mouse(string name, Subject sub) : base(sub)  
        {     
            this.name = name;  
        }  
        public override void Response()  
        {  
            Console.WriteLine(name + " attempt to escape!");  
        }  
    }  
    public class Master : Observer  
    {  
        public Master(Subject sub) : base(sub){}  
        public override void Response()  
        {  
            Console.WriteLine("host waken");  
        }  
    }  
    class Class1  
    {  
        static void Main(string[] args)  
        {  
            Cat cat = new Cat();  
            Mouse mouse1 = new Mouse("mouse1", cat);  
            Mouse mouse2 = new Mouse("mouse2", cat);  
            Master master = new Master(cat);  
            cat.Cry();  
        }

    }

1. C#中 property 与 attribute的区别，他们各有什么用处，这种机制的好处在哪里？

2. 讲一讲你理解的web service,在dot net framework中，怎么很好的结合xml?（讲概念就行了）

3. C#, Java 和 c++的特点，有什么相同的地方，不同的地方，C#分别从c++和java中吸取了他们那些优点？

4. C#可否对内存进行直接的操作？(这可是个难点哦？要注意！）

5. 用Visual C++ 6.0编写的代码(unmanaged code)，如何在CLR下和其他dot net component结合？

6. 以前作过的一些项目？介绍一下自己的编程经验。(这是蒙混不过去的，基本一问就知道是真是假)

7. 你对公司有什么期望？你的薪水期望值？为什么学计算机？以前的工作经验（这非常重要）？

Q3： 维护数据库的完整性、一致性、你喜欢用触发器还是自写业务逻辑？为什么

Q4：ADO。NET相对于ADO等主要有什么改进？

Q5：ASP。NET与ASP相比，主要有哪些进步？

Q6：C#中的委托是什么？事件是不是一种委托？

Q7：描述一下C#中索引器的实现过程，是否只能根据数字进行索引？

Q8：C#中要使一个类支持FOREACH遍历，实现过程怎样？

Q10：写一个HTML页面，实现以下功能，左键点击页面时显示“您好”，右键点击时显示“禁止右键”。并在2分钟后自动关闭页面。

Q11：你对XMLHTTP、WEBSERVICE 了解吗？简单描述其特点、作用

Q12：接口和抽象类有什么区别？你选择使用接口和抽象类的依据是什么？

Q13：自定义控件和一般用户控件的异同？如果要用这两者之一，你会选择哪种？为什么

Q14：大概描述一下ASP。NET服务器控件的生命周期

Q15：UML

Q16：面向对象的概念，主要特点

Q17：类划分的依据。类粒度问题

Q18：事物管理问题

第一，谈谈final, finally, finalize的区别。

第二，Anonymous Inner Class (匿名内部类) 是否可以extends(继承)其它类，是否可以implements(实现)interface(接口)?

第三，Static Nested Class 和 Inner Class的不同，说得越多越好(面试题有的很笼统)。

第四，&和&&的区别。

第五，HashMap和Hashtable的区别。

第六，Collection 和 Collections的区别。

第七，什么时候用assert。

第八，GC是什么? 为什么要有GC?

第九，String s = new String("xyz");创建了几个String Object?

第十，Math.round(11.5)等於多少? Math.round(-11.5)等於多少?

第十一，short s1 = 1; s1 = s1 + 1;有什么错? short s1 = 1; s1 += 1;有什么错?

第十二，sleep() 和 wait() 有什么区别?

第十三，Java有没有goto?

第十四，数组有没有length()这个方法? String有没有length()这个方法?

第十五，Overload和Override的区别。Overloaded的方法是否可以改变返回值的类型?

第十六，Set里的元素是不能重复的，那么用什么方法来区分重复与否呢? 是用==还是equals()? 它们有何区别?

第十七，给我一个你最常见到的runtime exception。

第十八，error和exception有什么区别?

第十九，List, Set, Map是否继承自Collection接口?

第二十，abstract class和interface有什么区别?

第二十一，abstract的method是否可同时是static,是否可同时是native，是否可同时是synchronized?

第二十二，接口是否可继承接口? 抽象类是否可实现(implements)接口? 抽象类是否可继承实体类(concrete class)?

第二十三，启动一个线程是用run()还是start()?

第二十四，构造器Constructor是否可被override?

第二十五，是否可以继承String类?

第二十六，当一个线程进入一个对象的一个synchronized方法后，其它线程是否可进入此对象的其它方法?

第二十七，try {}里有一个return语句，那么紧跟在这个try后的finally {}里的code会不会被执行，什么时候被执行，在return前还是后?

第二十八，编程题: 用最有效率的方法算出2乘以8等於几?

第二十九，两个对象值相同(x.equals(y) == true)，但却可有不同的hash code，这句话对不对?

第三十，当一个对象被当作参数传递到一个方法后，此方法可改变这个对象的属性，并可返回变化后的结果，那么这里到底是值传递还是引用传递?

第三十一，swtich是否能作用在byte上，是否能作用在long上，是否能作用在String上?

第三十二，编程题: 写一个Singleton出来。

数据库方面：

1.存储过程和函数的区别

2.事务是什么？

3.游标的作用？如何知道游标已经到了最后？

4.触发器分为事前触发和事后触发，这两种触发有和区别。语句级触发和行级触发有何区别。

1。用C＃实现以下功能

   a 产生一个int数组，长度为100，并向其中随机插入1-100，并且不能重复。

   b 对上面生成的数组排序，需要支持升序、降序两种顺序

2。请说明在.net中常用的几种页面间传递参数的方法，并说出他们的优缺点。

3。请说明.net中的错误处理机制，并举例

4。请说出强名的含义

5。请列出c＃中几种循环的方法，并指出他们的不同

6。请指出.net中所有类型的基类

7。请指出GAC的含义

8。SQL SREVER中，向一个表中插入了新数据，如何快捷的得到自增量字段的当前值

您在什么情况下会用到虚方法？它与接口有什么不同？

Q：Override与重载有什么区别？

Q：值类型与引用类型有什么区别？

Q：怎样理解静态变量？

Q：向服务器发送请求有几种方式？

Q：DataReader与Dataset有什么区别？

Q：如果在一个B/S结构的系统中需要传递变量值，但是又不能使用Session、Cookie、Application，您有几种方法进行处理？

Q：用.net做B/S结构的系统，您是用几层结构来开发，每一层之间的关系以及为什么要这样分层？

Q：软件开发过程一般有几个阶段？每个阶段的作用？

Q：微软推出了一系列的Application Block，请举出您所知道的Application Block并说明其作用？

Q：请列举一些您用到过的设计模式以及在什么情况下使用该模式？

Q：您对WebService的体会？

Q：您对编程的兴趣如何？工作中遇到不懂的问题是怎样去解决的？您一般怎样去提高自己的编程水平？

Q：您离职的原因是什么？

Q：通过超链接怎样传递中文参数？

Q：请编程遍历页面上所有TextBox控件并给它赋值为string.Empty？

Q：请编程实现一个冒泡排序算法？

1、override与重载的区别

2、.net的错误处理机制是什么

3、C＃中接口和类的异同

4、DataReader和DataSet的异同

1.有哪几种方法可以实现一个类存取另外一个类的成员函数及属性,并请举列来加以说明和分析.

2.如果需记录类的实例个数,该如何实现,请写一个简单的类于以证明.

3.A类是B类的基类,并且都有自己的构造,析构函数,请举例证明B类从实例化到消亡过程中构造,析构函数的执行过程.

4.需要实现对一个字符串的处理,首先将该字符串首尾的空格去掉,如果字符串中间还有连续空格的话,仅保留一个空格,即允许字符串中间有多个空格,但连续的空格数不可超过一个.

一.填空题

1.c#中的三元运算符是\_\_?:\_\_\_?

2.当整数a赋值给一个object对象时，整数a将会被\_\_装箱(封装)\_\_\_?

3.类成员有\_\_\_\_\_种可访问形式？  this.;new Class().Method;

4.public static const int A=1;这段代码有错误么？是什么？  const不能用static修饰

5.float f=-123.567F;

  int i=(int)f;

i的值现在是\_\_\_\_\_?   123

6.利用operator声明且仅声明了“==”，有什么错误么?

7.委托声明的关键字是\_\_\_\_\_\_? delagete

8.用sealed修饰的类有什么特点？密封，不能继承

9.在Asp.net中所有的自定义用户控件都必须继承自\_\_\_\_\_\_\_\_?Control

10.在.Net中所有可序列化的类都被标记为\_\_\_\_\_?

11.在.Net托管代码中我们不用担心内存漏洞，这是因为有了\_\_\_\_\_\_?gc

12.下面的代码中有什么错误吗？\_\_\_\_\_\_\_

     using System;

     class A

     {

          public virtual void F(){

              Console.WriteLine("A.F");

           }

      }

      abstract class B:A

       {

           public abstract override void F(); // new public abstract void F();

       }

13.当类T只声明了私有实例构造函数时，则在T的程序文本外部，\_\_\_可以\_\_\_（可以 or 不可以）从T派生出新的类，不可以\_\_\_\_（可以 or 不可以）直接创建T的任何实例。

14.下面这段代码有错误么？

    switch (i){

    case():

        CaseZero();

        break;

    case 1:

        CaseOne();

        break;

    case 2:

        dufault;  //wrong

        CaseTwo();

        break;

   }

15.在.Net中，类System.Web.UI.Page 可以被继承么？可以

二.简答题

1.在c#中using和new这两个关键字有什么意义，请写出你所知道的意义？using 指令 和语句 new 创建实例 new 隐藏基类中方法

4.谈谈类和结构的区别？类是引用类型、结构是值类型

5.一个长度为10000的字符串，通过随机从a-z中抽取10000个字符组成。请用c＃语言编写主要程序来实现。

6.对于这样的一个枚举类型：

      enum Color:byte{

          Red,

          Green,

          Blue,

          Orange

       }

     string[] ss=Enum.GetNames(typeof(Color));

     byte[]   bb=Enum.GetValues(typeof(Color));

试写一段程序显示出枚举类型中定义的所有符号名称以及它们对应的数值。

7.您了解设计模式么？请列出您所知道的设计模式的名称。 //

8.请在SQL Server中设计表来保存一个树状结构的组织结构图（假设结构图中只有名称这一项内容需要保存），如果我想查询某一职位下的所有职位，用一个存储过程来实现，你有什么思路？

9.什么叫做SQL注入，如何防止？请举例说明。

10.下面这段代码输出什么？为什么？

     int i=5;

     int j=5;

     if (Object.ReferenceEquals(i,j))

          Console.WriteLine("Equal");

     else

          Console.WriteLine("Not Equal");

//不相等，因为比较的是对象

1.写一个实现对一段字符串翻转的方法，附加一些条件，如其中包括“，”、“.”，对其设计测试用 例 。

2.对一支纸杯设计测试用例（可以是广义的杯，不一定是某一支特定功能的杯）

开发语言概念题

3.什么是反射？

4.用Singleton如何写设计模式

5.C#中的垃圾回收机制是怎样的？

6.什么是Application Pool？

7.链表和数组的区别，各有什么优缺点.

8.Remoting在客户端服务器怎么实现 ?

9.什么是友元函数？

10.用标准C如何实现多态？

11.什么是虚函数？

12.什么是抽象函数？

13.什么是内存泄漏，怎样最简单的方法判断被存泄漏 ?

英语题

14.用英文介绍一下使用C#/C++做的项目，主要功能

15.如果要与美国开电话会议，会如何与美国的工程师沟通

16.如果老板认为你的技术落后，你会怎么回答

数据库知识题

17.使用什么工具来调用存储过程

18.SQL Server的两种索引是何形式？索引的作用？索引的优缺点？

19.触发器的作用

其它知识题及问题

20.什么是Web Service？

21.什么是XML？

22.Socket怎么实现？

**笔试部分**

**1．  有三张数据表，分别为老师表（老师编号、名字、年龄）、学生表（学生编号、学生名字、成绩）、老师学生关系表（老师编号、学生编号、班级），查询出年龄在30岁以上的老师所带学生在12岁以上的的学生成绩平均分，并显示这些老师的详细信息（写出SQL语句）**

**2．  使用compute后缀查询出指定列的数据之和（有多列需要统计和）**

**3．  在VB中，inputdialog（“today”，“tomorrow”，5，“The Day After Tomorrow”）中的每个值分别代表什么含义？**

**4．  在ASP.NET中控件分为哪两种？**

**5．  在C#中接口和抽象类的异同？(错误：类可以多重继承。)**

**6．  用VB.net或C#实现文件打开对话框并获取文件名及路径**

**7．  Socket的实现步骤（写实现步骤，不写代码）**

**8．  谈谈水晶报表的制作**

**9. 写一个带输出参数的存储过程。**

**10。关于枚举类型的定义。（不能为负数）**

**11..net中连接数据库的字符串（Sqlconnection(“server = localhost, uid =”” , pwd =”” ,database = “””)**

**上机题**

**使用MVC模式实现增、删、改、查，有两张数据表第一张是客户表（主表），第二张从表（货物表），一对多的关系，一个客户表对应多张货物表**

**C++/MFC试题**

**一．填空题(26分)**

1． WIN32平台下，sizeof(short) = \_\_\_\_，sizeof(int) = \_\_\_\_，sizeof(long) = \_\_\_\_。(3分)

2．请给出如下程序的结果(2分)

int a = 3;

int b = a << 3;

a = \_\_\_\_，b = \_\_\_\_。

3．请给出如下程序的结果(2分)

int aaa = 0x01;

htonl(aaa) = \_\_\_\_。

4．请给出如下程序的结果(2分)

#define MAX\_NUM 100+200

int nTemp = MAX\_NUM\*10;

则Temp = \_\_\_\_。

5．请给出如下程序的结果(3 分)

char szTemp[1000] = "";

int nLen1 = sizeof(szTemp);

int nLen2 = strlen(szTemp);

strcpy(szTemp, "abc");

int nLen3 = sizeof(szTemp);

int nLen4 = strlen(szTemp);

int nTemp[100];

int \*pTemp = nTemp;

int nLen5 = sizeof(pTemp);

char szResult[200] = "";

sprintf(szResult, "%d,%d,%d,%d,%02d.", nLen1, nLen2, nLen3, nLen4, nLen5);

则szResult = \_\_\_\_。

6．MFC中，大部分类是从哪个类继承而来（CCmdTarget、CObject、CWinApp、CWnd）？(2分)\_\_\_\_

7．内存是进程范围or线程范围；\_\_\_\_

CPU调度时，针对进程or线程；\_\_\_\_

函数调用堆栈，针对进程or线程。\_\_\_\_(3分)

8．调用函数bbb后，输出是什么(4分)

void ccc(int x)

{

char szTemp[10] = "";

x = 2;

sprintf(szTemp, "%d,", x);

afxDump << szTemp;

if(x = 3)

{

int x = 4;

sprintf(szTemp, "%d,", x);

afxDump << szTemp;

}

sprintf(szTemp, "%d,", x);

afxDump << szTemp;

}

void bbb()

{

char szTemp[10] = "";

int x = 7;

ccc(x);

sprintf(szTemp, "%d,", x);

afxDump << szTemp;

}

**二．改错题(总共15分,每题5分)。**

1．下面代码有何错误

void func1()

{

int \*pa = NULL;

func2(pa);

delete pa;

}

void func2(int \*pb)

{

pb = new int(5);

}

2．下面代码有何错误

void func2(int \*value)

{

\*value = 2;

}

void func1()

{

int \*p = 0;

func2(p);

}

3．

int func1(int& b)

{

return 0;

}

void func2()

{

int bbb = 3;

func1(&bbb);

func1(bbb);

}

func2中有何错误，func1的参数b的类型是什么。

**三．简答题(64分)**

1.请简述C、C++、VC、MFC在概念上的区别(4分)

2．请写一个函数重载的简单例子(4分)

3.用什么函数开启新进程、线程。(4分)

4.SendMessage和PostMessage有什么区别(4分)

5.WaitForSingleObject有何作用；m\_pThrd的类型是CWinThread\*时，WaitForSingleObject(m\_pThrd->m\_hThread, INFINITE);有何作用。(4分)

6. \_\_stdcall、\_\_cdecl、\_\_pascal在什么方面有所不同。(4分)

7．请把下述代码加上异常处理。(6分)

int MyWriteFile(CString strFileName, CString strText)

{

int nRet = 0;

CFile myFile;

myFile.Open(strFileName, CFile::modeWrite|CFile::shareExclusive|CFile::modeCreate, NULL);

int nLen = strText.GetLength();

myFile.Write((char\*)(LPCSTR)strText, nLen);

myFile.Close();

return nRet;

}

8．请解释“func”为何种类型，这种类型的作用什么，变量ttt 的值是多少？(6分)

typedef int (\*func)(int, int\*);

int xxx(int a, int \*p)

{

return a + \*p;

}

int dowork(func aaa, int bbb, int \*ccc)

{

return aaa(bbb, ccc);

}

int sss = 4;

int ttt = dowork(&xxx, 3, &sss);

9．请问下述代码中: int operator+(…）起什么作用？this是什么？ccc 的值最终为多少？(6分)

class Fruit

{

public:

Fruit()

{

weight = 2;

}

Fruit(int w)

{

weight = w;

}

int operator+(Fruit f)

{

return this->weight \* f.weight;

}

private:

int weight;

};

Fruit aaa;

Fruit bbb(4);

int ccc = aaa + bbb;

10.请解释下面代码采用了何种C++特性（C语言不具备），作用是什么？(6分)

template<typename T>

T sum(T a, T b)

{

return (a + b);

}

11．请解释aaa.h中下面代码的功能(5分)

#if !defined(AFX\_MYSUDU\_H\_\_9B952BEA\_A051\_4026\_B4E5\_0598A39D2DA4\_\_INCLUDED\_)

#define AFX\_MYSUDU\_H\_\_9B952BEA\_A051\_4026\_B4E5\_0598A39D2DA4\_\_INCLUDED\_

**... ...**

#endif

12．CMemoryState主要功能是什么(5分)

13．请阅读下述代码，写出程序执行的结果（6分）

#include <iostream>

using namespace std;

class CBase

{

public:

virtual void print()

{

cout<< "base" << endl;

}

void DoPrint()

{

print();

}

};

class CChild1: public CBase

{

public:

virtual void print()

{

cout<< "child1" << endl;

}

};

class CChild2: public CBase

{

public:

virtual void print()

{

cout<< "child2" << endl;

}

};

void DoPrint(CBase \*base)

{

base->DoPrint();

}

void main()

{

CBase\* base = new CBase();

CChild1\* child1 = new CChild1();

CChild2\* child2 = new CChild2();

DoPrint(child1);

DoPrint(child2);

DoPrint(base);

delete base;

base = child1;

base->print();

delete child1;

delete child2;

}

课程 IL001100

ORACLE数据库基础知识

ISSUE1.0



目录

[课程说明 1](#_Toc50975894)

[课程介绍 1](#_Toc50975895)

[课程目标 1](#_Toc50975896)

[相关资料 1](#_Toc50975897)

[第1章 ORACLE数据库概述 2](#_Toc50975898)

[1.1 产品概述 2](#_Toc50975899)

[1.1.1 产品简介 2](#_Toc50975900)

[1.1.2 基本概念 3](#_Toc50975901)

[1.2 ORACLE系统结构 4](#_Toc50975902)

[1.2.2 ORACLE物理结构 4](#_Toc50975903)

[1.2.3 系统全局区 6](#_Toc50975904)

[1.2.4 进程 8](#_Toc50975905)

[1.3 存储管理 9](#_Toc50975906)

[1.3.2 逻辑结构 10](#_Toc50975907)

[1.3.3 表（Table） 14](#_Toc50975908)

[1.3.4 视图（View） 18](#_Toc50975909)

[1.3.5 索引（Index） 18](#_Toc50975910)

[1.3.6 同义词（Synonym） 19](#_Toc50975911)

[1.3.7 序列（Sequence） 19](#_Toc50975912)

[1.3.8 数据库链（Database Link） 20](#_Toc50975913)

[第2章 管理ORACLE数据库 21](#_Toc50975914)

[2.1 基本概念 21](#_Toc50975915)

[2.1.1 数据字典 21](#_Toc50975916)

[2.1.2 事务管理 23](#_Toc50975917)

[2.1.3 数据库管理员(DBA) 24](#_Toc50975918)

[2.1.4 ORACLE的四种状态 25](#_Toc50975919)

[2.2 SQL\*Plus方式的ORACLE数据库启动和关闭 26](#_Toc50975920)

[2.2.1 启动数据库 26](#_Toc50975921)

[2.2.2 关闭数据库 26](#_Toc50975922)

[2.3 svrmgrl方式的ORACLE数据库启动和关闭 28](#_Toc50975923)

[2.3.1 启动数据库 28](#_Toc50975924)

[2.3.2 关闭数据库 30](#_Toc50975925)

[2.4 应用开发工具（SQL \* Plus） 31](#_Toc50975926)

[2.4.1 SQL 32](#_Toc50975927)

[2.4.2 PL/SQL 33](#_Toc50975928)

[2.4.3 数据库管理工具 36](#_Toc50975929)

[2.5 ORACLE用户及权限管理 36](#_Toc50975930)

[2.5.1 ORACLE的用户管理 37](#_Toc50975931)

[2.5.2 ORACLE的权限管理 38](#_Toc50975932)

[2.6 ORACLE数据库的备份与恢复 39](#_Toc50975933)

[2.6.1 Export 转入程序 40](#_Toc50975934)

[2.6.2 Import 转入程序 42](#_Toc50975935)

[2.6.3 增量卸出/装入 44](#_Toc50975936)

[第3章 ORACLE数据库的网络应用 46](#_Toc50975937)

[3.1 SQL\*Net产品介绍 46](#_Toc50975938)

[3.2 配置客户机/服务器结构 47](#_Toc50975939)

[3.2.1 配置listener.ora 47](#_Toc50975940)

[3.2.2 配置tnsnames.ora文件 48](#_Toc50975941)

[第4章 常用任务示例 51](#_Toc50975942)

[4.1 如何恢复被误删的数据文件 51](#_Toc50975943)

[4.2 如何杀掉吊死session 51](#_Toc50975944)

[4.3 如何修改字符集 51](#_Toc50975945)

[4.4 如何追加表空间 51](#_Toc50975946)

[4.5 如何加大表的maxextents值 52](#_Toc50975947)

[4.6 如何查询无效对象 52](#_Toc50975948)

[4.7 怎样分析SQL语句是否用到索引 52](#_Toc50975949)

[4.8 怎样判断是否存在回滚段竞争 53](#_Toc50975950)

[4.9 怎样手工跟踪函数/存储过程执行情况 54](#_Toc50975951)

[4.10 多种业务使用同一数据库如何分配回滚段 54](#_Toc50975952)

[4.11 怎样倒出、倒入文本数据 54](#_Toc50975953)

[4.11.1 倒出 54](#_Toc50975954)

[4.11.2 倒入 55](#_Toc50975955)

[4.12 如何更新当前数据库日志备份方式为archive 56](#_Toc50975956)

[4.13 Unix环境下如何实现自动备份 56](#_Toc50975957)

[4.13.1 设置运行环境 56](#_Toc50975958)

[4.13.2 倒出数据 56](#_Toc50975959)

[4.13.3 异地备份 57](#_Toc50975960)

[4.13.4 启动备份进程 58](#_Toc50975961)

[4.14 怎样分析ORACLE故障 59](#_Toc50975962)

[小结 61](#_Toc50975963)

[附录A ORACLE数据字典与视图 62](#_Toc50975964)

[附录B 动态性能表 68](#_Toc50975965)

[附录C SQL语言运算符与函数 70](#_Toc50975966)

# 课程说明

## 课程介绍

本课程对应的产品版本为：Oracle 8.1.7。

## 课程目标

完成本课程的学习后，您应该能够：

掌握ORACLE数据库系统结构中物理结构、系统全局区和进程的概念

掌握ORACLE数据库的存储管理知识

掌握管理ORACLE数据库的方法

了解ORACLE数据库的网络应用

## 相关资料

ORACLE公司发布的相关资料。

# ORACLE数据库概述

## 产品概述

### 产品简介

数据库技术产生于60年代末70年代初，到现在比较知名的大型数据库系统有ORACLE、Sybase、Informix、DB2（IBM公司的）、Ingress、RDB、SQL Server等。

ORACLE公司于1979年，首先推出基于SQL标准的关系数据库产品，可以在100多种硬件平台上运行（包括微机、工作站、小型机、中型机和大型机），支持多种操作系统。1986年，ORACLE推出具有分布式结构的版本5，可将数据和应用驻留在多台计算机上，而相互间的通信是透明的。1988年，推出版本6（V6.0）可带事务处理选项，提高了事务处理的速度。1992年推出了版本7，可带过程数据库选项、分布式数据库选项和并行服务器选项，称为ORACLE7数据库管理系统，它释放出了开放的关系型系统的真正潜力。目前，新版本为ORACLE9i，通用版本为ORACLE8i。版本Oracle 8i 是在企业内和因特网上开发、布署和管理应用高移动性和可伸缩性的数据库。Oracle 8i 最主要的新特性是在数据库中包括J a v a 和因特网能力。Oracle 8i 还包括能够管理和访问多媒体（视频、音频、图像、文本、空间）的Oracle interMedia 。Oracle 8i 还包括Oracle We b D B ，这个We b 开发环境使开发者能建立使用标准的We b 浏览器和Oracle 8i 数据库的动态的、数据驱动的We b 网站。ORACLE8i是一种通用叫法，它包括许多更细版本，如ORACLE8.1.5、ORACLE8.1.6、ORACLE8.1.7，华为公司TELLIN U-NICA产品使用版本ORACLE8.1.7序列。

🕮 说明：

“i”表示ORACLE数据库与internet的结合。

ORACLE用多达5个号码来完全识别一个版本，如“8.1.7.2.3”，命名方法如下：

|  |  |  |
| --- | --- | --- |
| 序号 | 值 | 含义 |
| 第一位 | 8 | 版本号（Version），是主要的标识符，表示软件的主要版本，包含显著的新功能，例如：版本8（也标识为版本8.0）。 |
| 第二位 | 1 | 新特征版本号（New Features Release Number），该位表示新特征版本级，例如：版本8.1。 |
| 第三位 | 7 | 维护版本号（Maintenance Release Number），该位表示一个维护版本级。也可能包含一些新特征，例如：版本8.0.4、8.1.7。 |
| 第四位 | 2 | 通用补丁级号（Generic Patch Set Number），该位表示一个通用补丁级。补丁级可以跨越所有操作系统和硬件平台使用，例如：补丁级版本8.1.7.2 |
| 第五位 | 3 | 专用平台补丁级（Platform Specific Patch Set Number），该位表示一个只能应用到特定的操作系统和硬件平台的补丁级，例如：补丁级8.1.7.2.3。 |

可以用如下的SQL语句得到ORACLE版本信息：

SQL> select \* from product\_component\_version;

### 基本概念

要了解ORACLE数据库，需要先了解以下基本概念：

数据库：是一个数据集合，我们大多数情况下讲的数据库概念不仅是指物理数据，还指内存、进程对象的组合。ORACLE数据库都将数据存储在文件中，在其内部，数据库结构提供了数据对文件的逻辑映射，允许不同类型的数据分开存放。这些逻辑划分称为表空间。关于表空间的概念将在下文中介绍。

表空间（Tablespace）：是数据库的逻辑划分，每个数据库至少有一个表空间（system表空间）。

数据文件（DataFile）：每个表空间由同一磁盘上的一个或多个文件组成，这些文件叫数据文件。建立新表空间需要建立新的数据文件。

实例（Instance）：也称为服务器（server），是存取和控制数据库的软件机制，它由系统全局区SGA和后台进程组成。

ORACLE数据库启动时，实际是启动ORACLE实例，（安装并打开数据库）。一个数据库可以被多个实例访问（这是ORACLE的并行服务器选项）。决定实例的大小及组成的参数存储在init.ora文件中，实例启动时需要读这个文件，并且在运行时可以由数据库管理员修改，但要在下次启动实例时才会起作用。

🕮 说明：

实例名字可以由环境变量ORACLE\_SID来定义。

实例的init.ora文件名通常包含该实例的名字，格式为initSID.ora：如一个实例的名字（SID）为ora817，则init.ora文件名通常为initora817.ora。

## ORACLE系统结构

任何硬件平台或操作系统下的ORACLE体系结构是相同的，包括如下三个方面：

1. 物理结构：数据文件，日志文件，控制文件，参数文件。
2. 系统全局区（SGA）：共享池，数据缓冲区，日志缓冲区，字典缓冲区。
3. 进程：用户进程，服务器进程，后台进程。

具体结构如图1-1所示。



ORACLE数据库体系结构

### ORACLE物理结构

一个ORACLE数据库是数据的集合，被处理成一个单位。一个ORACLE数据库有一个物理结构和一个逻辑结构。数据库物理结构是由构成数据库的操作系统文件所决定。每一个ORACLE数据库是由四种类型的文件组成：数据文件、日志文件、控制文件和参数文件。数据库的文件为数据库信息提供真正的物理存储。

ORACLE数据库的物理结构由四类文件组成

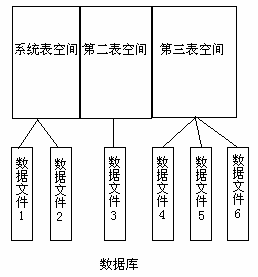
#### 数据文件

数据文件是物理存储ORACLE数据库数据的文件，每一个ORACLE数据库有一个或多个物理的数据文件(data file)。一个数据库的所有数据文件包含了全部数据库数据。逻辑数据库结构（如表、索引）的数据物理地存储在数据库的数据文件中。数据文件有下列特征：

每一个数据文件只与一个数据库联系

一个表空间可包含一个或多个数据文件

图1-2给出数据库、表空间及数据文件之间关系。



数据库、表空间及数据文件之间关系

#### 日志文件

每一个数据库有两个或多个日志文件(redo log file)的组，每一个日志文件组用于收集数据库日志。日志的主要功能是记录对数据所做的修改，所以对数据库做的全部修改是记录在日志中。日志文件的主要作用是保护数据库以防止故障。为了防止日志文件本身的故障，ORACLE允许镜象日志(mirrored redo log)，可以在不同磁盘上维护两个或多个日志副本。日志文件中的信息仅在系统故障或介质故障时用来恢复数据库。

#### 控制文件

每一个ORACLE数据库有一个控制文件(control file)，它记录数据库的物理结构，包含下列信息类型：

数据库名。

数据库数据文件和日志文件的名字和位置。

数据库建立日期。

为了安全起见，允许控制文件被镜象。每一个ORACLE数据库的实例启动时，它的控制文件用于标识数据库和日志文件，当着手数据库操作时它们被打开。当数据库的物理组成更改时，ORACLE自动更改该数据库的控制文件。数据恢复时，也要使用控制文件。

#### 参数文件

是一个文本文件，可直接使用文本编辑器对其内容进行修改，该文件只在建立数据库和启动实例时被访问，修改该文件后必须重启实例才生效。

系统初始参数文件：init .ora

生成的参数文件为：initSID.ora

参数文件作用如下：

设置SGA的大小。

设置数据库的全部缺省值。

设置数据库的范围。

在数据库建立时定义数据库的物理属性。

指定控制文件名和路径。

通过调整内存结构，优化数据库性能。

🕮 说明：

ORACLE8.1.7版本数据库实例参数文件“initSID.ora”是在安装ORACLE数据库过程中产生的，一般情况下不需要手工修改。

### 系统全局区

SGA（System Global Area）是ORACLE系统为实例分配的一组共享缓冲存储区，用于存放数据库数据和控制信息，以实现对数据库数据的管理和操作，图2-3给出实例与SGA及数据库关系。



实例与SGA及数据库关系

SGA分为以下几个部分：

1. 共享池。
2. 数据块缓冲区。
3. 日志缓冲区。
4. 字典缓冲区。

##### 下面分别介绍：

🕮 说明：

SGA的尺寸应小于物理内存的一半。

在ORACLE系统中，所有用户与ORACLE数据库系统的数据交换都要经过SGA区。

#### 共享池（Shared Pool）

由共享SQL区和数据字典区组成，参数shared\_pool\_size确定共享池大小，共享SQL区包括：

1. SQL或PL/SQL语句的文本。

SQL或PL/SQL语句的语法分析形式。

SQL或PL/SQL语句的执行方案。

🕮 说明：

shared\_pool\_size=物理内存×（15~20）%

对于OPS取下限。

#### 数据块缓冲区

用于存储从数据文件中读出的数据，其大小由以下两个参数决定：

DB\_BLOCK\_SIZE确定数据块的大小，一般为2K或4K。

DB\_BLOCK\_BUFFERS确定数据块的数目。

DB\_BUFFERS=DB\_BLOCK\_BUFFERS×DB\_BLOCK\_SIZE=物理内存×（15～25）%， 对于OPS取上限。该值应尽量大。

#### 日志缓冲区

以记录项的形式备份数据库缓冲区中被修改的缓冲块，这些记录项将被写到日志文件中。日志缓冲区大小由参数LOG\_BUFFER确定

#### 字典缓冲区

用于存放数据字典信息行。

#### SGA设置大小总原则

在设置各参数调整SGA大小时，应保证其小于物理内存的一半。

SGA的计算方法如下：

SGA=DB\_BLOCK\_BUFFERS×DB\_BLOCK\_SIZE+SHARE\_POOL\_SIZE+SORT\_AREA\_SIZE+1M+3×DB\_BLOCK\_SIZE×（CPUs+2）×PARALELL\_MAX\_SREVERS

🕮 说明：

OPS才有最后一项。

SORT\_AREA\_SIZE：排序区，并发用户数较多，需要排序时该值宜大。

PARALELL\_MAX\_SREVERS：2×CPU个数×4×并发用户数。对单CPU不能修改此值。

### 进程

进程是操作系统中的一种机制，它可执行一系列的操作步骤（在有些操作系统中使用作业(JOB)或任务(TASK)的术语来表示进程）。一个进程通常有它自己的专用存储区。

ORACLE实例有两种类型：单进程实例和多进程实例。通常用的是多进程实例。

单进程ORACLE（又称单用户ORACLE）是一种数据库系统，一个进程执行全部ORACLE代码。由于ORACLE部分和客户应用程序不能分别以进程执行，所以ORACLE的代码和用户的数据库应用是单个进程执行。在单进程环境下的ORACLE 实例，仅允许一个用户可存取。例如在MS-DOS上运行ORACLE 。

多进程ORACLE实例（又称多用户ORACLE）使用多个进程来执行ORACLE的不同部分，对于每一个连接的用户都有一个进程。在多进程系统中，进程分为两类：用户进程和ORACLE进程。当一用户运行一应用程序，如PRO\*C程序（Pro\*C是一种开发工具， 它把过程化语言C和非过程化语言SQL最完善地结合起来， 具有完备的过程处理能力，又能完成任何数据库的处理品任务，使用户可以通过编程完成各种类型的报表。）或一个ORACLE工具（如SQL\*PLUS），为用户运行的应用建立一个用户进程。ORACLE系统进程又分为服务器进程和后台进程。

#### 服务器进程

用于处理用户请求的进程。处理过程如下：

1. 分析SQL命令并生成执行方案。
2. 从数据缓冲区或磁盘中读取数据。
3. 将执行结果返回给用户。

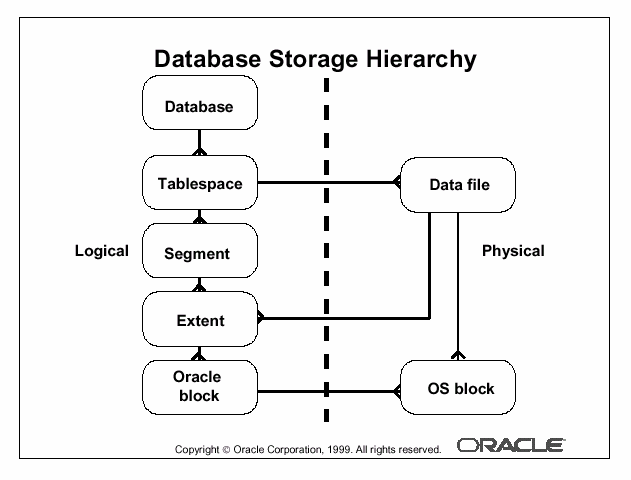
#### 后台进程

为所有数据库用户异步完成各种任务，主要包括如下进程（ora817表示实例名）：

|  |  |
| --- | --- |
| ora\_reco\_ora817 | 恢复进程：该进程是在具有分布式选项时所使用的一个进程，自动地解决在分布式事务中的故障。一个结点RECO后台进程自动地连接到包含有悬而未决的分布式事务的其它数据库中，RECO自动地解决所有的悬而不决的事务。任何相应于已处理的悬而不决的事务的行将从每一个数据库的悬挂事务表中删去。  当一数据库服务器的RECO后台进程试图建立同一远程服务器的通信，如果远程服务器是不可用或者网络连接不能建立时，RECO自动地在一个时间间隔之后再次连接。  RECO后台进程仅当在允许分布式事务的系统中出现，而且DISTRIBUTED – TRANSACTIONS参数是大于0。 |
| ora\_dbw0\_ora817 | 数据库写进程：该进程执行将缓冲区写入数据文件，是负责缓冲存储区管理的一个ORACLE后台进程。当缓冲区中的一缓冲区被修改，它被标志为“弄脏”，此进程的主要任务是将“弄脏”的缓冲区写入磁盘，使缓冲区保持“干净”。 |
| ora\_pmon\_ora817 | 进程监控进程：该进程在用户进程出现故障时执行进程恢复，负责清理内存储区和释放该进程所使用的资源。例：它要重置活动事务表的状态，释放封锁，将该故障的进程的ID从活动进程表中移去。PMON还周期地检查调度进程（DISPATCHER）和服务器进程的状态，如果已死，则重新启动（不包括有意删除的进程）。PMON有规律地被呼醒，检查是否需要，或者其它进程发现需要时可以被调用。 |
| ora\_lgwr\_ora817 | 日志写进程：该进程将日志缓冲区写入磁盘上的一个日志文件，它是负责管理日志缓冲区的一个ORACLE后台进程。 |
| ora\_ckpt\_ora817 | 检查点写进程：该进程在检查点出现时，对全部数据文件的标题进行修改，指示该检查点。 |
| ora\_smon\_ora817 | 系统监控进程：该进程实例启动时执行实例恢复，还负责清理不再使用的临时段。在具有并行服务器选项的环境下，SMON对有故障CPU或实例进行实例恢复。SMON进程有规律地被呼醒，检查是否需要，或者其它进程发现需要时可以被调用。 |

## 存储管理

前面已经提到过，ORACLE数据库都将数据存储在数据文件中，在其内部，数据库结构提供了数据对数据文件的逻辑映射，允许不同类型的数据分开存放。这些逻辑划分称为表空间，图1-4给出了详细的ORACLE数据库存储结构图。



ORACLE存储结构图

对于用户，通常关心的是ORACLE的逻辑结构，是在逻辑上是如何组成的，是如何存储数据的，下面详细介绍。

### 逻辑结构

ORACLE的逻辑结构是由一个或多个表空间组成，一个数据库划分为一个或多个逻辑单位，该逻辑单位称为表空间（tablespace）。一个表空间可将相关的逻辑结构组合在一起。

一个表空间由一组分类段组成。

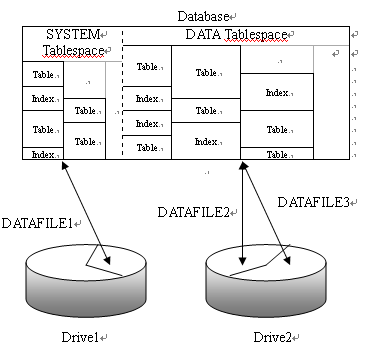
一个段由一组范围组成。

一个范围由一批数据库块组成。

一个数据库块对应一个或多个物理块。

每一个ORACLE数据库包含有一个名为SYSTEM的表空间，在数据库建立时自动建立。在该表空间中包含有整个数据库的系统信息数据，最小的数据库可只需要SYSTEM表空间。表空间利用增加数据文件可扩大表空间，表空间的大小为组成该表空间的数据文件的大小的和。ORACLE数据库中一表空间是由一个或多个物理数据文件组成，一个数据文件只可与一个表空间相联系。当为一表空间建立一数据文件时，ORACLE建立该文件，分配指定的磁盘空间容量。在数据文件初始建立后，所分配的磁盘不包含有任何数据。

一个模式（schema）为模式对象（schema object）的一个集合，每一个数据库用户对应一个模式。模式对象为直接引用数据库数据的逻辑结构，模式对象包含如表、视图、索引、同义词等结构。模式对象是逻辑数据存储结构，每一种模式对象在磁盘上没有一个相应文件存储其信息。一个模式对象逻辑地存储在数据库的一个表空间中，每一个对象的数据物理地包含在表空间的一个或多个数据文件中。例如：表、索引等模式对象，在指定表空间的数据文件上为该对象分配多少空间。图1-5说明模式对象、表空间和数据文件之间的关系。



模式对象、表空间和数据文件之间的关系

模式与表空间之间的关系为：一个表空间可包含不同模式的对象，而一个模式中的对象可包含在不同的表空间中。

#### 数据库块

数据库块（database block）是ORACLE逻辑分配空间的最底层，又称逻辑块、页或ORACLE块。

数据库块是数据库使用和分配空间的最小单元，也可以说是使用的最小I/O单元，一个数据块与磁盘上指定的物理空间大小相一致，一个数据库块对应一个或多个物理块，块的大小由参数db\_block\_size确定。

PCTFREE和PCTUSED是开发人员用来控制数据块中可用插入和更新数据的空闲空间大小的参数。

PCTFREE：设置数据块中保持空闲的百分比。

PCTUSED：当数据块空闲空间达到PCTFREE时，此块不允许插入数据，只能修改或删除块中的行，更新时可能使数据块空闲空间变大，已用数据空间变小，当已用空间低于PCTUSED时，则可以重新插入数据。

PCTFREE及PCTUSED的选择：

经常做查询（select）的表，应使PCTFREE小些，尽量减少存储空间浪费。

经常做插入（insert）的表，应使PCTUSED大一些。

经常做更新（update）的表，应使PCTFREE大一些，给更新留出更大的空间，减少行移动。

🕮 说明：

这两个参数只能在创建、修改表和聚簇（数据段）时指定。另外，在创建、修改索引（索引段）时只能指定PCTFREE参数。

**经常做插入的表设置举例：**

create table COMMINFOR

( SERIALNUMBER VARCHAR2(20) not null,

ENTERHISTIME DATE not null,

ACCEPTBEGINTIME DATE null ,

ACCEPTDURATION NUMBER(10) null ,

ACCEPTERNO VARCHAR2(4) null ,

CALLINGPHONENO VARCHAR2(20) null ,

ACCEPTID NUMBER(10) null ,

OPERATIONTYPE VARCHAR2(10) null ,

APPELLANTNAME VARCHAR2(16) null ,

CONTACTID NUMBER(10) null ,

)

pctfree 10

pctused 90

initrans 2

maxtrans 255

TABLESPACE SERVICE\_RPT\_DAT；

**经常做查询的表设置举例：**

create table AUTH

(

AUTHID NUMBER(10) not null,

DESCRIPTION VARCHAR2(50) null

constraint PK\_AUTH primary key (AUTHID)

)

pctfree 10

pctused 40

TABLESPACE SERVICE\_RPT\_DAT；

#### 范围（Extent）

数据库存储空间分配的逻辑单位，一个范围由一组数据块组成，范围是由段分配的，分配的第一个范围称初始范围，以后分配的范围称增量范围。

Extent是段中分配空间的逻辑单元。它有如下特性：

一个或多个范围构成一个段。

当段增长时，范围自动添加到段中。

DBA可以手工把范围加到一个段中。

一个范围不能跨数据文件，即一个扩展只属于一个数据文件。

一个范围由一片连续的ORACLE block构成。

每个段在定义时有许多存储参数来控制范围的分配，主要是STORGAE参数，主要包括如下几项：

INITIAL ：分配给段的第一个范围的字节数，缺省为5个数据块。

NEXT：分配给段的下一个增量范围的字节数，缺省为5个数据块。

MAXEXTENTS ：最大扩展次数。

PCTINCREASE：每一个增量范围都在最新分配的增量范围上增长，这个百分数缺省为50％，建表时通常设置为0，建表空间时为1%。

范围在分配时，遵循如下分配方式：

初始创建时，分配INITIAL指定大小的区。

空间不够时，按NEXT大小分配第二个区。

再不够时，按NEXT + NEXT \*PCTINCREASE分配。

可以对表、聚集、索引、回滚段、表空间等实体设置存储参数。

#### 段

是表空间中一个指定类型的逻辑存储结构，它由一个或多个Extent组成，段将占用并增长存储空间，但是一个段不能跨越表空间，ORACLE中有数据段、索引段、临时段、回滚段和自举段（包含打开数据库时要装入的数据字典表）。

当执行建表命令时，系统将自动创建一个段，若没有指定存储参数子句storage，则系统会用相应表空间的storage，当初始区用完后，系统会按next和pctincrease的值分配新的区。

在这里要讲一个比较重要的概念：回滚段。回滚段是当某事务修改一个数据块时，用以存放数据以前映像信息的数据段。回滚段中的信息用以保存读连续性，并进行事务回滚和事务恢复。例如，如果事务通过把一列的关键值从10改为20来修改数据块，则原值10要存放于回滚段中，而数据块将具有新值20。如果事务被回滚，则值10从回滚段拷回数据块。事务产生的重做记录保证在事务提交或回滚之前保持在回滚段中，而一个事务只能用一个回滚段存放其所有的重做记录，因此，如果回滚段大小配置不恰当，当ORACLE执行一个大的事务时，就会出现回滚段溢出的错误。所以设置回滚段大小是一个比较重要的问题，这取决于数据库应用的主要事务模式（稳定的平均事务速度、频繁大型事务、不频繁大型事务），并可通过一些测试来确定。

这里还有另外一个概念：临时段。用于以下SQL操作：

CREATE INDEX

带DISTINCT，ORDER BY，GROUP BY，UNION，INTERSECT和MINUS子句的SELECT语句

无索引的JION语句

某些相互关联的子查询

了解了逻辑结构后，下面介绍数据库中存储对象。

### 表（Table）

#### 表的组成

表是数据库的最基本的逻辑结构，一切数据都存放在表中，一个ORACLE数据库就是由若干个数据表组成。其它数据库对象都是为了用户很好地操作表中的数据。表是关系模型中反映实体与属性关系的二维表格，它由列和行组成，通过行与列的关系，表达出了实体与属性的关系，如图1-6所示。



表（Table）的组成

表的组成部分有：

列/域/字段：表的内容，也就是实体的各个属性，组成了表的各个列。

列名：列的名称。

长度：该列所能容纳的最大数据位数。

类型：该列存储的数据类型，常用数据类型如0.0.0 0. 所示。

关键字：该列能唯一表示一行内容，则称该列为关键字。

非空列：该列值是不能为空的。

常用数据类型

|  |  |  |
| --- | --- | --- |
| 数据类型 | 说明 | 串长度 |
| Char（长度） | 定长字符串 | 255 |
| Varchar（长度） | 变长字符串 | 2000 |
| Number（精度，比例） | 数字 |  |
| Date | 日期时间 |  |
| Long（长度） | 大型可变长字符串 | 21亿 |

行/记录：表中所有列组合在一起形成的一条信息，称之为一行或一条记录。

所属表：记录一定是归属于某一张表的。

行号：每条记录在数据库中的一个定位位置。在ORACLE数据库中，每张表有一系统提供的伪列来定位每条记录。

记录长度：该记录所有非空列的长度求和得出的字节数。

#### 建表命令

建表命令如下：

CREATE TABLE tablename

(column1 datatype [DEFAULT expression] [constraint],

column1 datatype [DEFAULT expression] [constraint],

……）

[STORAGE子句]

[其他子句]；

**建表举例：**

create table AUTOCALLLOG

( ID VARCHAR2(20) not null ,

CityId NUMBER(4) NULL

)

TABLESPACE SERVICE\_HIST\_DAT

PCTFREE 10

PCTUSED 90

STORAGE

( INITIAL 10M

NEXT 10M

MINEXTENTS 2

MAXEXTENTS 500

pctincrease 0

)；

#### 删除表中数据

删除表中数据有以下两种方式：

Delete ：删除数据时，数据库需要写日志，ORACLE数据库还需要占用回滚段，每次事务越大，对数据库的冲击越大，所以在删除或者更新数据时，一定要注意控制事务的大小。ORACLE数据库在删除或更新数据时，where条件中加rownum控制每次删除或更新的数量，如：

delete from t\_my\_table where ... and rownum < 2000；

Truncate：删除表中所有数据，并回收空间。

🕮 说明：

用Delete删除数据时，只删除数据，但不释放空间，并且产生碎片。

可以采用导出后truncate 表，然后再导入的方法解决碎片的问题。

#### 删除表

删除表命令为DROP，利用DROP命令删除表注意：

删除表中全部行。

删除表中全部索引。

如果它不是聚集的成分，系统将收回该表的所有存储空间。

如果该表为视图的基表，或在存储过程、函数或包中被引用，ORACLE将使这些对象无效但没有被删除，在表恢复后，这些对象仍可使用。

#### 建表约束

建表约束包括：

1. 非空完整性约束。
2. 唯一完整性约束。
3. 主码完整性约束。
4. check完整性约束。
5. 引用完整性约束。

例如：在下面语句中包括了除“唯一完整性约束”外的所有约束。

SQL>create table emp

(eno char(4) constraint e1 primary key ,

ename char(8) not null,

sal number(8,2) constraint ck1

check (sal between 5000 and 10000),

mgr char(4),

dno char(2) constraint d2

references dep(dno)

on delete cascade

) ；

引用完整性约束时需要注意以下几点：

1. 先建主表,后从表。
2. 从表引用的必须是主表的主码或定义了唯一性约束的列。
3. 从表的外来码值必须匹配与主表的引用码的值
4. 当删除主表记录时，从表指定了on delete cascade子句，连带删除从表记录，否则，若从表中有对应记录,则不能删除主表记录。

### 视图（View）

视图，数据中表的窗口，在表上的查询所形成的一个数据集体。通过视图，你将看到你所需要的信息，而排除其它不关心的内容。

将表中所需要的列和行选取出来传递给用户。

不是真正将数据重新复制一遍，不占用存储空间。

不仅在表的基础上建立，还可在视图的基础之上再建立视图。

建立视图主要基于：

保护数据安全，防止机密数据泄露。

简化数据查询方式，建立有效的查询。

保持数据独立性，保证程序不会随着数据的位置变化、名称变化而需要修改。

🕮 说明：

ORACLE数据库的视图请参考“*附录A ORACLE数据字典与视图*”。

### 索引（Index）

索引是与表相关的一种选择结构。索引可建立在一表的一列或多列上，一旦建立，由ORACLE自动维护和使用，对用户是完全透明的。索引是逻辑地和物理地独立于数据，它们的建立或删除对表没有影响，所有应用可继续处理。

关于索引需要注意以下几点：

1. 索引是一种数据库对象，并不改变表的逻辑结构，而是在物理存储结构上增加一些辅助信息，以提高查询速度。
2. 当表中记录增加或删除时，索引结构均要发生变化，因此，当有大量数据装入数据库时，应该先装入数据，后建立索引，以提高数据装入速度。
3. 只有当按指定的索引列的值查找或按索引列的顺序存取表时，才可利用索引提高性能。

🕮 说明：

当索引被删除后,原分配给索引的数据块将全部返回给索引所在的表空间.

建立索引可以：

* 加快查询速度。
* 确保唯一性特征。可以为表中某一列建一个唯一性索引，那么如果有人企图向表中插入这样一行记录，即它在这个有索引的列的数值与以前已有值重复，则这个操作就会失败。

### 同义词（Synonym）

同义词是表、视图或其它数据库实体的一个别名。可以用与存取表、视图等实体同样的方法来存取别名。

简单性（用户名.表名→同义词名）。

表的独立性（只需修改同义词的定义，避免修改大量应用程序）。

### 序列（Sequence）

序列是一个数据库对象，利用它可以生成唯一的整数，序列的值是由ORACLE程序自动生成。

创建序列：

SQL>create sequence zxk increment by 10

start with 10 maxvalue 1000 cycle ;

上例说明：

increment by 10 增量值为10。

start with 10 指定生成的第一个序列号为10。

maxvalue 1000 指定序列可生成的最大值。

cycle 序列上升到最大值1000后，继续由开始值10开始生成。

引用序列：

序列当前值：zxk.currval

序列下一个值： zxk.nextval

### 数据库链（Database Link）

数据库链是本地数据库中的一个对象，利用它可以存取远程数据库上的对象。

在远程表或视图之后附加@db链名，即可在SQL语句中引用远程表或视图。

有专用数据库链和公用数据库链之分，专用数据库链仅为建立者使用；公用数据库链必须用关键字public定义，公用数据库链可为全部用户使用。

创建数据库链：

对由连接串“icd\_main”指定的数据库上的用户icdmain，口令为icd，建立一名为icdmainold的数据库链。

SQL>create database link icdmainold

>connect to icdmain

>identified by icd

>using 'icd\_main' ;

下面给出使用数据库链进行查询、插入、修改、删除举例：

SQL>select \* from emp@icdmainold;

SQL>insert into emp@icdmainold

>values('0001','zxk',5000,'0010','00');

SQL>update emp@icdmainold

>set sal=sal+1000;

SQL>delete from emp@icdmainold

>where ename='aaaa';

# 管理ORACLE数据库

本章主要讲述管理ORACLE数据库的一些必备知识。

## 基本概念

### 数据字典

数据字典是ORACLE数据库的最重要的部分之一，是由一组只读的表及其视图所组成。这些表和视图是数据库被建立同时由数据库系统建立起来的，起着系统状态的目录表的作用。数据字典描述表、列、索引、用户、访问权以及数据库中的其它实体，当其中的一个实体被建立、修改或取消时，数据库将自动修改数据字典。因此，数据字典总是包含着数据库的当前描述。数据字典提供有关该数据库的信息，可提供的信息如下：

ORACLE用户的名字。

每一个用户所授的特权和角色。

模式对象的名字(表、视图、索引、同义词等)。

关于完整性约束的信息。

列的缺省值。

有关数据库中对象的空间分布信息及当前使用情况。

审计信息(如谁存取或修改各种对象)。

其它一般的数据库信息。

可用SQL存取数据字典，由于数据字典为只读，只允许查询。数据字典中全部基本表和用户可存取视图为ORACLE用户SYS所持有，所有对象包含在SYS模式中，安全管理员对该帐号要严格控制。

当ORACLE数据库系统启动后，数据字典总是可用，它驻留在SYSTEM表空间中。数据字典包含视图集，在许多情况下，每一视图集有三种视图包含有类似信息，彼此以前缀相区别，前缀为USER、ALL和DBA。

* 前缀为USER的视图，为用户视图，是在用户的模式内。
* 前缀为ALL的视图为扩展的用户视图（为用户可存取的视图）。
* 前缀为DBA的视图为DBA的视图（为全部用户可存取的视图）。

常用数据字典与视图

ORACLE数据库有三个可查看系统数据库字典的用户：system、internal、sys，其它用户只能查看授权表空间的数据库字典，以某一用户登录数据库后可查看一些数据字典，了解数据库的资源情况，对日常维护的故障定位能起很大的帮助，下面就一些常用的数据表（视图）进行简单的说明：

1. dba\_tablespaces（user\_tablespaces）

除三个系统用户以外的用户登录可查看user\_tablespaces表。

dba\_tablespaces（user\_tablespaces）表可查看本数据库的表空间，它存放各个表空间的属性，包括：表空间名、表空间宿主、以及存储参数（initial、next、maxextents、minextents、blocks等）等。

dba\_data\_files（user\_data\_files）

该表可查看数据库所有数据库数据文件，它存放了各个数据文件的属性，包括：表空间名、宿主、数据文件名、大小、可用空间以及ID号等。

dba\_segments（user\_segments）

该表可查看数据库所有段，它存放了各个段的属性，包括：段名、表空间名、宿主、大小、第一个区间的块号等。

dba\_rollback\_segs

该表可查看数据库所有回滚段，它存放了各个数据库回滚段的属性，包括：回滚段名、表空间名、宿主、存储参数（initial、next、maxextents、minextents等）、第一个区间的块号、占用数据库文件ID号。

dba\_extents

该表可查看数据库所有区间，它存放了各个区间的属性，包括：区间ID、使用段ID、所属对象（object）名、宿主、占用数据库文件ID（对应dba\_data\_files中的ID号）、块号等。

常见应用：查看各对象的区间是否已达到了最大区间数量（maxextents）。

select object\_name,sum(\*) from dba\_extents group by object\_name;

dba\_free\_spaces

可查看数据库表空间的使用情况，如查看各数据库表空间的可用空间：

select tablespace\_name,sum(bytes) from dba\_free\_spaces group by talbespace\_name;

以上为常用的数据字典，可在维护中灵活应用。在维护过程中，第一步应查看数据库管理系统的运行日志，日志文件的存放路径应版本不同而不同，一般：ORACLE workgroup版，日志文件存放路径为：$ORACLE\_HOME/rdbms/alert\_'SID'.log，OPS版，日志文件存放路径为：$ORACLE\_HOME/../../admin/bdump/alert\_'SID'.log。可用命令：tail logfilename查看，或用vi logfilename,按shift+g到文件尾进行查看。

🕮 说明：

附录A详细列出了ORACLE的数据字典与视图，供参考。

### 事务管理

#### 事务(transaction)

数据库是多用户的共享资源，在多个用户并行地存取数据时，应对数据做并发控制，以免存取不正确的数据，破坏数据库的一致性。

数据库的任何状态变化都要能反映客观世界的某种真实存在的合理状态，反映到数据库中，就是数据要满足一定的约束条件，这种约束条件就是一致性约束。

有时，为了完成复杂的修改动作，往往需要临时破坏数据库的状态一致性。因此，就有必要把这种对数据库复杂修改的一连串动作序列合并起来，这就是事务。

事务是数据库维护数据一致性的单位，它将数据库从一致性状态转换成新的一致性状态。

#### 事务提交

提交事务：即将在事务中由SQL语句所执行的改变永久化。数据库数据的更新操作提交以后，这些更新操作就不能再撤消。ORACLE的提交命令如下：

SQL>COMMIT；

#### 事务回退

事务回退的含义是：撤消未提交事务中的SQL语句所作的对数据修改。ORACLE允许撤消未提交的整个事务，也允许撤消部分(需设置保存点)。回退之后，数据库将恢复事务开始时的状态或保留点状态。回退命令如下：

SQL>ROLLBACK；

#### 保存点

保存点就是将一个事务划分成为若干更小的部分，以便在必要时，使当前事务只回退一部分，而其余工作得到保留。其格式为：

SAVEPOINT 保存点名；

ROLLBACK TO 保存点名；

### 数据库管理员(DBA)

任何一个系统都需要一个负责管理和维护的人员，负责管理和维护ORACLE数据库的人就是数据库管理员（DataBase Administrator，简称DBA）。

每个ORACLE数据库至少要有一名专职的数据库管理员，其职责如下：

保证数据的完整性和一致性。

提高执行速度，调整系统性能。

管理数据库存储，减少数据冗余。

安装、升级ORACLE Server 和应用工具。

定期对数据进行备份。

DBA为做好上述工作，需要了解以下知识：

ORACLE产品结构。

DBA工具。

数据库备份和恢复。

数据字典用途。

在ORACLE系统初始安装后，存在以下三个具有DBA特权的数据库用户：

internal用户。

sys用户。

system用户。

#### DBA的操作系统帐户

在进行许多数据库的管理任务时，必须能够执行操作系统命令，所以需要有个操作系统帐户，以便能完成操作系统的操作，此时要求此帐户具有比一般数据库用户更多的操作系统权限或存取特权。

数据库的启动（startup）和关闭（shutdown）是重要的管理任务，只有通过internal用户连接到ORACLE数据库，才能执行启动和关闭。对于不同的操作系统，必须完成下列要求之一，才能用internal登录ORACLE。

你的操作系统帐户有允许你作为internal登录ORACLE操作系统权限。

你有权限用internal登录ORACLE。

若数据库用internal登录ORACLE时需要口令，你必须知道口令。

#### DBA角色和用户

每个数据库总是自动创建一个预先定义的角色“DBA”，该角色包括所有数据库系统权限，具有很强的特权，只应授予数据库管理员。

在数据库创建时，自动创建了两个用户，并授予DBA角色，具体如下：

SYS：初始口令为“CHANGE\_ON\_INSTALL”，所有数据字典基表和视图被存储在SYS用户中。

SYSTEM：初始口令为“MANAGER”，附加的表和视图（一些ORACLE选择件使用）以及被ORACLE工具使用的表和视图存放在SYSTEM用户中。

安装完毕后，建议立即修改初始口令。

DBA用户通常工具如下：

Sql \* DBA

Sql \* Loader

EXP和IMP

### ORACLE的四种状态

ORACLE数据库任何时候可以处于四种状态之一：

SHUTDOWN：数据库关闭

NOMOUNT：例程启动

MOUNT：例程启动，控制文件打开

OPEN：例程启动，所有数据文件打开

数据库可以在几种状态之间转变：

NOMOUNT到MOUNT

ALTER DATABASE MOUNT

MOUNT到OPEN

ALTER DATABASE OPEN

ALTER SYSTEM ENABLE RESTRICTED SESSION

## SQL\*Plus方式的ORACLE数据库启动和关闭

🕮 说明：

对TELLIN U-NICA产品的维护建议用SQL\*Plus方式来启动和关闭ORACLE数据库。

### 启动数据库

数据库启动包括的步骤：准备启动实例、启动数据库选项。

#### 准备启动实例

用户在准备启动数据库实例前需要执行的预备步骤为以系统DBA来连接数据库：

$ sqlplus user\_name/user\_password

🕮 说明：

使用具有系统DBA角色的用户连接ORACLE数据库管理系统。如：sys用户、system用户、unica用户等。

#### 启动数据库选项

SQL>START database\_name;

这个命令启动实例装入并打开数据库。

🕮 说明：

如果不指定database\_name，如：“SQL> START;”系统将启动.bash\_profile中定义的缺省的数据库实例。

### 关闭数据库

关闭数据库的种类：

用IMMEDIATE选项关闭。

用NORMAL选项关闭。

用TRANSACTIONAL选项关闭。

用ABORT选项关闭。

关闭数据库及数据库实例要用系统DBA角色的用户来连接数据库，然后用SHUTDOWN命令来关闭数据库。

$ sqlplus user\_name/user\_password

SQL>

#### 用IMMEDIATE选项关闭

在下面情况下使用立即关闭数据库：

当很快就要关电源时。

数据库或其他应用程序功能不正常时。

立即关闭数据库通过下面的情况进行：

未授权的事务退回（如果长期有未授权的事务存在，这种关闭的方法可能不会完成得很快）。

ORACLE不等待用户当前的数据库连接断开，ORACLE不问原因地退回激活事务，并断开所有的连接用户。

数据库下次启动将不要求任何实例恢复程序。

命令如下：

SQL> SHUTDOWN IMMEDIATE;

#### 用NORMAL选项关闭

在下面情况下使用此方式：

在语句发生后不允许新的连接。

在数据库关闭前，ORACLE等待当前所有与数据库连接的用户断开连接。

数据库的下一次启动将不需要任何实例恢复程序。

命令如下：

SQL> SHUTDOWN NORMAL;

#### 带有TRANSACTIONAL选项关闭

当用户希望在允许活动事务首先完成时进行一个计划中的事务关闭时应用此种方式。

命令如下：

SQL> SHUTDOWN TRANSACTIONAL;

#### 带有ABORT选项关闭

在如下情况下执行：

数据库或它的应用程序不正常，并且其他形式的关闭不起作用。

用户需要立即关闭数据库时（例如用户知道一分钟内就要停电时）。

用户在启动数据库实例遇到问题时。

退出实例关闭数据库会产生如下的后果：

ORACLE处理的当前客户端SQL语句立即中断。

未递交的事务将不会退回。

ORACLE将不等待当前与数据库的连接断开，即ORACLE将断开所有的连接用户。

数据库的下一次启动将要求实例恢复程序。

如果正常与立即关闭选项都不能工作，要立即退出当前的数据库实例。命令如下：

SQL> SHUTDOWN ABORT;

## svrmgrl方式的ORACLE数据库启动和关闭

### 启动数据库

打开数据库的前提条件如下所示：

# su - ORACLE（取得ORACLE用户权限）

$ svrmgrl （进入ORACLE数据库服务器管理交互界面）

*ORACLE Server Manager Release 3.1.7.0.0 - Production*

*Copyright (c) 1997, 1999, ORACLE Corporation. All Rights Reserved.*

*ORACLE8i Enterprise Edition Release 8.1.7.0.0 - Production*

*JServer Release 8.1.7.0.0 - Production*

SVRMGR> connect internal（通过internal用户连接ORACLE数据库，如有口令需要输入口令，即connect internal/口令）

🕮 说明：

退出ORACLE数据库交互界面命令为：

SVRMGR> quit

#### 启动例程不装载数据库

SVRMGR> startup nomount

启动一个实例而没有安装（mount）数据库，这步操作一般在创建数据库和重新创建控制文件时才会用到，到这一步为止，数据库做的操作有：

* 读初始化参数文件initSID.ora。
* 创建SGA。
* 启动后台进程。
* 打开告警文件和跟踪文件。

#### 启动例程并装载数据库，但不打开。

SVRMGR> startup mount

为执行一些特定的维护操作，经常需要启动一个实例并安装（mount）数据库，但不打开数据库。例如，如下操作就需要启动数据库到这一步：

* 为数据文件改名。
* 增加、删除或改名重做日志文件。
* 启动或停止数据库到归档模式。
* 进行完整的数据库恢复。

mount一个数据库包括如下几步：

把数据库与已经启动（nomount）的实例联系起来。

定位并打开初始化参数CONTROL\_FILE指定的控制文件（获取实例锁）。

读控制文件获取数据文件和重做日志文件的名字和状态（此时，数据库并不检查这些数据文件和联机重做日志文件是否存在）。

#### 启动例程，装载并打开数据库

SVRMGR> startup open

数据库正常表明实例已经启动，并且数据库已经安装（mount）和打开，这时所有合法的数据库用户才可连到数据库，并执行典型的数据访问操作。打开一个数据库包括：

打开联机数据文件。

打开联机重做日志文件。

用此种方式打开数据库，则允许所有合法用户对数据库做联接并执行各种数据库存取操作。

这一步中，ORACLE检查所有的数据文件和联机重做日志文件是否可以被打开，并检查数据库的一致性。对于一些可以自动恢复的错误，后台进程SMON在数据库打开之前将执行恢复操作，对于那些不能自动恢复的错误，打开数据库将失败并报错。

#### 启动例程并限制对数据库访问

SVRMGR> startup restrict

用此种方式打开数据库，只允许管理员使用，而不允许一般用户访问，即只有具有create session权限和restricted session系统权限的用户才可以联接到数据库上（只有管理员才有此权限），利用该种方式启动，通常用于执行如下操作：

进行结构维护，例如重建索引。

进行数据库exp或imp。

进行数据装入（sql \* loader）。

#### 强制数据库启动

SVRMGR> startup force

在用Normal和Immediate选项无法成功关闭数据库，或启动时出错，通常用此种方式打开数据库。

### 关闭数据库

关闭一个数据库分以下三步：

1. Closing the database

ORACLE首先把高速缓冲区和重做日志缓冲区中的内容分别写入数据文件和联机日志文件，然后关闭所有联机数据文件和日志文件，这时控制文件仍处于打开状态。

Dismounting the database

从一个实例卸装数据库，之后ORACLE关闭控制文件，但实例依然存在。

Shutdown the Instance

ORACLE关闭ALERT文件和跟踪文件，释放SGA，终止后台进程，彻底关闭数据库。

ORACLE在关闭数据库时有三个常用选项：normal、immediate、abort。三个选项的关闭过程如下所示：

# su - ORACLE（取得ORACLE用户权限）

$ svrmgrl

SVRMGR> connect internal

#### 在正常情况下关闭数据库

SVRMGR> shutdown normal

用该种方式关闭数据库，关闭进程取消所有用户访问数据库，等待直至所有用户完成请求并与服务器脱离，清除缓冲区和重做日志文件并更新数据文件和联机重做日志文件，打开文件锁，完成正在进行的事务，更新文件头，关闭线程、打开数据库实例锁，使控制文件和数据文件同步。简言之，使用normal选项关闭数据库，卸装数据库，并完全关闭实例。该选项是关闭数据库时经常建议的选项。

#### 立即关闭数据库

SVRMGR> shutdown immediate

在特定条件下，关闭数据库时可能要选择immediate选项。例如，DBA可能决定在初始化文件中增加PROCESSES参数，如果这需要立即完成，则DBA使用immediate选项。如果使用该选项关闭数据库，则ORACLE正在处理的当前SQL语句立即被终止，任何未提交的事务被回滚，数据库被关闭。使用这一选项的唯一缺点是ORACLE不等待当前用户断开与数据库的连接，但数据库是连续的，且在下次启动时不需要恢复。

#### 异常中止例程

SVRMGR> shutdown abort

当紧急情况发生时，可以用abort选项关闭数据库。如当某个后台进程死掉后，可能导致无法用normal或immediate选项关闭数据库，要用到abort选项。当使用abort选项关闭数据库时，当前SQL语句立即停止，且未提交的事务不回滚，下次启动时要进行实例恢复。

## 应用开发工具（SQL \* Plus）

SQL \* Plus是ORACLE的交互查询工具，允许执行SQL语句和PL/SQL块。

SQL：是一种灵活高效的查询语言，其主要功能是对关系数据库中的数据进行操作和处理。

PL/SQL：是ORACLE对SQL规范的扩展，是一种块结构语言，包括一整套的数据类型、条件结构、循环结构和异常处理结构，PL/SQL可以执行SQL语句，SQL语句中也可以使用PL/SQL函数。

下面给出SQL \* Plus常用命令

|  |  |
| --- | --- |
| 进入SQL \* Plus命令 | $sqlplus username/pass@servername |
| 获取帮助命令 | help 命令字。例如：sql > help accept |
| 查看表结构 | desc tablename |
| 运行命令 | @ 例如： SQL>@icdmain.sql |
| 终止符 | / |

命令行编辑命令

|  |  |
| --- | --- |
| a text | text行尾增加 |
| c / old/new | 在当前行中将old改为new |
| c/text | 从当前行删除text |
| cl buff | 删除SQl缓冲区所有行 |
| Del | 删除当前行 |
| L \* | 显示当前行 |

### SQL

#### 基本查询命令

select colummn\_name from tablename

where

group by

having

order by

where字句可使用运算符：=、<>、！=、>=、<=、in、between、like（%\_）、is null 或is not null、not、or、and。

order by：asc（升序）或desc（降序），缺省升序。

group by：将查询结果分组，在包含group by 子句的查询中，select 子句的列表中的所有个体值（除聚组函数avg、count等外）必须是groub by子句中的表达式或常量。

having：如果需要对各个组返回的结果进行筛选，查询满足一定条件的分组值，可以使用having。例如：

select deptno,sum(sal)

from emp

group by deptno

having sum(sal)>9000;

#### 插入记录及数据更新

插入记录

insert into tablename(l1,l2)

values(v1,v2); 可以带&

insert into tablename（列表）

select 语句；

更新数据

update tablename

set 列名1=，列名2=

where

delete from tablename where

#### 事务控制

commit

set autocommit on

rollback

savepoint 保存点

### PL/SQL

是ORACLE对SQL规范的扩展，是一种块结构语言，包括一整套的数据类型、条件结构、循环结构和异常处理结构，PL/SQL可以执行SQL语句，SQL语句中也可以使用PL/SQL函数。

#### 主要优点

1. 模块化结构。
2. 定义标识符（常量、变量、游标等）。
3. 用过程化语言控制结构进行程序设计。
4. 错误处理。
5. 提高操作性能。

#### PL/SQL语句种类

PL/SQL语句种类

|  |  |
| --- | --- |
| DDL（数据定义语言） | create（创建）、alter（修改结构）、drop（删除）、truncate（截 断）、（其他：rename）。 |
| DML（数据操纵语言） | insert（增）、delete（删）、select（查询）、update（改）。 |
| DCL（数据控制语言） | grant（授权）、revoke（回收权限）、set role（角色）。 |
| 事务控制 | commit、rollback、savepoint（其他：lock table、set constraint(s)、set transaction）。 |
| 审计控制 | audit、noaudit。 |
| 系统控制 | alter system。 |
| 会话控制 | alter session。 |
| 其他语句 | comment（添加注释）、explain plan、analyze（收集统计）、validate、call。 |

#### PL/SQL块的基本结构

分三部分：

声明部分（declarative section）（可选）。

执行部分（executable section）（必须）。

异常处理部分（exception section）（可选）。

**例如：**

DECLARE

tmp\_dt date;

.....

BEGIN

select sysdate into tmp\_dt from dual;

…

EXCEPTION

…

END;

#### PL/SQL常见的类型转换函数

PL/SQL常见的类型转换函数

|  |  |  |
| --- | --- | --- |
| 函数 | 说明 | 转换可使用的类型 |
| TO\_CHAR | 按照可选的格式将参数转换为字符类型 | 数字型、日期型 |
| TO\_DATE | 按照可选的格式将参数转换为日期类型 | 字符型 |
| TO\_NUMBER | 按照可选的格式将参数转换为NUMBER类型 | 字符型 |

**例如：**

To\_char(sysdate,’yyyy-mm-dd hh24:mi:ss’)

To\_date(‘20020908','yyyymmdd')

To\_number(‘20020911’)

to\_number('123.4')

#### PL/SQL控制结构

分以下几类：

条件控制：IF-THEN-ELSE

循环：LOOP、FOR、WHILE

跳转控制：GOTO

1. IF-THEN-ELSE语法

IF boolean\_expression1 THEN

.....

[ ELSIF boolean\_expression2 THEN

..... ]

[ ELSE

.......]

END IF

LOOP

LOOP

……;

IF condition THEN

EXIT;

END IF;

END LOOP;

WHILE

WHILE condition LOOP

Sequence\_of\_statements;

END LOOP;

FOR

FOR counter IN lower\_bound..higher\_bound LOOP

Sequence\_of\_statements;

END LOOP;

GOTO

goto labelname;

<<labelname>>

🕮 说明：

可以是同一块中语句之间的跳转、可以从子块跳转到父块、不能从父块跳转到子块。

### 数据库管理工具

主要有以下几个工具：

SQL Worksheet：功能类似SQL Plus。

Instance Manager：可启动、关闭数据库，修改初始化参数，查看会话信息。

Schema Manager：可直接管理数据库对象，例如：表、存储过程、函数等。

Security Manager：对用户、角色进行管理。

Storage Manager：对表空间、数据文件、控制文件等进行管理。

## ORACLE用户及权限管理

合理的用户和权限管理对于数据库系统的高效、安全、可靠是很关键的。ORACLE在用户及权限管理上有许多新的概念和特性。

🕮 说明：

对用户及权限的管理需要进入SQL\*Plus交互工具。每一个SQL语句后要以分号“；”结束。退出交互工具命令为：quit

SQL命令语句及可选项不区分大小写，本文中出现大写的地方是强调作用。

### ORACLE的用户管理

每个ORACLE数据库都有许多合法用户，这些用户可以根据用户名和口令登录数据库，并使用SQL语言存取数据。

1. 创建用户

|  |  |
| --- | --- |
| **创建用户命令格式** | **说 明** |
| create user 用户名**;** | 创建用户的操作必须由DBA来做，一般用户无权创建用户。用户名必须是唯一的，即同一数据库中不能有两个相同的用户。 |
| identified by 口令**;** | 为用户设置口令 |
| default tablespace 表空间名**;** | 表示该用户存放数据的缺省表空间 |
| temporary tablespace 表空间名**;** | 表明用户使用的缺省临时表空间名 |
| quota 大小 on 表空间名**;** | quota 可以限制用户在某个表空间上最多可使用多少字节 |
| profile 资源文件**;** | profile 为用户指定各种资源的使用 |

下面是一个创建用户的完整例子：

|  |
| --- |
| create user scott； |
| identified by tiger； |
| default tablespace data\_ts； |
| temporary tablespace temp\_ts； |
| quota 500K on data\_ts； |
| profile newprofile； |

修改用户

对用户的修改包括：口令字、缺省表空间、临时表空间、表空间限量、profile、缺省角色。角色是ORACLE7的一个新概念，我们在“权限管理”里再讨论。在这里可把角色看成具有某些权限的一个特殊用户。修改用户的缺省角色也就是为用户指明另一个权限的集合。下面举例说明修改用户的操作：

|  |  |
| --- | --- |
| **任 务** | **命 令** |
| 将scott的口令改为hello | ALTER USER scott IDENTIFIED BY hello； |
| 将scott的缺省表空间改为data2\_ts | ALTER USER scott DEFAULT TABLESPACE data2\_ts； |
| 将scott的临时表空间修改为temp2\_ts | ALTER USER scott TEMPORARY TABLESPACE temp2\_ts； |
| 将scott的资源文件改为otherprofile | ALTER USER scott PROFILE otherprofile； |
| 将scott的缺省角色改为DEVELOPER | ALTER USER scott DEFAULT ROLE DEVELOPER； |
| 将当前系统所有角色都授予scott，除Payroll外 | ALTER USER scott DEFAULT ROLE ALL EXCEPT Payroll； |

删除用户

删除用户的命令为：

DROP USER 用户名 [CASCADE]；

若不使用CASCADE选项，则必须在该用户的所有实体都删除之后，才能删除该用户。使用CASCADE后，则不论用户实体有多大，都一并删除。

### ORACLE的权限管理

ORACLE的安全机制，是由系统权限、实体权限和角色权限这三级体系结构组成的。

|  |  |
| --- | --- |
| **权限类型** | **说 明** |
| 系统权限 | 是指对数据库系统及数据结构的操作权，例如创建/删除用户、表、同义词、索引等等 |
| 实体权限 | 是指用户对数据的操作权，如查询、更新、插入、删除、完整性约束等等 |
| 角色权限 | 是把几个相关的权限组成角色，角色之间可以进一步组合而成为一棵层次树，以对应于现实世界中的行政职位。角色权限除了限制操作权、控制权外，还能限制执行某些应用程序的权限。 |

这样的安全控制体系，使得整个系统的管理人员及程序开发人员能控制系统命令的运行、数据的操作及应用程序的执行。

1. 系统权限

系统权限的授予命令为GRANT，例如把创建任何表视图的权限授予scott用户：

GRANT create any view TO scott；

系统权限的回收命令为REVOKE，例如将create any view 权限从scott用户手中收回：

REVOKE create any view FROM scott；

实体权限

每种类型的实体有与之相关的实体权限。

授予实体权限的命令举例(将EMP表上的Select和Insert权限授给scott)：

GRANT select，insert ON emp TO scott；

回收实体权限的命令举例(将EMP表上的Select权限从scott手中回收)：

REVOKE select ON emp FROM scott；

管理角色

角色是许多权限和角色的组合，它极大地方便了ORACLE的权限管理。

创建角色，如创建一个名为dept1的角色，口令字为hello：

CREATE ROLE dept1 IDENTIFIED BY hello；

使用角色，可以通过修改用户的缺省角色来使用角色，或通过授权的方法来将角色授予其它角色或用户。如将scott用户的缺省角色修改为DEVELOPER：

ALTER USER scott DEFAULT ROLE DEVELOPER；

将角色dept1授予manager角色：

GRANT manager TO scott；

使角色生效或失效，DBA可以通过控制角色的生效或失效，来暂时回收用户的一部分权限。如使dept1角色失效：

SET ROLE dept1 DISABLE；

删除角色，这将会影响到拥有该角色的用户和其它角色的权限。用DROP ROLE命令删除角色，如：

DROP ROLE dept1；

## ORACLE数据库的备份与恢复

ORACLE系统提供的Export/转入（备份）、Import/转出（恢复）使用程序实现备份与恢复功能。

Export是在数据库打开并能使用的情况下备份数据库数据的实用程序。用Export将数据库中的数据写到以二进制形式表示的操作系统文件中（ORACLE），该文件叫卸出文件，它可以是磁盘文件，也可以是磁带文件。这些文件独立于数据库存在。卸出文件在需要时能被再装入到ORACLE数据库中，也可装入到另一个CPU上的ORACLE数据库上，还可把它装入到不同ORACLE版本的数据库上。用Export可实现应用程序失败时的恢复，例如可把某个表或某些表恢复到执行该Export时的状态。

由于卸出文件的特殊格式，所以只能用Import实用程序将其读入数据库中。

### Export 转入程序

ORACLE数据库有两类备份方法，第一类为物理备份，该方法实现数据库的完整恢复，但数据库必须运行在归档模式下，且需要极大的外部存储设备，例如磁带机；第二类备份方式为逻辑备份，客户服务中心业务数据库就是采用这种方式，这种方法不需要数据库运行在归档模式下，不但备份简单，而且可以不需要外部存储设备。

逻辑备份又分为三种模式。

表模式（T）：这种模式可以卸出当前用户数据库模式下的表，甚至是所有的表。具有特权的用户可根据所指定的数据库模式来（限制表）卸出他们所包含的表。缺省情况是卸出属于当前正在进行卸出的用户的所有表。

用户模式（U）：这种模式可以卸出当前用户数据库模式下的所有实体（表、数据和索引）。

全数据库模式（F）：只有具有EXP\_FULL\_DATABASE角色的用户才可能以这种模式卸出。以这种模式进行卸出的用户，除SYS模式下的内容之外，数据库中所有实体都可以卸出。 下面列出给用户赋予EXP\_FULL\_DATABASE角色的方法。

|  |  |
| --- | --- |
| # su – ORACLE | 假设当前为超级用户身份 |
| $ sqlplus sys/change\_on\_install | 以sys用户登录并进入Sql\*Plus |
| SQL> GRANT "EXP\_FULL\_DATABASE" TO 用户名 | 在Sql\*Plus下执行此命令 |

要选择表、用户或全数据库方式，可相应指定TABLES＝tablelist、OWNER＝userlist或FULL＝y。

#### 表模式

EXP ICDMAIN/ICD BUFFER=8192(或64000）

FILE=EXP\_ICDMAIN\_SERVICEINFO.DMP 或（磁带设备/dev/rmt0）

TABLES=ICDMAIN.SERVICEINFO（或ICDMAIN.COMMONINFORMATION，ICDMAIN.DEALINFO .....)

ROWS=Y COMPRESS=N

LOG= EXP\_ICDMAIN\_SERVICEINFO.LOG

参数说明：

|  |  |
| --- | --- |
| BUFFER | 缓冲区大小 |
| FILE | 由Export创建的输出文件的名字 |
| TABLES | 将要卸出的表名列表 |
| ROWS | 指明是否卸出表中数据的行数，缺省为“Y”。 |
| COMPRESS | 指明在装入期间是否将表中数据压缩到一个区域中。如果在卸出数据时，指定参数COMPRESS=Y，那么装入时，就会将数据压缩到一个初始区域中。这种选择可以保持初始化区域的原始大小。缺省为“Y”。 |
| LOG | 指定一个接收有用信息和错误信息的文件 |

#### 用户模式

EXP ICDMAIN/ICD OWNER=ICDMAIN BUFFER=8192（或64000）

FILE=EXP\_ICDMAINDB.DMP 或（磁带设备/dev/rmt0）

ROWS=Y

COMPRESS=N

LOG= EXP\_ICDMAINDB.LOG

参数说明：

|  |  |
| --- | --- |
| OWNER | 将要卸出的用户名列表 |
| BUFFER、FILE、ROWS、COMPRESS、LOG | 同上 |

#### 全数据库模式

EXP ICDMAIN/ICD BUFFER=8192(或64000)

FILE=EXP\_ICDMAIN\_DB.DMP （或磁带设备/dev/rmt0）

FULL=Y ROWS=Y COMPRESS=N

LOG= EXP\_ICDMAIN\_DB.LOG

对于数据库备份，建议采用增量备份，即只备份上一次备份以来更改的数据。

增量备份命令：

EXP ICDMAIN/ICD BUFFER=8192（或64000）

FILE=EXP\_ICDMAIN\_DB.DMP （或磁带设备/dev/rmt0）

FULL=Y INCTYPE=INCREMENTAL ROWS=Y COMPRESS=N

LOG=EXP\_ICDMAIN\_DB.LOG

参数说明：

|  |  |
| --- | --- |
| BUFFER、FILE、ROWS、COMPRESS、LOG | 同上 |
| FULL | 指明是否卸出完整的数据库。如果FULL=Y，将以全数据库模式进行卸出。 |
| INCTYPE | 增加卸出的类型，有效值有complete（完全）、comulative（固定）和incremental（增量）。 |
| complete | 输出所有表 |
| comulative | 将输入第一次完全输出后修改过的表 |
| incremental | 将输出前一次输出后修改过的表 |

 说明：

关于增量备份必须满足下列条件：

只对数据库备份有效，且第一次需要FULL=Y参数，以后需要INCTYPE=INCREMENTAL参数。

用户必须有EXP\_FULL\_DATABASE。

话务量较小时方可采用数据库备份。

使用Export备份数据时，可以使用联机帮助命令取得帮助信息，命令如下：

exp help=y

如果磁盘有空间，建议备份到磁盘，然后再备份到磁带。

### Import 转入程序

Import和Export是两个相配套的实用程序，Export把数据库中的数据卸出到操作系统文件中，而Import实用程序则把Export卸出的数据恢复到数据库中。

按备份方案确定恢复方案，例如：采用表逻辑备份方案，则恢复方案也采用恢复到表的方式（不应恢复到用户）。

要使用Import，必须具有CREATE SESSION特权，以便能注册到ORACLE RDBMS中去。这一特权属于在数据库创建时所建立的CONNECT角色。

如果卸出文件是由某用户利用EXP\_FULL\_DATABASE角色创建的全数据库卸出，那么只有具有IMP\_FULL\_DATABASE角色的用户才能装入这样的文件。下面给用户赋予IMP\_FULL\_DATABASE角色的方法。

|  |  |
| --- | --- |
| # su – ORACLE | 假设当前为超级用户身份 |
| $ sqlplus sys/change\_on\_install | 以sys用户登录并进入Sql\*Plus |
| SQL> GRANT "IMP\_FULL\_DATABASE" TO 用户名 | 在Sql\*Plus下执行此命令 |

数据库的逻辑恢复分为表、用户、数据库三种模式。

#### 表模式

恢复方法为：

IMP ICDMAIN/ICD FILE=文件名 LOG=LOG文件名

ROWS=Y COMMIT=Y BUFFER=Y IGNORE=Y

TABLES=（表名1，表名2，表名3，表名4，.......）

参数说明：

|  |  |
| --- | --- |
| BUFFER | 缓冲区大小 |
| FILE | 用于装入的卸出文件名字 |
| TABLES | 将要装入的表名列表 |
| ROWS | 指明是否装入表数据的行数，缺省为“Y”。 |
| IGNORE | 指明如何处理实体创建错误。指定IGNORE=Y，当试图创建数据库实体时，忽略实体存在错误。对除了表之外的其他实体，指定IGNORE=Y，Import不报告错误，继续执行。而指定IGNORE=N时，Import在继续执行前报告实体创建错误。 |
| COMMIT | 指明在每个矩阵插入之后是否提交。缺省时，Import在装入每个实体之后提交。指定COMMIT=N时，如有错误产生，Import在记录装入下一个实体之前，完成一个回退。指定COMMIT=Y时，可以抑制回滚字段无限制增大，并改善大量装入时的性能，表具有唯一约束时，这种选择比较好。如果再次开始装入，将拒绝装入已经装入的任何行，原因是非致命性错误。表具有非唯一约束时，指定COMMIT=N可能是比较好的选择。因为重新装入可能会产生重复行。 |
| LOG | 指定一个接收有用信息和错误信息的文件 |

#### 用户模式

如果备份方式为用户模式，采用下列恢复方法：

IMP SYSTEM/MANAGER FROMUSER=ICDMIAN TOUSER=ICDMAIN

FILE=文件名 LOG=LOG文件名 ROWS=Y COMMIT=Y

BUFFER=Y IGNORE=Y

参数说明同上。

#### 数据库模式

如果备份方式为数据库模式，采用下列恢复方法：

IMP SYSTEM/MANAGER FULL=Y

FILE=文件名 LOG=LOG文件名 ROWS=Y COMMIT=Y

BUFFER=Y IGNORE=Y

#### 字符集转换

对于单字节字符集（例如US7ASCII），恢复时，数据库自动转换为该会话的字符集（NLA\_LANG参数）；对于多字节字符集（例如ZHS168CGB），恢复时，应尽量使字符集相同（避免转换），如果要转换，目标数据库的字符集应是输出数据库字符集的超集。

### 增量卸出/装入

下面介绍利用Export/Import实用程序对ORACLE数据库进行备份、恢复的方法：增量卸出/装入。增量卸出是一种常用的数据备份方法，包括3个子类：

1. “完全”增量卸出

就是对整个ORACLE数据库进行完全卸出。如：

$ exp system/口令 inctype=complete full=y file=today.dmp

“增量型”增量卸出

即从ORACLE数据库中卸出上次卸出操作之后所有数据库的变化信息。如：

$exp system/口令 inctype=incremental file=today.dmp

增量型卸出文件的大小，可能只是完全卸出文件大小的1%，具体要看“新信息或更新过的信息”的总量而定。

“累积型”增量卸出

累积型卸出方式只是卸出自上次“完全” 卸出之后数据库中变化了的信息。用法如下：

$exp system/口令 inctype=cumulative file=today.dmp

DBA可以排定一个备份日程表，用数据卸出的三个不同方式合理高效地完成数据库的备份任务。比如DBA作如下安排：

星期一：完全卸出(F1)

星期二：增量卸出(I1)

星期三：增量卸出(I2)

星期四：累积卸出(C1)

星期五：增量卸出(I3)

星期六：增量卸出(I4)

如果在星期日，数据库遭到意外破坏，DBA可按以下步骤来恢复数据库：

用命令CREATE DATABASE重新生成你的数据库结构；

最近增量装入I4：$imp system/口令 inctype=system full=y file=I4

完全增量装入F1：$imp system/口令 inctype=restore full=y file=F1

累积增量装入C1：$imp system/口令 inctype=restore full=y file=C1

对于由累积装入或完全装入尚未能装入的信息，作增量装入：

$imp system/口令 inctype=restore full=y file=I3

$imp system/口令 inctype=restore full=y file=I4

注意 注意：

在I1和I2中的信息已包括在C1中了。

# ORACLE数据库的网络应用

ORACLE数据库有极强的网络功能，它支持各种网络协议，如TCP/IP、SPX/IPX、X.25、LU6.2、Decnet等等，支持异种网络协议互联、异种数据库互联。SQL\*Net是ORACLE的网络软件。

## SQL\*Net产品介绍

ORACLE不仅支持传统的主机─终端方式，还支持流行的Client/Server体系结构方式，使得应用与数据库核心分布处理，极大地提高了资源的利用率和处理效率。Client/Server的Server端安放ORACLE Server；Client端安放用户界面程序或ORACLE的Client端产品，如SQL\*Plus；SQL\*Net是ORACLE的网络软件，通过SQL\*Net可以实现Client与Server的连接，SQL\*Net是建立在某一网络协议之上的网络中间件产品，如所示：



SQL\*Net网络连接

目前，SQL\*Net有SQL\*Net V1和SQL\*Net V2两种版本。从上图中可以看出，必须在底层网络已经联通的前提下，才能对SQL\*Net作合适的配置。

SQL\*Net提供了以下网络功能：

网络透明性：在一个本地数据库上开发的应用，可以不加修改地分布到网络上去运行。

协议独立性：在任何一台计算机上编制的应用，都可以不加修改地分布到任何协议的网络上去运行。

异种机网络：通过多协议转换，可实现多种网络协议之间的互联。

位置透明：通过位置透明策略，可以使网络中数据，如同放在单一本地数据库中一样使用。以后当数据分布发生变化时，应用程序不用作任何修改。

自动字符集转换：两个通讯节点之间不同的字符集和NLS(国际语言支持)数据描述可以通过SQL\*Net来管理。

下面介绍SQL\*Net运行过程中相关的文件(以UNIX环境中TCP/IP协议为例)：

lsnrctl：放在“/ORACLE\_HOME/ product/8.1.7/bin”目录下，用于启动SQL\*Net V2.0监听进程。

listener.ora和tnsnames.ora：放在“/ORACLE\_HOME/ product/8.1.7/network/admin”子目录下，是SQL\*Net V2正常工作所需的配置文件及整个网络的配置文件。listener.ora是服务器监听进程网络配置文件，tnsnames.ora是客户机与服务器联络所需的网络配置文件。

TCP/IP的配置文件：与SQL\*Net有关的TCP/IP协议配置文件有两个：

/etc/hosts：存放整个网络环境中每个节点的IP地址

/etc/services：存放TCP/IP协议使用的各个端口的地址

## 配置客户机/服务器结构

使用ORACLE的SQL\*Net V.2连接客户机和服务器，需配置每个客户机和服务器。在服务器上需配置的文件有：listener.ora、/etc/services，在客户机上需配置tnsnames.ora文件。

### 配置listener.ora

listener.ora文件中存放了客户机与服务器连接所需要的监听地址，以及服务器启动监听进程时的信息。listener.ora文件缺省放置在/ORACLE\_HOME/product/8.1.7/network/admin之下。一个完整的listener.ora文件样式如下：

LISTENER =

(DESCRIPTION\_LIST =

(DESCRIPTION =

(ADDRESS\_LIST =

(ADDRESS = (PROTOCOL = TCP)(HOST = sun62)(PORT = 1521))

)

(ADDRESS\_LIST =

(ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC))

)

)

)

SID\_LIST\_LISTENER =

(SID\_LIST =

(SID\_DESC =

(SID\_NAME = PLSExtProc)

(ORACLE\_HOME = /opt/ORACLE/product/8.1.7)

(PROGRAM = extproc)

)

(SID\_DESC =

(GLOBAL\_DBNAME = ora817.huawei.com)

(ORACLE\_HOME = /opt/ORACLE/product/8.1.7)

(SID\_NAME = ora817)

)

)

其中，LISTENER是服务器中监听进程的名称；PROTOCOL表明所使用的网络协议，若是TCP/IP协议，则该值必须为“TCP”；HOST表示服务器的网络地址；PORT指TCP/IP协议在主机中所占用端口号，ORACLE缺省使用1521，是在安装ORACLE数据库时定义的，在安装时可以改变，安装完成后不能改变此值；SID\_NAME指服务器上运行的ORACLE数据库名称，该值应与.bash\_profile文件中的环境变量ORACLE\_SID相同（一般情况下此文件为缺省值“PLSExtProc”）；ORACLE\_HOME是指ORACLE数据库的主目录，该值也应与. bash\_profile文件中的环境变量ORACLE\_HOME相同。

### 配置tnsnames.ora文件

客户机为了和服务器连接，必须先和服务器上的监听进程联络。ORACLE通过tnsnames.ora文件中的连接描述符来说明连接信息。一般tnsnames.ora 是建立在客户机上的。如果是客户机/服务器结构，整个网络上只有一台机器安装了ORACLE数据库服务器，那么只需在每个要访问ORACLE服务器的客户机上定义该文件，在服务器上无需定义。但是，如果网络上有多台机器均安装了ORACLE数据库服务器，并且服务器之间有数据共享的要求，那么在每台服务器上都必须定义该文件。tnsnames.ora文件缺省放在/ORACLE\_HOME/ product/8.1.7/network/admin目录下。下面是tnsnames.ora的文件样式：

ORA817 =

(DESCRIPTION =

(ADDRESS\_LIST =

(ADDRESS = (PROTOCOL = TCP)(HOST = sun62)(PORT = 1521))

)

(CONNECT\_DATA =

(SERVICE\_NAME = ora817.huawei.com)

)

)

INST1\_HTTP =

(DESCRIPTION =

(ADDRESS\_LIST =

(ADDRESS = (PROTOCOL = TCP)(HOST = sun62)(PORT = 1521))

)

(CONNECT\_DATA =

(SERVER = SHARED)

(SERVICE\_NAME = ora817.huawei.com)

(PRESENTATION = http://admin)

)

)

EXTPROC\_CONNECTION\_DATA =

(DESCRIPTION =

(ADDRESS\_LIST =

(ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC))

)

(CONNECT\_DATA =

(SID = PLSExtProc)

(PRESENTATION = RO)

)

)

字段含义如下：

|  |  |
| --- | --- |
| ADDRESS\_LIST | 表示该客户机要经由多种协议与一台或多台服务器连接。在该样式文件中就表示该客户机要用TCP/IP协议来和服务器相连。 |
| PROTOCOL | 指明要连接使用的协议。 |
| SERVICE\_NAME | “SERVICE\_NAME”就是“Global Database Name”，ORACLE8i数据库使用“Global Database Name”来唯一标识自己，通常的格式为“name.domain”，此处的值为“ora817.huawei.com”。 |
| HOST | 是TCP/IP协议使用的服务器IP地址。 |
| PORT | 是TCP/IP使用的端口地址。 |
| SID | 指定要连接的服务器上ORACLE数据库的ORACLE\_SID。 |
| SERVER=DEDICATED | 表示用专用服务器连接ORACLE数据库。 |

下面是一个tnsnames.ora文件内容举例：

|  |  |
| --- | --- |
| *ora* = | 连接描述符名：*ora* |
| (description = | 描述 |
| (address = | 网络地址之一 |
| (protocol = tcp) | 网络协议（tcp表示TCP/IP协议） |
| (host = *129.9.114.22*) | 服务器IP地址：*129.9.114.22* |
| (port = *1251*) | 服务器端口号:*1251* |
| ) |  |
| (connect\_data = | 连接数据 |
| (sid = *ora7*) | 数据库标识名：*ora7* |
| ) |  |
| ) |  |

# 常用任务示例

## 如何恢复被误删的数据文件

1. Svrmgrl> alter database datafile 文件名 offline；或重启ORACLE；
2. Svrmgrl>alter database create datafile 原文件名 as 新文件名；
3. Svrmgrl>recover datafile 新文件名
4. Svrmgrl>alter database datafile 新文件名 online;
5. Svrmgrl>alter database open;

## 如何杀掉吊死session

1. 找出吊死session

Select sid,serial#,program,machine,lockwait from v$session;

杀死session

Svrmgrl>alter system kill session 'init1,init2';

🕮 说明：

其中init1为sid，init2为serial#。

## 如何修改字符集

以sys用户执行如下命令

Update props$ set value$='新字符集'

Where ltrim(name)='NLS\_CHARACTERSET'；

commit;

注意：如果有数据，不要修改数据集。

## 如何追加表空间

以sys用户：

Alter tablespace 表空间 add datafile 文件名（带路径）size Xm;

## 如何加大表的maxextents值

Alter table 表名 storage(maxextents 新值)

## 如何查询无效对象

1. 以sys用户登录SQL Plus

查询无效对象

Select substr(object\_name,1,30) object\_name,object\_type

from user\_objects

where status ='INVALID'

恢复失效存储过程

对于存储过程，如果存储过程或函数脚本中某个表或所调用的存储过程被drop或重新编译，则此存储过程可能变为invalid，正常情况下，再次调用此存储过程时，系统会自动编译使其变为valid，也可以手工编译：Alter procedure 名称 compile；

## 怎样分析SQL语句是否用到索引

ORACLE提供的策略分析器Explain plan能很好地分析SQL语句使用索引情况，分析步骤如下

1. 检查当前用户下是否存在策略分析表plan\_table

检查表结构是否正确（建立数据库时只有sys用户下存在此表），若没有，需要在当前用户下建此表：

CREATE TABLE PLAN\_TABLE

(

STATEMENT\_ID VARCHAR2(30),

TIMESTAMP DATE,

REMARKS VARCHAR2(80),

OPERATION VARCHAR2(30),

OPTIONS VARCHAR2(30),

OBJECT\_NODE VARCHAR2(128),

OBJECT\_OWNER VARCHAR2(30),

OBJECT\_NAME VARCHAR2(30),

OBJECT\_INSTANC NUMBER(38),

OBJECT\_TYPE VARCHAR2(30),

OPTIMIZER VARCHAR2(255),

SEARCH\_COLUMNS NUMBER(38),

ID NUMBER(38),

PARENT\_ID NUMBER(38),

POSITION NUMBER(38),

COST NUMBER(38),

CARDINALITY NUMBER(38),

BYTES NUMBER(38),

OTHER\_TAG VARCHAR2(255),

OTHER LONG

);

执行分析语句：

sql>delete from plan\_table;

sql>explain plan for

sql>select \* from tab where tname like 'T%';

sql>select object\_name,options,operation from plan\_table;

执行完上述三步，可查看“select \* from tab where tname like 'T%'”语句where条件是否用到索引。

## 怎样判断是否存在回滚段竞争

1. 查询等待值

Select class,count

from v$waitstat

Where class in ('system undo header','system undo block','undo header','undo block');

将查询得出的数值与所需的回滚段数目比较

Select sum(value) from v$sysstat

Where name in ('db\_block\_gets','consistent gets');

如果任何类型的waits数目比现有数值高出1%以上，则应该增加回滚段。

## 怎样手工跟踪函数/存储过程执行情况

如果有PLSql软件，可用该软件进行跟踪，本处介绍的是在无该工具的情况下，如何跟踪执行情况。

通过编写PL/SQL块可以手工跟踪函数/存储过程的执行情况，例如：

Sql>set serveroutput on --设置屏幕输出

Sql>declare

a1 integer:=1;

a2 integer;

a3 integer;

begin

a3:=testfunc(a1,a2); --假设输入a1，输出a2

dbms\_output.put\_line(a2);

dbms\_output.put\_line(a3);

end;

/

通过执行编写的PL/SQL块可以查看参数显示。

## 多种业务使用同一数据库如何分配回滚段

关键要分析业务性质：有些业务有大事务处理，但业务处理不频繁，这需要少量的大的回滚段；有些业务没有大事务处理，但业务处理频繁，这需要大量的小的回滚段。设计回滚段时要兼顾各个业务的特性，如果只考虑某一方，另一方的业务系统很可能受到影响。对于极个别的大事务处理，建议指定事务回滚段。

## 怎样倒出、倒入文本数据

### 倒出

用ORACLE spool命令，步骤：

1. 控制显示格式

sql>set numwidth 8 --integer类型输出长度

sql>set linesize 800

sql>set feedback off

sql>set echo off

sql>set heading off

sql>set pagesize 0

sql>set space 0

sql>set pause off

Sql>Spool c:\temp.txt

Sql>Select \* from 表名

Sql>Spool out

将文件中的SQL语句删掉

### 倒入

用Sql\*Load命令，控制文件（\*.ctl）格式如下：

Load data

Infile '数据文件名'

Append --往表中追加数据时有此项

Into table 表名

(

字段1 position(\*) char(l1),

字段2 position(\*) char(l2),

……

字段n position(\*) char(ln)

)

注：字段类型必须为char，日期型数据无法倒入。

## 如何更新当前数据库日志备份方式为archive

要修改日志备份方式

1. Svrmgrl>connect internal
2. Svrmgrl>startup mount
3. Svrmgrl>alter database archivelog;
4. Svrmgrl>alter database open;

## Unix环境下如何实现自动备份

备份原理：利用Unix操作系统的Cron命令定时启动ORACLE备份命令exp，然后再ftp到另一台Unix机器上去，实现自动备份。Cron的实质就是定时启动任务表Crontab中的命令，如果任务表的名称是Unix用户名，Cron就登录到此用户，执行任务表指定的命令或文件，与手工登录到此用户执行命令的区别是：Cron登录时不执行此用户下的.profile，所以需要设置运行环境。

备份计划：在凌晨1点做export导出备份，在凌晨4点将备份文件拷贝到另一台UNIX机器上，为此我们可进行如下操作：

### 设置运行环境

在ORACLE用户目录下编写文件“.bash\_profile”，对ORACLE的参数进行初始化，其内容参考如下：

export ORACLE\_BASE=/export/oracle

export ORACLE\_HOME=/export/oracle/product/8.1.7

export ORACLE\_SID=ora817

export LD\_LIBRARY\_PATH=${ORACLE\_HOME}/lib

export PATH=/usr/bin:${PATH}:${ORACLE\_HOME}/bin

### 倒出数据

export命令将数据库中的数据备份成一个二进制文件，它通常有三种模式：用户模式、表模式和整个数据库模式。本文拟采用表模式，备份之前，应先建立一个备份目录，以容纳备份文件，比如可建一个/backup目录。接着我们可在UNIX的ORACLE用户目录下（也可以是其它目录）建立一个文件ora—backup。

ora—backup文件对数据库做export导出，导出的文件名可以任意定，本文定为字母“icd\_”加当天日期，即假如当天日期是12月10号，则导出的文件名为“114\_1210.dmp”，以区别于其它日期的备份文件。

ora—backup文件内容：

|  |  |
| --- | --- |
| 值 | 说明 |
| . /disc1/home/ORACLE73/ora\_env | //设置运行环境，注意“.”与第一个“/”间必须有空格，$HOME=disc1/home/ORACLE73 |
| rq=`date +"%m%d"` | 日期变量，“`”为键盘数字1左边的“`”，不是回车键左边“'”，且“rq=`”之间不能有空格。 |
| rm /disc1/home/ORACLE73/backup/\* | 删除旧文件 |
| exp dlp114/dlp114 FILE=/disc1/home/ORACLE73/backup/114\_$rq.dmp TABLES=\(personal\_tel,public\_tel\) | 倒出指定表的数据，括号（）前的“\”是必须的  本命令用于导出dlp114用户表personal\_tel，public\_tel的数据（其口令亦为dlp114），导出文件及日志均放在/backup目录下。 |

### 异地备份

我们知道，通常可用FTP命令在两台主机间传输数据，但一般是通过交互方式实现的，即需要手工输入目标主机的IP地址、用户名、口令等。显然，这不符合自动备份的要求。所幸的是，我们可以通过编写一个.netrc的文件来达到目标。这一文件必须命名为.netrc，且必须存放在启动FTP命令的机器上的用户目录中，该文件的权限应禁止组内或其它用户进行读访问。这样，当用户使用FTP命令的时候，系统将会在该用户的目录中寻找.netrc文件，如果能够寻找到，将会首先执行该文件，否则，会交互式地提示用户输入用户名、口令等。

在使用FTP命令之前，应先在另一台作备份用的UNIX机器上建一目录，以容纳备份文件，本文建的目录是/pub 。需要指出的是，为了加快备份速度，两台主机之间的传输速率应尽可能的高，最好位于同一局域网上。

.netrc文件内容如表4-1所示。

.netrc文件内容

|  |  |
| --- | --- |
| 设置值 | 说明 |
| machine host2 | host2为作备份用的主机名，也可以指定IP地址，指定主机名时本机/etc/hosts文件中必须有说明。 |
| login ORACLE | “ORACLE”为备份主机上的一个用户 |
| password ORACLE | “ORACLE”用户的口令为“ORACLE” |
| 以下语句间不能有空行 | |
| macdef init | 定义一个名为init的宏，它将在自动注册进程的最后被执行 |
| bin | 文件的传输方式设为二进制 |
| lcd /backup | 进入本地工作目录/backup |
| cd /pub | 进入备份主机目录/pub |
| mput \* | 将/backup目录下的所有文件传输至备份主机 |
| bye | 退出FTP会话进程 |

.netrc文件编写完成后，使用下述命令：

chmod 600 .netrc

这样，.netrc文件就只能被该用户所访问。

### 启动备份进程

Cron是一个永久进程，它由/etc/rc.local启动执行。Cron检查/var/spool/cron/crontabs/目录中的文件，找到所要执行的任务和执行任务的时间。

Crontab文件的每一行由六个域（minutes、hours、day of month、month、day of week、 command）组成，域之间用空格或Tab分开，其中：

* minutes：分钟域，值的范围是0到59
* hours：小时域，值的范围是0到23
* day of month：日期，值的范围是1到31
* month：月份，值的范围是1到12
* day of week：星期，值的范围是0到6，星期日值为0
* command：所要运行的命令

如果一个域是“\*”，表明命令可以在该域所有可能的取值范围内执行。

如果一个域是由连字符隔开的两个数字，表明命令可以在两个数字之间的范围内执行（包括两个数字本身）。

如果一个域是由逗号隔开的一系列值组成的，表明命令可以在这些值组成的范围内执行。

如果日期域和星期域都有值，则这两个域都有效。

现在，我们编写一个文件，用以启动自动备份进程。值得注意的是，该文件只能在ORACLE用户名下用crontab －e 命令来编辑，否则将不会被定时执行，文件名定为ORACLE，文件将放在/var/spool/cron/crontabs 目录下。编辑完成后，可以在ORACLE的＄提示符下，用crontab －l命令来查看。

ora—env、ora—backup文件编写完成后，分别使用下述命令：

chmod 755 ora—env

chmod 755 ora—backup

这样，三个文件就都变成了可执行文件。

ORACLE文件内容：

　0 1 \* \* \*    /disc1/home/ORACLE73/ora—backup

　　＃ 每天23点对数据库执行备份

　　0 4 \* \* \*    ftp －i host2

＃ 每天4点将文件备份到另一台主机上 ，也可以是IP地址，一定要与.netrc文件匹配

经过以上的操作后，系统每天晚上将自动产生一个备份，并将备份文件拷贝到另一台机器上，实现自动备份。

## 怎样分析ORACLE故障

系统查询中断或变慢、系统挂起、系统宕机等等故障发生时，分析过程如下：

1. 分析alert\_<sid>.log文件

可以从该文件中分析故障发生的时间、现象日志、跟踪文件（\*.trc）。比如，大事务操作造成回滚段、临时段溢出而系统短暂中断或变慢时，日志文件中会记录表空间溢出；因为某条SQL语句造成故障时，跟踪文件会记录语句内容。

检查init<sid>.ora

该文件是ORACLE启动文件，任何参数的配置错误都会造成ORACLE不能启动，任何参数的不合理配置都可能造成系统故障，文件内容说明如下：

* gc\_rollback\_segments = 40 //回滚段数目（缺省为18）只有ORACLE7.3.x有此参数
* rollback\_segments= (r01,r02,r03,r04) //系统使用的回滚段
* Db\_block\_buffer=60000 //数据块缓冲区：120M 单位：块（2048Bytes）
* Share\_pool\_size=30000000 //共享池：30M 单位：byte
* Processes=200 //进程数
* Log\_buffer=163840 //日志缓冲区：160M 单位：byte（注意：要是块2048的整数倍）
* compatible = 7.3.0.0 //必须是7.3以上版本，ORACLE8不需要此参数

🕮 说明：

Db\_block\_buffer：是SGA的主要参数，数据存放的缓冲区。

Share\_pool\_size（共享池）：存放ORACLE所有脚本，例如存储过程等等，不需要太大。

Log\_buffer（日志缓冲区）：事务操作时的日志缓冲区，如果过小，一个简单的update操作就不能提交，造成系统短暂停顿。

rollback\_segments：指定ORACLE使用的回滚段，缺省为四个，创建回滚段后必须修改此值，重启ORACLE，所建的回滚段才起作用。

compatible：兼容性版本，缺省为7.1.0.0，有些SQL语句要求必须版本必须为7.3.0.0以上。

注意 注意：

上述参数配置只是一个参考，不能代表所有，在实际应用中，应该根据机器配置、使用业务不同而灵活配置。

# 小结

本教材阐述了ORACLE数据库的技术概念：ORACLE物理结构、系统全局区、进程等。介绍了ORACLE数据库的SQL\*Plus语句、怎样启动和关闭数据库、如何进行了数据库的备份和恢复、如何实现对数据库用户及权限的等管理知识。

ORACLE数据库采用了客户机/服务器的通信方式，本课程对ORACLE的客户机/服务器结构进行了介绍，例举出3个重要的网络配置文件并进行了说明。

请参考ORACLE公司发布的资料加深对ORACLE数据库知识的理解，为更好地完成TELLIN U-NICA系统维护管理任务做好知识铺垫。

# 附录A ORACLE数据字典与视图

当ORACLE数据库系统启动后，数据字典总是可用，它驻留在SYSTEM表空间中。数据字典包含视图集，在许多情况下，每一视图集有三种视图包含有类似信息，彼此以前缀相区别，前缀为USER、ALL和DBA。

前缀为USER的视图，为用户视图，是在用户的模式内。

前缀为ALL的视图为扩展的用户视图（为用户可存取的视图）。

前缀为DBA的视图为DBA的视图（为全部用户可存取的视图）。

在数据库ORACLE还维护了一组虚表（virtual table）记录当前数据库的活动，这些表称为动态性能表。动态性能表不是真正的表，许多用户不能直接存取，DBA可查询这些表，可以建立视图，给其它用户授予存取视图权。

ORACLE数据字典与视图

|  |  |
| --- | --- |
| 视 图 名 | 说 明 |
| ALL\_CATALOG | 为用户可存取的全部表、视图和序列 |
| ALL\_COL\_COMMENTS | 为用户可存取的表和视图列上的注释 |
| ALL\_COL\_PRIVS | 在列上授权，该用户或PUBLIC是被授与者 |
| ALL\_COL\_PRIVS\_MADE | 在列上授权，该用户为持有者或授与者 |
| ALL\_COL\_PRIVS\_RECD | 在列上授权，该用户或PUBLIC是被授与者 |
| ALL\_CONSTRAINTS | 在可存取表上的约束定义 |
| ALL\_CONS\_COLUMN | 关于在约束定义中可存取列的信息 |
| ALL\_DB\_LINKS | 用户可存取的数据库链 |
| ALL\_DBF\_AUDIT\_OPTS | 在对象建立时，所应用的缺省对象审计选择 |
| ALL\_DEPENDENCIES | 用户可存取的对象之间的从属关系 |
| ALL\_ERROES | 在用户可存取对象上的当前错误 |
| ALL\_INDEXES | 在用户可存取的表上的索引说明 |
| ALL\_IND\_COLUMNS | 在可存取的表上的索引列 |
| ALL\_OBJECTS | 用户可存取的对象 |
| ALL\_SEQUENCES | 用户可存取的序列说明 |
| ALL\_SNAPSHOTS | 用户可存取的全部快照 |
| ALL\_SOURCE | 用户可存取的全部存储对象文本源程序 |
| ALL\_SYNONYM | 用户可存取的全部同义词 |
| ALL\_TABLES | 用户可存取的表的说明 |
| ALL\_TAB\_COLUMNS | 用户可存取的表、视图、聚集的列 |
| ALL\_TAB\_COMMENTS | 用户可存取的表或视图上的注释。 |
| ALL\_TAB\_PRIVS | 在对象上授权，该用户或PUBLIC被授与者 |
| ALL\_TAB\_PRIVS\_MADE | 在对象上的授权或用户授权 |
| ALL\_TAB\_PRIVS\_RECD | 在对象上授权，该用户或PUBLIC是被授与者 |
| ALL\_TRIGGERS | 用户可存取的触发器 |
| ALL\_TRIGGER\_COLS | 显示用户持有的表中的列和用户所持有的触发器中列的使用，或者用户具有CREATE ANY TRIGGER特权时在所有触发器上列的使用 |
| ALL\_USERS | 关于数据库是所有用户的信息 |
| ALL\_VIEW | 用户可存取的视图文本 |
| AUDIT\_ACTIONS | 审计跟踪动作类型代码描述表 |
| CAT | 为USER\_CATALOG的同义词 |
| CHAINED\_ROWS | ANALYZE CHAINED ROWS命令的缺省值 |
| CLU | 为USER\_CLUSTERS的同义词 |
| COLS | 为USER\_TAB\_COLUMNS的同义词 |
| COLUMN\_PRIVILEGES | 在列上授权，用户是其授权者，被授与权者、持有者或授与PUBLIC |
| DBA\_2PC\_NEIGHBORS | 关于悬挂事务入和出连接的信息 |
| DBA\_2PC\_PENDING | 关于在PREPARED状态时失败的分式事务信息 |
| DBA\_AUDIT\_EXISTS | 由AUDIT EXISTS命令建立的审计跟踪记录 |
| DBA\_AUDIT\_OBJECT | 系统中全部对象的审计跟踪记录 |
| DBA\_AUDIT\_SESSION | 系统中涉及CONNECT和DISCONNECT的全部审计跟踪记录 |
| DBA\_AUDIT\_STATEMENT | 系统中涉及GRANT、REVOKE、AUDIT、NOAUDIT和ALTER SYSTEM语句的全部审计记录 |
| DBA\_AUDIT\_TRAIL | 为系统中全部审计记录的集合 |
| DBA\_BLOCKERS | 为会话集，它们具有别的会话正等待的一封锁，而它们本身不等待封锁 |
| DBA\_CATALAOG | 为全部数据库表、视图、同义词和序列 |
| DBA\_CLUSTERS | 数据库中全部聚集的说明 |
| DBA\_CLU\_CLOUMNS | 表列对聚集列的映射 |
| DBA\_COL\_COMMENTS | 在所有表和视图的列上的注释 |
| DBA\_COL\_PRIVS | 在数据库列上的全部授权 |
| DBA\_CONSTRAINTS | 在数据库全部表上的约束定义 |
| DBA\_CONS\_CLOUMNS | 关于约束定义中全部列的信息 |
| DBA\_DATA\_FILES | 关于数据文件的信息 |
| DBA\_ DB\_LINKS | 在数据库中的全部数据链 |
| DBA\_ DDL\_LOCKS | 数据库中当前所有全部DDL封锁和所有未完成的DML封锁请求 |
| DBA\_ DEPENDENCIES | 全部对象之间的从属关系 |
| DBA\_DML\_LOCKS | 数据库中当前所持有DDL封锁和所有未完成的DML封锁请求 |
| DBA\_ERRORS | 数据库中全部存储对象上的当前错误 |
| DBA\_EXP\_FILES | 输出文件说明 |
| DBA\_EXP\_OBJECTS | 已有增量输出的对象 |
| DBA\_EXP\_VERSION | 最后的输出会话的版本 |
| DBA\_EXTENTS | 数据库中全部段的范围 |
| DBA\_FREE\_SPACE | 在所有表空间中未用的范围 |
| DBA\_INDEXES | 数据库中全部索引的说明 |
| DBA\_IND\_COLUMN | 全部表和聚集上的索引列 |
| DBA\_LOCKS | 在数据库中持有的全部封锁和未完成请求的封锁（包括DML和DDL封锁） |
| DBA\_OBJECT | 在数据库中定义的全部聚集、数据库链、索引、包、包体、序列、同义词、表和视图 |
| DBA\_OBJECT\_SIZE | 数据库中的全部PL/SQL对象 |
| DBA\_OBJ\_AUDIT\_OPTS | 为全部表和视图的审计选择 |
| DBA\_PRIV\_AUDIT\_OPTS | 特权审计选择 |
| DBA\_PROFILES | 赋给每个环境文件的资源限制 |
| DBA\_ROLES | 在数据库中已有的全部角色 |
| DBA\_ROLE\_PRIVS | 授权给用户或角色的角色的说明 |
| DBA\_ROLLBACK\_SEGS | 回滚段的说明 |
| DBA\_SEGMENTS | 分配给全部数据库段的存储 |
| DBA\_SEQUENCES | 在数据库中全部序列的说明 |
| DBA\_SNAPSHOTS | 在数据库中的全部快照 |
| DBA\_SNAPSHOTS\_LOGS | 在数据库中的全部快照日志 |
| DBA\_SOURCE | 在数据库中全部存储对象的源文本 |
| DBA\_SYNONYMS | 在数据库中的全部同义词 |
| DBA\_STMT\_AUDIT\_OPTS | 为当前系统审计选择 |
| DBA\_SYS\_PRIVS | 授权给用户或角色的系统特权 |
| DBA\_TABLES | 在数据库中的全部表的说明 |
| DBA\_TABLESSPACES | 数据库中的全部表空间的说明 |
| DBA\_TAB\_CLOUMNS | 全部表、视图和聚集中的列 |
| DBA\_TAB\_COMMENTS | 在数据库中全部表和视图上的注释 |
| DBA\_TAB\_PRIVS | 在数据库中对象上的全部授权 |
| DBA\_TRIGGERS | 在数据库中全部触发器的说明 |
| DBA\_TRIGGERS\_COLS | 显示由用户定义或在任何用户表上的触发器中列的用法 |
| DBA\_TS\_QUOTAS | 为全部用户的表空间份额 |
| DBA\_USERS | 关于数据库全部用户的信息 |
| DBA\_VIEWS | 数据库中全部视图的文本 |
| DBA\_WAITERS | 等待封锁的全部会话和持有该锁的会话 |
| DICT | 为DICTIONARY的同义词 |
| DICTIONARY | 数据库字典表和视图的说明 |
| DICT\_COLUMNS | 数据库字典表和视图中的列的说明 |
| EXCEPTIONS | 违反完整性约束的信息 |
| GLOBAL\_NMAE | 当前数据库的全局名 |
| IND | 为USER\_\_INDEXES的同义词 |
| INDEX\_STATE | 存储VAILDATE INDEX命令的信息 |
| OBJ | 为USER\_ OBJECT的同义词 |
| RESOURCE\_COST | 每种资源的费用 |
| ROLE\_ROLE\_PRIVS | 授权给其它角色的角色的信息 |
| ROLE\_SYS\_PRIVS | 授权给角色的系统特权的信息 |
| ROLE\_TAB\_PRVS | 授权给角色的表特权的信息 |
| SEQ | 为USER\_SEQUENCES的同义词 |
| SEESSIONS\_PORIVS | 用户当前可用的特权 |
| SESSION\_ROLES | 用户当前已使其使能打角色 |
| SYN | 为USER\_SYNONYMS的同义词 |
| SYSTEM\_PRILEGE\_MAP | 为系统特权代码的说明表 |
| TABLE\_PRIVILEGES | 在对象上授权 |
| TABLE\_PRIVILEGE\_MAP | 为存取特权代码的说明表 |
| TABS | 为USER\_TABLES的同义词 |
| USER\_AUDIT\_OBJECT | 涉及对象审计跟踪记录 |
| USER\_AUDIT\_SESSION | 涉及连接或删除连接的全部审计跟踪记录 |
| USER\_AUDIT\_STATEMENT | 为用户发出的GRANT、REVOKE、AUDIT、NOAUDIT、ALL SYSTEM语句的审计跟踪项 |
| USER\_AUDIT\_TRAIL | 与用户有关的审计跟踪项 |
| USER\_CATALOG | 为用户所持有的表、视图、同义词和序列 |
| USER\_CLUSTERS | 为用户持有的聚集的说明 |
| USER\_CLU\_CLOUMNS | 用户的表列到聚集的映射 |
| USER\_COL\_COMMENTS | 在用户的表或视图的列上的注释 |
| USER\_COL\_PRIVS | 在列上的授权，该用户是持有者、授权者或被授与者 |
| USER\_COL\_PRIVS\_MADE | 为用户持有的对象的列上的全部授权 |
| USER\_COL\_PRIVS\_RECD | 该用户是被授权者的列上的授权 |
| USER\_CONSTRAINT | 在用户表上的约束定义 |
| USER\_CONS\_COLUMNS | 由用户持有约束定义中的列的信息 |
| USER\_DB\_LINKS | 为用户持有的数据库链 |
| USER\_DEPENDENCIES | 用户的对象之间的从属关系 |
| USER\_ERRORS | 用户的存储对象上的当前错误 |
| USER\_EXTRNTS | 属于用户对象的段的范围 |
| USER\_FREE\_SPACE | 用户可存取的表空间中未用的范围 |
| USRE\_INDEXES | 用户自己的索引说明 |
| USER\_IND\_CLOUMNS | 用户索引的列或用户表上的列 |
| USER\_OBJECTS | 用户所持有的对象 |
| USER\_OBJECT\_SIZE | 用户的PL/SQL对象 |
| USER\_OBJ\_AUDIT\_OPTS | 为用户的表和审计选择 |
| USER\_RESOURCE\_LIMITS | 为当前用户的资源限制 |
| USER\_ROLE\_PRIVS | 特权给用户的角色 |
| USER\_SEGMENT | 属于用户对象的数据库段的存储分配 |
| USER\_SEQUENCE | 用户自己的序列的说明 |
| USER\_SNAPSHOTS | 用户可查看的快照 |
| USER\_SNAPSHOT\_LOGS | 用户可持有的快照日志 |
| USER\_SOURCE | 属于用户的全部存储对象的源文本 |
| USER\_SYNONYM | 用户专用同义词 |
| USER\_SYS\_PRIVS | 特权给用户的系统特权 |
| USER\_TABLES | 用户持有表的说明 |
| USER\_TABLESPACES | 可存表空间的说明 |
| USER\_TAB\_COLUMNSS | 用户的表、视图和聚集的列 |
| USER\_TAB\_COMMENTS | 用户所持的表和视图上的注释 |
| USER\_TAB\_PRIVS | 用户为授权者、持有者或被授权者的对象上的授权 |
| USRE\_TAB\_PRIVS\_MADE | 用户所持有的对象的全部特权 |
| USER\_TAB\_PRIVS\_RECD | 用户为被授权者的对象上授权 |
| USER\_TRIGGRS | 用户触发器的说明 |
| USER\_TRIGGRS\_COLS | 用户所持有的或在用户表上的触发器中的列的用法 |
| USER\_TS\_QUOTAS | 用户在表空间上的份额 |
| USER\_USERS | 关于当前用户的信息 |
| USER\_VIEWS | 用户持有的视图的文本 |

# 附录B 动态性能表

|  |  |
| --- | --- |
| 表名 | 说明 |
| V$ACCESS | 显示数据库中的对象信息 |
| V$ARCHIVE | 数据库系统中每个索引的归档日志方面的信息 |
| V$BACKUP | 所有在线数据文件的状态 |
| V$BGPROCESS | 描述后台进程 |
| V$CIRCUIT | 有关虚拟电路信息 |
| V$DATABASE | 控制文件中的数据库信息 |
| V$DATAFILE | 控制文件中的数据文件信息 |
| V$DBFILE | 构成数据库所有数据文件 |
| V$DB\_OBJECT\_CACHE | 表示库高速缓存中被缓存的数据库对象 |
| V$DISPATCHER | 调度进程信息 |
| V$ENABLEDPRIVS | 那些特权接通 |
| V$FILESTAT | 文件读/写统计信息 |
| V$FIXED\_TABLE | 显示数据库中所有固定表、视图和派生表 |
| V$INSTANCE | 当前实例状态 |
| V$LATCH | 每类闩锁的信息 |
| V$LATCHHOLDER | 当前闩锁占有者的信息 |
| V$LATCHNAME | 在V$LATCH表中表示的闩锁的译码闩锁名 |
| V$LIBRARYCACHE | 库高速缓冲存储管理统计 |
| V$LICENSE | 许可限制信息 |
| V$LOADCSTAT | SQL\*Loader在直接装入执行过程中的编译统计 |
| V$LOCK | 有关封锁和资源信息，不包含DDL封锁 |
| V$LOG | 控制文件中的日志文件信息 |
| V$LOGFILE | 有关日志文件信息 |
| V$LOGHIST | 控制文件中的日志历史信息 |
| V$LOG­HISTORY | 日志历史中所有日志的归档日志名 |
| V$NLS\_PARAMETERS | NLS参数的当前值 |
| V$OPEN\_CURSOR | 每一个用户会话期当前已打开和分析的光标 |
| V$PARAMETER | 当前参数值的信息 |
| V$PROCESS | 当前活动进程的信息 |
| V$QUEUE | 多线索信息队列的信息 |
| V$REVOVERY\_LOG | 需要完成介质恢复的归档日志 |
| V$RECOVERY\_FILE | 需要介质恢复的文件状态 |
| V$REQDIST | 请求时间直方图，分为12个范围 |
| V$RESOURCE | 有关资源信息 |
| V$ROLLNAME | 所有在线回滚段的名字 |
| V$ROLLSTAT | 所有在线回滚段的统计信息 |
| V$ROWCACHE | 数据字典活动的统计信息（每一个包含一个数据字典高速缓存的统计信息） |
| V$SESSION | 每一个当前会话期的会话信息 |
| V$SESSION\_WAIT | 列出活动会话等待的资源或事件 |
| V$SESSTAT | 对于每一个当前会话的当前统计值 |
| V$SESS\_IO | 每一个用户会话的I/O统计 |
| V$SGA | 系统全局区统计信息 |
| V$SGASTAT | 系统全局区的详细信息 |
| V$SHARED\_SERVER | 共享服务器进程信息 |
| V$SQLAREA | 共享光标高速缓存区的统计信息，每一个有一个共享光标的统计信息 |
| V$SQLTEXT | 属于SGA中的共享SQL光标的SQL语句文本 |
| V$STATNAME | 在V$SESSTAT表中表示的统计信息的译码统计名 |
| V$SYSSTAT | 表V$SESSETA中当前每个统计的全面的系统值 |
| V$THREAD | 从控制文件中得到线索信息 |
| V$TIMER | 以百分之一秒为单位的当前时间 |
| V$TRANSACTION | 有关事务的信息 |
| V$TYPE\_SIZE | 各种数据库成分的大小 |
| V$VERSION | ORACLE Server中核心库成员的版本号，每个成员一行 |
| V$WAITSTAT | 块竞争统计，当时间统计可能时，才能更新该表 |

# 附录C SQL语言运算符与函数

表1 各种运算符

|  |  |
| --- | --- |
| 运算符 | 含 义 |
| ＝ | 相等 |
| ！＝或＜＞ | 不等于 |
| ＞ | 大于 |
| ＞＝ | 大于等于 |
| ＜ | 小于 |
| ＜＝ | 小于等于 |
| IN(列表) | 等于列表中的任意值 |
| BETWEEN值1 AND值2 | 大于等于值1并且小于等于值2 |
| LIKE％或－ | 模式匹配。“％”匹配0个或任意个字符，“－”匹配一个字符 |
| IS NULL | 空值 |
| IS NOT NULL | 非空值 |
| NOT | 逻辑非 |
| AND | 逻辑与 |
| OR | 逻辑或 |

表2 字符函数列表

|  |  |  |
| --- | --- | --- |
| 函数 | 功能 | 注释 |
| ASCII(char) | 计算char的第一个字符的ASCII值或EBCDIC码值 | 函数返回值取决于计算机系统采用的字符 |
| CHAR(n) | 计算ASCⅡ码值或EBCDIC码值是n的字符 | 函数n依赖于计算机系统采用的字符集，n的取值为0～127或0～254之间 |
| INITCAP(char) | 将char串口的每个单词的首字母变成大写，其余字母变为小写 | 单词之间用数字、空格、逗号、顿号、冒号、分号、句号、1、@、#、$等字符分隔 |
| INSTR(char1,char[,m[,n]]) | 求char1中从m位置起char2第n次出现的位置 | m,n缺省值为1，当＞0时，表示从char1的首部起始(从左向右)正向搜索；n＜0时，表示从char1的尾部起始（从右向左）反向搜索 |
| LENGTH(char) | 计算字符串char的长度 |  |
| LOWER(char) | 将char中所有的字母改成小写 |  |
| LPAD(char1,n[,char2]) | 从左侧用char2补齐char1至长度n | char2省略时，用空格填充，n＜char1的长度时，表示截取char1从左至右侧n个字符 |
| RPAD(char1,n[,char2]) | 从右侧用char2补齐char1至长度n | char2省略时，用空格填充，n＜char1的长度时，表示截取char1右侧n个符 |
| LTRIM(char[,SET]) | 把char1中最左侧的若干个字符去掉，以使其首字符不在SET中 | SET表示单个字符组在的字符集合。SET若被省略时，表示截取char左边的前置空格 |
| RTRIM(char[,SET]) | 把char中最右侧的若干个字符去掉，以使其尾字符不在SET中 | SET表示单个字符组成的字符集合。SET若被省略时，表示截取char右边的后置空格 |
| REPLACE(char1,char2[,char3] | 将char1中出现的所有char2用char3来代替 | char2和char3同时被省略时，函数返回NULL，仅char3省略时，则表示删除char1中出现的所有char2 |
| SOUNDEX(char) | 求与char中一个或多个单词发音相同的字符串 |  |
| SUBSTR(char,m[,n]) | 返回char中第m个字符起始n个字符长的子串 | n省略时，表示截取char中第m个字符后的子串 |
| TRANSLATE(char1,from,to) | 将from字符集转换为to字符集，char中以from表达的字符用to中相对应的字符所代替 |  |
| UPPER(char) | 将char中所有的字母改变大写 |  |

表3 数值函数列表

|  |  |  |
| --- | --- | --- |
| 函数 | 功能 | 注释 |
| ABS(n) | 计算n的绝对值 | 允许是数值型常数、数值型列名、包含一个有效数值的文字串或包含一个有效数值的字符型列名 |
| CELL(n) | 计算大于或等于n的最小整数 | 数学中的向上取整运算 |
| FLOOR(n) | 计算大于或等于n的最大整数 | 数学中的向下取整运算 |
| MOD(m,n) | 计算m除以n的余数n＝0时，返回m | MOD(m,1)＝0说明m是一个整数 |
| POWER(m,n)  ROUND(m,[,n]) | 计算m的n次方，n＞0时，将m四舍五入到小数点右边n位n＜0时，将m四舍五入到小数点左边n位 | n要求为整数，否则出错n＝0或n被省略表示对m进行取整 |
| SIGN(n) | 判断n的正负 | 如果n＞0，函数返回1；n＝0，函数反加0，n＜0，函数反回 |
| SQRT(n)  TRUNC(m,[,n]) | 计算n的平方根，n＞0时，将m小数点右边n位截断；n＜0时，将m小数点左边的n位后各位截断，并添加n个0 | n＜0时，函数返回NULLn＝0或n被省略表示对m进行取整 |

表4 日期函数列表

|  |  |  |
| --- | --- | --- |
| 函数 | 功能 | 注释 |
| ADD\_MONTHS(d,n) | 计算d加上n个月的日期 | n要求为整数，n＞0时，返回d之后n个月的日期；n＜0时，返回d之前n个月的日期 |
| LAST\_DAY(d) | 计算d所在月份最后一天的日期 | 用来确定给定月份中的天数 |
| MONTHS\_BETWEEN  (d1,d2) | 计算d1和d2之间相隔的月数 | 返加结果＞0时，表示d1晚于d2；返回结果＜0时，表示d1早于d2 |
| NEXT\_DAY(d,s) | 计算晚于d的第一个s的日期 | s要求是’Sunday’，’Monday’，  ’Tuesday’，’Wednesday’，’Thursday’，  ’Friday’或’Saturday’ |
| SYSDATE | 求系统当前日期和时间 |  |

表5 聚组函数列表

|  |  |
| --- | --- |
| 函 数 | 功 能 |
| AVG(e) | 计算一组行中e值的平均值 |
| COUNT(e) | 计算一组行中e值为非空值的行数 |
| COUNT(\*) | 计算表中的行数(包括重复值和空值) |
| MAX(e) | 计算一组行中e值的最大值 |
| MIN(e) | 计算一组行中e值的最小值 |
| STDDEV(e) | 计算一组行中e值的标准差 |
| SUM(e) | 计算一组行中e值的总和 |
| VARIANCE(e) | 计算一组行中e值的方差 |

1. **基础查询**

-------更改列标题的三种方法

select au\_id 作者编号,作者姓名=au\_lname, phone as 电话 from authors

-------where条件查询

select \* from authors

where state='ca' --查询住在加州的作者

-------between ..and...条件查询

select \* from titles

where price between 10 and 20 --查找图书价格在10至20元间的图书

------逻辑运算符、关系运算符的应用

select \* from titles

where price>=10 and price <=20 --查找图书价格在10至20元间的图书

-----like的查询,%表示：匹配0—n个任意字符

select \* from titles

where title like '%busy %' --查找书名中含有Busy的图书

-----like的查询,[]表示:匹配在方括号内列出的任意一字符

select \* from titles --查找书名以C或S或T开头的书名

where title like '[cst]%'---或者'[c,s,t]%'

-----like的查询,[^]表示:匹配不在括号内列出的任意一字符

select \* from titles

where title like '[^Y]%'---查找书名不以Y开头的书名

-----like的查询,\_ 表示:匹配1个任意字符

select \* from titles

where title like '%b\_sy%'

-----like的查询中转义字符escape的用法

-----空值检索

select \* from titles

where price is null -------查找价格为null的记录

select \* from titles

where price is not null -------查找价格不为null的记录

-----限制记录行检索:top 关键字限制结果集中返回的行数

select top 3 \* from titles --查找前3本图书

select top 3 \* from titles

order by price desc --查找价格最贵的3本书

-----限制记录行检索:distinct 关键字 限制重复行

select distinct state from authors --查找所有著书作者的州名

-----按书店编号和定单号分组统计销量

select stor\_id,ord\_num,sum(qty) from sales

group by stor\_id,ord\_num

with cube

---按图书类别和出版社统计图书销量

select type,pub\_id,sum(advance) from titles

group by type,pub\_id

with cube

**2、多表查询与系统函数**

select \* from authors,titleauthor

select \* from titleauthor

select \* from authors

inner join titleauthor

on authors.au\_id=titleauthor.au\_id

select \* from titleauthor

inner join authors

on authors.au\_id=titleauthor.au\_id

select \* from authors

inner join titleauthor

on authors.au\_id=titleauthor.au\_id

inner join titles

on titleauthor.title\_id=titles.title\_id

select \* from authors

left join titleauthor

on authors.au\_id=titleauthor.au\_id

select \* from titleauthor

left join authors

on authors.au\_id=titleauthor.au\_id

select \* from titleauthor

right outer join authors

on authors.au\_id=titleauthor.au\_id

join titles

on titleauthor.title\_id=titles.title\_id

select \* from titles

select \* from titleauthor

full outer join authors

on authors.au\_id=titleauthor.au\_id

right join titles

on titleauthor.title\_id=titles.title\_id

select au\_id ,au\_fname from authors

union

select title\_id,price from titles

union

select au\_id ,au\_fname from authors

select charindex('abc','cccabcfffabc')

select patindex('%abc%','aaaccabcaaccabc')

select dateadd(year,10,'2005-2-5')

select datediff(month,'2005-1-1','2005-2-25')

select datepart(month,'2005-2-25')

select getdate()

select power(2,3)

select isdate('205-2-25')

select isnull('sdd','dfgdf')

select isnull(null,'fdsf')

select isnumeric('a123')

select convert(char(10),getdate(),108)

**3、子查询**

select stor\_id from sales

where qty >

(

select avg(qty) from sales

)

select au\_id,au\_fname from authors

where au\_id in

(select au\_id from titleauthor

group by au\_id

having count(au\_id)>=2

)

select au\_id,au\_fname from authors

where au\_id not in

(select au\_id from titleauthor)

select au\_id,au\_fname from authors

where not exists

(

select au\_id from titleauthor

where authors.au\_id=titleauthor.au\_id

)

select \* from sales

select \* from stores

select title\_id,title,price

from titles

where price < 20 and title\_id in

(

select title\_id from sales

where stor\_id in

(

select stor\_id from stores

where state='CA'

)

group by title\_id

having sum(qty)\*price>250

)

**4、数据库管理**

create database mydata

on primary

(

name='mydata',

filename='c:\aaa\mydata.mdf',

size=5mb,

maxsize=50,

filegrowth=5

)

log on

(

name='mydatalog',

filename='c:\aaa\mydatalog.ldf',

size=5mb,

filegrowth=5

)

alter database mydata

add file

(

name='mydata2',

filename='c:\aaa\mydata2.ndf',

size=5mb,

maxsize=50,

filegrowth=5

)

select \* from sysfiles

select \* from sysfiles1

select \* from sysfilegroups

alter database mydata

add filegroup group1

alter database mydata

remove filegroup group1

alter database mydata

modify name= mydata2

alter database mydata

modify file

(

name='mydata',

maxsize=150

)

create database mydata1

on primary

(

name='mydata1',

filename='c:\aaa\mydata1.mdf',

size=5mb,

maxsize=50,

filegrowth=5

),

(

name='mydata11',

filename='c:\aaa\mydata11.ndf',

size=5mb,

maxsize=50,

filegrowth=5

),

filegroup group1

(

name='mydata12',

filename='c:\aaa\mydata12.ndf',

size=5mb,

maxsize=50,

filegrowth=5

),

(

name='mydata13',

filename='c:\aaa\mydata13.ndf',

size=5mb,

maxsize=50,

filegrowth=5

)

log on

(

name='mydatalog1',

filename='c:\aaa\mydatalog1.ldf',

size=5mb,

filegrowth=5

),

(

name='mydatalog2',

filename='c:\aaa\mydatalog2.ldf',

size=5mb,

filegrowth=5

)

alter database mydata1

modify filegroup group1 default

dbcc shrinkdatabase(mydata1,80,truncateonly)

dbcc shrinkfile(mydata1,3,truncateonly)

drop database mydata1

sp\_helpdb mydata2

sp\_databases

sp\_rename 'mydata2','mydata3','database'

sp\_dboption 'mydata3','select into/bulkcopy','true'

**5、表的管理与数据类型**

create table students

(

sno int identity(1000,1) primary key,

sname varchar(20) not null,

age int not null,

address varchar(30)

)

select \* from students

create table #tt

(

sno int identity(1000,1) primary key,

sname varchar(20) not null,

age int not null,

address varchar(30)

)

select \* from #tt

create table ##ttt

(

sno int identity(1000,1) primary key,

sname varchar(20) not null,

age int not null,

address varchar(30)

)

select \* into #tt1 from students

select \* into tt1 from students

select \* from #tt1

select au\_id,state from

(select \* from authors) s

select \* from #tt

alter table students

add phone varchar(11)

alter table students

drop column address

alter table students

alter column age tinyint

select \* from students

sp\_help students

drop table #tt

sp\_rename 'students','stus'

select \* from stus

sp\_help stus

delete stus

insert stus values('zhan san',20,'65025215')

select \* from students

insert students values('johnson',20,'dfdsaf')

insert students(sname,age) values('john',20)

--insert students(age) values(20)

set identity\_insert stu1 on

insert students(sno,sname,age,address) values(1006,'johnson',20,'dfdscf')

set identity\_insert students off

insert students(sname,age,address) values('johnson',20,'dfdscf')

select \* from students

select \* into stu1 from students

select \* from stu1

sp\_help stu1

delete stu1

insert stu1 select \* from students

select \* into authors1 from authors

select \* from authors1

alter table authors1

drop column au\_fname

alter table authors1

add sex char(2)

insert authors1(au\_id,au\_lname,phone,address,city,state,zip,contract,sex) select au\_id,au\_lname,phone,address,city,state,zip,contract,'M' from authors

update authors1

set state='CC',city='bk'

where state='CA'

delete authors1

where state='CC'

truncate table authors1

select \* from authors1

sp\_addtype phonetype,'varchar(8)'

sp\_addtype age ,tinyint

sp\_droptype age

select \* from systypes

**6、数据完整性**

create table students

(

sno int identity(1000,1) primary key,

sname varchar(20) not null,

age int not null,

sid varchar(18) not null

)

sp\_help students3

create table students1

(

sno int identity(1000,1) constraint pk\_sno primary key,

sname varchar(20) not null,

age int not null,

sid varchar(18) not null

)

create table students2

(

sno int identity(1000,1) ,

sname varchar(20) not null,

age int not null,

sid varchar(18) not null

)

alter table students2

add constraint pk\_sno2 primary key(sno)

create table students3

(

sno int identity(1000,1) primary key,

sname varchar(20) not null,

age int not null,

sid varchar(18) unique

)

create table students4

(

sno int identity(1000,1) primary key,

sname varchar(20) not null,

age int not null,

sid varchar(18) constraint uk\_sid unique not null

)

sp\_help students4

create table scores1

(

sno1 int not null constraint fk\_sno1 foreign key(sno1) references students4(sno),

cno char(5) not null,

grade float,

constraint pk\_smo\_cno primary key(sno1,cno)

)

sp\_help scores1

sp\_help titleauthor

select \* from authors

select \* from titleauthor

delete authors

where au\_id='172-32-1176'

update authors

set au\_id='111-11-1111'

where au\_id='213-46-8915'

alter table titleauthor

drop constraint FK\_\_titleauth\_\_au\_id\_\_0519C6AF

alter table titleauthor

add constraint fk11 foreign key(au\_id) references authors(au\_id) on delete cascade on update cascade

create table students5

(

sno int identity(1000,1) primary key,

sname varchar(20) not null,

age int not null,

sex char(2) not null constraint che1 check (sex in ('M','F')),

sid varchar(18) constraint uk\_sid1 unique not null

)

sp\_help students5

create table students6

(

sno int identity(1000,1) primary key,

sname varchar(20) not null,

age int not null default 18,

sex char(2) not null constraint che11 check (sex in ('M','F')),

sid varchar(18) constraint uk\_sid11 unique not null

)

sp\_help students7

create table students7

(

sno int identity(1000,1) primary key,

sname varchar(20) not null,

age int not null,

sex char(2) not null constraint che21 check (sex in ('M','F')),

sid varchar(18) constraint uk\_sid21 unique not null

)

alter table students7

add constraint def1 default 18 for age

alter table students7

drop constraint def1

create rule rule\_age

as @age between 0 and 255

sp\_bindrule rule\_age,'students7.age'

sp\_unbindrule 'students7.age',futureonly

sp\_help students7

drop rule rule\_age

create default def\_age

as 18

sp\_bindefault def\_age,'students7.age'

sp\_unbindefault 'students7.age'

drop default def\_age

**7、视图与索引**

if exists(select \* from sysobjects where name='v\_author' and xtype='V')

drop view v\_author

go

create view v\_author

with encryption

as

select au\_id,state,city

from authors

where state='CA'

with check option

select \* from v\_author

select \* from authors

update v\_author

set city='bj'

where au\_id='213-46-8915'

sp\_helptext v\_author

select text from syscomments where id =

(select id from sysobjects where name='v\_author' and xtype='V')

insert v\_author values('222-22-2222','CA','fg')

sp\_help

if exists(select \* from sysobjects where name='v\_author1' and xtype='V')

drop view v\_author1

go

create view v\_author1

as

select au\_id,au\_lname,au\_fname,phone,contract,city

from authors

select \* from v\_author1

insert v\_author1 values('111-11-1111','fgdf','fhf','100 111-1111',1,'kh')

select \* from authors

if exists(select \* from sysobjects where name='v\_author2' and xtype='V')

drop view v\_author2

go

create view v\_author2

as

select authors.au\_id,au\_lname,au\_fname,phone,contract,city,titleauthor.au\_id au\_id1,title\_id

from authors

join titleauthor

on authors.au\_id=titleauthor.au\_id

sp\_help titleauthor

select \* from v\_author2

--insert v\_author2 values('211-11-1111','fgdf','fhf','200 111-1111',1,'kh','211-11-1111','BU1032')

insert v\_author2(au\_id,au\_lname,au\_fname,phone,contract,city) values('211-11-1111','fgdf','fhf','200 111-1111',1,'kh')

insert v\_author2(au\_id1,title\_id) values ('211-11-1111','BU1032')

--update v\_author2

--set au\_lname='AAAAAAAAAAA',title\_id='BU1032'

--where au\_id='172-32-1176'

update v\_author2

set au\_lname='AAAAAAAAAAA'

where au\_id='172-32-1176'

update v\_author2

set title\_id='BU1032'

where au\_id='172-32-1176'

create table t2

(

sno int primary key,

sid varchar(18) unique not null,

age int not null

)

sp\_help t2

create index index\_age on t2(age desc)

with pad\_index,fillfactor=15

create table product

create table t1

(

sno int primary key,

sid varchar(18) unique not null,

sname varchar(20) not null,

age int not null

)

create view view\_t1

with schemabinding

as

select sno,sname,age

from dbo.t1

create unique clustered index sno\_index

on view\_t1(sno)

create index sname\_index

on view\_t1(sname)

**8、SQL编程**

select @@version

select @@servername

print @@error

raiserror('hello',12,1)

select \* from master..sysmessages

where error>50000

--sp\_addmessage 50001,3,'hello world',@lang='english'

/\*sp\_addmessage 50002,3,'你好',@lang='english'

\*/

sp\_help

waitfor delay '00:00:05'

select \* from authors

declare @n int,@name varchar(20)

select @n=contract,@name=au\_fname from authors

where au\_id='527-72-3246'

if @n=1

print 'can contract'

else

print 'can not contract'

print @name

select au\_id,state1=case state

when 'CA' then '加州'

when 'KS' then '肯莎州'

else 'unkown'

end

from authors

select au\_id,state1=case

when state='CA' then '加州'

when state='KS' then '肯莎州'

else 'unkown'

end

from authors

select \* from scores

select sno,A001=sum(case

when cno='A001' then score

else 0

end

),

A002=sum(case

when cno='A002' then score

else 0

end

),

A003=sum(case

when cno='A003' then score

else 0

end

)

from scores

group by sno

declare @n int

while (@n<10)

begin

print 'hello'

select @n=@n+1

end

if (select avg(price) from titles)>20 or (select max(price) from titles)>50

return

while 1=1

begin

update titles

set price=price+0.5

if (select avg(price) from titles)>20 or (select max(price) from titles)>50

break

end

select \* from authors

wher state='CA'

select \* from sales

go

update authors

set state='AA'

where au\_id='172-32-1176'

update sales

set qty=qty+5

where title\_id='BU1032'

go

if exists(select \* from sysobjects name='aa' and xtype='v')

drop view aa

go

create view aa

as

select \* from sales

**9、存储过程**

if exists(select \* from sysobjects where name='searchauthor' and xtype='P')

drop procedure searchauthor

go

create proc searchauthor

with encryption

as

select \* from authors

go

exec searchauthor

sp\_helptext searchauthor

select text from syscomments where id=(

select id from sysobjects where name='searchauthor' and xtype='P')

if exists(select \* from sysobjects where name='searchauthor1' and xtype='P')

drop procedure searchauthor1

go

create proc searchauthor1

@state char(2)

as

select \* from authors

where state=@state

go

exec searchauthor1 'CA'

if exists(select \* from sysobjects where name='searchauthor2' and xtype='P')

drop procedure searchauthor2

go

create proc searchauthor2

@price1 money,@price2 money

as

select \* from titles

where price between @price1 and @price2

go

exec searchauthor2 10,20

select \* from titleauthor

if exists(select \* from sysobjects where name='Addauthor' and xtype='P')

drop procedure Addauthor

go

create proc Addauthor

@au\_id char(11),

@title\_id char(6),

@au\_ord int,

@royaltyper int

as

insert titleauthor values (@au\_id,@title\_id,@au\_ord,@royaltyper)

if @@error<>0

return -1

else

return 1

go

sp\_help titleauthor

insert titleauthor values('172-32-1176','BU1032',1,1000)

exec Addauthor '172-32-1176','BU1111',1,10

declare @a int

exec @a=Addauthor '172-32-1176','PC1035',1,10

print @a

declare @a int

exec @a=Addauthor '111-11-1176','PC1035',1,10

print @a

if exists(select \* from sysobjects where name='Searchtitles' and xtype='P')

drop procedure Searchtitles

go

create proc Searchtitles

@title\_id char(6),

@title varchar(20) output,

@price money output

as

select @title=title,@price=price

from titles

where title\_id=@title\_id

go

declare @title varchar(20),@price money

exec Searchtitles 'BU1032',@title output,@price output

print @title+' '+convert(varchar(10),@price)

select \* from titles

if exists(select \* from sysobjects where name='Inserttitles' and xtype='P')

drop procedure Inserttitles

go

create proc Inserttitles

@title\_id char(6),

@title varchar(80),

@type varchar(20),

@pub\_id char(4),

@price money,

@advance money,

@royalty int,

@ytd\_sales int,

@notes varchar(200),

@pubdate datetime,

@avg\_price money output

as

insert titles values(@title\_id ,@title ,

@type ,@pub\_id ,@price ,@advance ,@royalty ,

@ytd\_sales ,@notes ,@pubdate)

select @avg\_price=avg(price) from titles

if @@error<>0

return -1000

else

return 100

go

select \* from titles

declare @n int,@avgprice money

exec @n=Inserttitles 'BU9988','good book','business','1389',30,5000,10,4500,'very good','2005-3-22',@avgprice output

print 'n='+convert(char(10),@n)+' '+'avgprice='+convert(char(20),@avgprice)

**10、游标与自定义函数**

create function (@title\_id char(6))

returns varchar(80)

as

begin

declare @title varchar(80)

select @title=title from titles where title\_id=@title\_id

return @title

end

select dbo.SearchBook('bu1032')

--查找图书的版税不符合标准版税的图书，并将其版税更改为标准版税

--written by 王传标, 2004.11.12

if exists(select \* from sysobjects where name='Correct\_royalty')

drop procedure Correct\_royalty

go

create procedure Correct\_royalty

as

begin

set nocount on

declare @title\_id char(6),@title varchar(80),@type varchar(12),@pub\_id char(4)

declare @price money,@advance money,@royalty int,@royalty1 int,@ytd\_sales int

create table #t(title\_id char(6),title varchar(80),type varchar(12),

pub\_id char(4),price money,advance money,royalty int,royalty1 int,ytd\_sales int)

declare title\_cur cursor for

select title\_id,title,type,pub\_id,price,advance,royalty,ytd\_sales

from titles

where ytd\_sales is not null

open title\_cur

fetch next from title\_cur into @title\_id,@title,@type,@pub\_id,@price,@advance,@royalty,@ytd\_sales

while (@@fetch\_status=0)

begin

select @royalty1=royalty from roysched where title\_id=@title\_id and (@ytd\_sales between lorange and hirange)

if @royalty<>@royalty1

begin

insert #t values (@title\_id,@title,@type,@pub\_id,@price,@advance,@royalty,@royalty1,@ytd\_sales)

update titles set royalty=@royalty1 where current of title\_cur

end

fetch next from title\_cur into @title\_id,@title,@type,@pub\_id,@price,@advance,@royalty,@ytd\_sales

end

close title\_cur

deallocate title\_cur

select \* from #t

end

go

exec Correct\_royalty

**11、游标与触发器**

if exists(select \* from sysobjects where name='ins\_authors' and xtype='TR')

drop trigger ins\_authors

go

create trigger ins\_authors

on authors

for insert

as

declare @au\_id char(11),@au\_fname varchar(20),@state char(2)

print 'insert new authors'

select @au\_id=au\_id,@au\_fname=au\_fname,@state=state

from inserted

print 'au\_id:'+@au\_id+' au\_fname='+@au\_fname+' '+'state='+@state

go

select \* from authors

insert authors values('222-11-1111','gd','gdg','101 111-1111','dfggf','vff','CA','12345',1)

select \* into authors1 from authors

if exists(select \* from sysobjects where name='ins\_authors1' and xtype='TR')

drop trigger ins\_authors1

go

create trigger ins\_authors1

on authors1

for insert

as

declare @au\_id char(11),@au\_fname varchar(20),@state char(2)

print 'insert new authors'

declare cur\_auth cursor for

select au\_id,au\_fname,state

from inserted

open cur\_auth

fetch next from cur\_auth into @au\_id,@au\_fname,@state

while @@fetch\_status=0

begin

print 'au\_id:'+@au\_id+' au\_fname='+@au\_fname+' '+'state='+@state

fetch next from cur\_auth into @au\_id,@au\_fname,@state

end

close cur\_auth

deallocate cur\_auth

go

insert authors1 select \* from authors

if exists(select \* from sysobjects where name='del\_authors' and xtype='TR')

drop trigger del\_authors

go

create trigger del\_authors

on authors1

for delete

as

declare @au\_id char(11),@au\_fname varchar(20),@state char(2)

print 'delete authors'

select @au\_id=au\_id,@au\_fname=au\_fname,@state=state

from deleted

print 'au\_id:'+@au\_id+' au\_fname='+@au\_fname+' '+'state='+@state

go

select \* from authors1

delete authors1 where au\_id='111-11-1111'

truncate table authors1

if exists(select \* from sysobjects where name='upd\_authors' and xtype='TR')

drop trigger upd\_authors

go

create trigger upd\_authors

on authors1

for update,insert

as

declare @au\_id char(11),@au\_fname varchar(20),@state char(2)

print 'delete authors'

select @au\_id=au\_id,@au\_fname=au\_fname,@state=state

from deleted

print 'au\_id:'+@au\_id+' au\_fname='+@au\_fname+' '+'state='+@state

print 'insert authors'

select @au\_id=au\_id,@au\_fname=au\_fname,@state=state

from inserted

print 'au\_id:'+@au\_id+' au\_fname='+@au\_fname+' '+'state='+@state

go

select \* from authors1

update authors1

set state='KK'

where au\_id='111-11-1111'

sp\_helptrigger authors1

sp\_helptext upd\_authors

select \* from titleauthor

select \* from sales

create view v\_auth\_sales

as

select au\_id,titleauthor.title\_id,au\_ord,royaltyper,

stor\_id, ord\_num,ord\_date,qty,payterms,sales.title\_id titled\_id1

from sales join titleauthor

on sales.title\_id=titleauthor.title\_id

select \* from v\_auth\_sales

if exists(select \* from sysobjects where name='v\_del' and xtype='TR')

drop trigger v\_del

go

create trigger v\_del

on v\_auth\_sales

instead of delete

as

declare @title\_id char(6)

select @title\_id=title\_id from deleted

delete sales where title\_id=@title\_id

if @@error<>0

raiserror('error',1,1)

delete titleauthor where title\_id=@title\_id

if @@error<>0

raiserror('error',1,1)

go

delete v\_auth\_sales where title\_id='BU1032'

select \* from sales

create trigger upd\_sales

on sales

for insert,delete,update

as

if update(qty)

print 'update qty'

go

update sales

set qty=10

where stor\_id='6380'

update sales

set payterms='dfgggg'

where stor\_id='6380'

**12、事物**

select \* from authors

select \* from sales

create proc tran\_test

as

begin transaction

delete sales

where stor\_id='7066'

if @@error<>0

begin

rollback transaction

return -10000

end

delete authors

where au\_id='172-32-1176'

if @@error<>0

begin

rollback transaction

return -10001

end

update authors

set state='YY'

where au\_id='213-46-8915'

if @@error<>0

begin

rollback transaction

return -10002

end

commit transaction

exec tran\_test

create proc tran\_test1

as

set xact\_abort on

begin transaction

delete sales

where stor\_id='7066'

delete authors

where au\_id='172-32-1176'

update authors

set state='YY'

where au\_id='213-46-8915'

commit transaction

exec tran\_test1

set transaction isolation level read uncommitted

begin tran

select \* from authors

where state='KK'

commit tran

set transaction isolation level read committed

begin tran

select \* from authors

where state='KK'

waitfor delay '00:00:20'

select \* from authors

where state='KK'

commit tran

if exists(select \* from sysobjects where name='printauthors' and xtype='P')

drop proc printauthors

go

create proc printauthors

as

declare @au\_id char(11),@au\_fname varchar(20),@state char(2),@phone char(15)

declare cur\_auth cursor for

select au\_id,au\_fname,state,phone from authors

open cur\_auth

fetch next from cur\_auth into @au\_id,@au\_fname,@state,@phone

while @@fetch\_status=0

begin

print 'au\_id:'+@au\_id

print 'au\_fname:'+@au\_fname

print 'state:'+@state

print 'phone:'+@phone

print '---------------------'

fetch next from cur\_auth into @au\_id,@au\_fname,@state,@phone

end

close cur\_auth

deallocate cur\_auth

go

select \* from titles

exec printauthors

**13、锁与特殊数据类型**

if exists(select \* from sysobjects where name='sp\_textcopy' and xtype='P')

drop procedure sp\_textcopy

go

CREATE PROCEDURE sp\_textcopy

@srvname varchar (30),

@login varchar (30),

@password varchar (30),

@dbname varchar (30),

@tbname varchar (30),

@colname varchar (30),

@filename varchar (30),

@whereclause varchar (40),

@direction char(1)

AS

DECLARE @exec\_str varchar (255)

SELECT @exec\_str =

'c:\textcopy /S ' + @srvname +

' /U ' + @login +

' /P ' + @password +

' /D ' + @dbname +

' /T ' + @tbname +

' /C ' + @colname +

' /W "' + @whereclause +

'" /F ' + @filename +

' /' + @direction

EXEC master..xp\_cmdshell @exec\_str

go

select \* from pub\_info

exec sp\_textcopy 'tb500','sa','sa','pubs','pub\_info','logo','c:\0877.bmp',"where pub\_id='0877'" ,'O'

exec sp\_textcopy 'tb500','sa','sa','pubs','pub\_info','logo','c:\cibab.bmp',"where pub\_id='0877'" ,'I'

exec sp\_textcopy 'tb500','sa','sa','pubs','pub\_info','logo','c:\cibab1.bmp',"where pub\_id='0877'" ,'O'

if exists(select \* from sysobjects where name='sp\_textcopy' and xtype='P')

drop procedure sp\_textcopy

go

CREATE PROCEDURE sp\_textcopy

@srvname varchar (30),

@login varchar (30),

@password varchar (30),

@dbname varchar (30),

@tbname varchar (30),

@colname varchar (30),

@filename varchar (30),

@whereclause varchar (40),

@direction char(1)

AS

DECLARE @exec\_str varchar (255)

SELECT @exec\_str =

'textcopy /S ' + @srvname +

' /U ' + @login +

' /P ' + @password +

' /D ' + @dbname +

' /T ' + @tbname +

' /C ' + @colname +

' /W "' + @whereclause +

'" /F ' + @filename +

' /' + @direction

EXEC master..xp\_cmdshell @exec\_str

Go

**14、数据安全性**

sp\_addlogin 'tb','123123','pubs'

select \* from syslogins

sp\_defaultdb 'tb','northwind'

sp\_password '123123','123456','tb'

sp\_droplogin 'tb'

sp\_grantlogin 'tb503\think'

sp\_denylogin 'tb503\think'

sp\_revokelogin 'tb503\think'

sp\_grantdbaccess 'tb','thinkbank'

select \* from sysusers

sp\_addsrvrolemember 'tb','dbcreator'

sp\_dropsrvrolemember 'tb','dbcreator'

sp\_helpsrvrolemember

sp\_srvrolepermission dbcreator

sp\_srvrolepermission sysadmin

sp\_addrolemember 'db\_owner','thinkbank'

sp\_helprolemember 'db\_owner'

sp\_addrole 'manager'

sp\_addrolemember 'manager','thinkbank'

grant select on syslogins to manager

grant select on syslogins to thinkbank

grant select on syslogins to manager,thinkbank

grant create database to manager,thinkbank

**15、数据库与备份DTS**

select \* from authors

update authors

set state='KK'

where au\_id='172-32-1176'

update authors

set state='KK'

where au\_id='213-46-8915'

update authors

set state='KK'

where au\_id='238-95-7766'

sp\_addumpdevice 'disk','pubsbak','c:\pubs.4133'

backup database pubs to pubsbak

sp\_addumpdevice 'disk','pubsbak1','c:\pubs.4134'

backup database pubs to pubsbak1 with differential

restore database pubs from pubsbak --1 with standby

湖北省软件产业基地培训中心机试试卷

机 试 (ASP.Net)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **学期 : 第三学期** | |  | **课程 :.Net** | |
|  | |  |  | |
| **日期 :** | |  | **时间 : 2小时** | |
|  | | | | |
|  | **总分数 :100分** | | |  |

|  |  |  |
| --- | --- | --- |
| **姓名：** |  | **考号：** |

|  |
| --- |
| **证件号码：** |

**注意：考试结束试卷必须交回，不交回试卷者成绩无效**

**问题：个人信息提交页面设计**

在做本题目前，要求学员的计算机有以下环境：

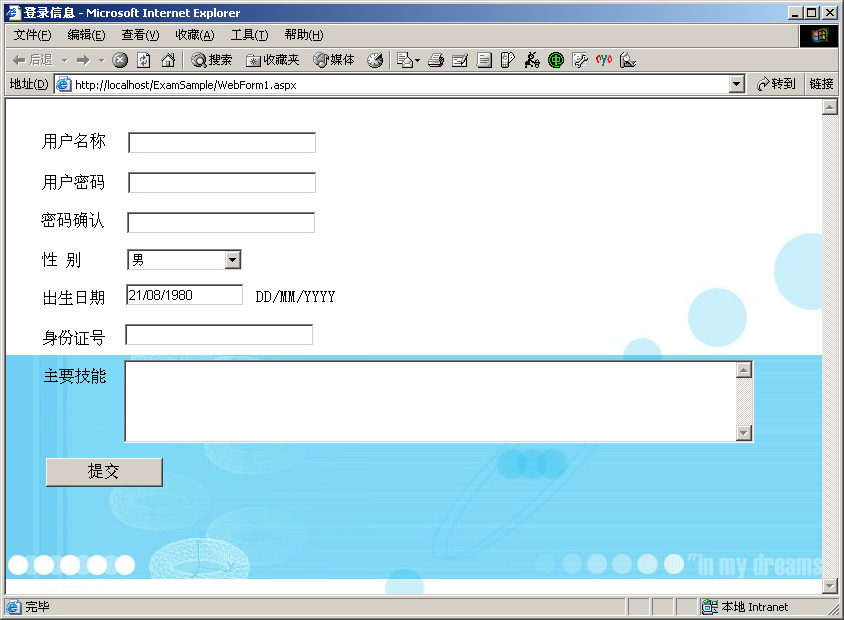
1. IIS4.0以上版本；

２、.Net环境；

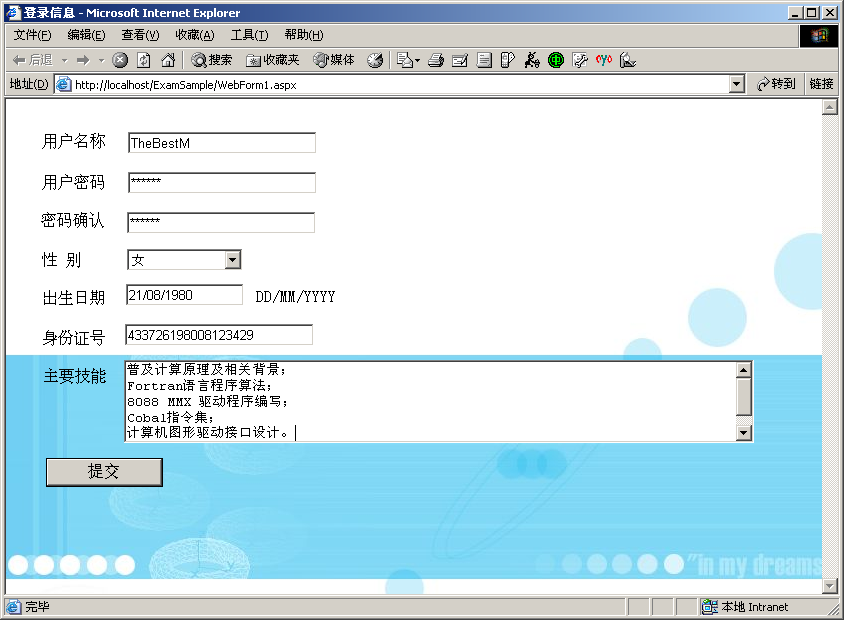
３、SQL Server数据库。

|  |  |  |
| --- | --- | --- |
| 本试题利用Asp.Net中的服务器控件完成一个Web个人信息页面的设计，当用户输入的信息不符合要求，会提示错误信息，用户输入的信息符合要求，把输入的数据提交到服务器端的数据库保存。  要求实现数据库、信息页面的设计，同时编码实现验证、提交到数据库等相关的工作，数据提交成功后，显示提交成功的页面。 | | |
| **一、数据库** | | |
|  | | 数据库名称：PerInfo |
|  | | 数据库表结构如下： |
| |  |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | --- | | 表名称 | | User | | 主键 | | Username | | | 序号 | 字段名称 | | 字段说明 | 类型 | 位数 | 属性 | 备注 | | 1 | Username | | 用户名 | Varchar | 16 | 非空 |  | | ２ | Password | | 用户口令 | Varchar | 16 | 非空 |  | | ３ | Sex | | 性别 | char | 2 |  | “男”或“女” | | ４ | Birthday | | 出生日期 | Varchar | 16 |  | 格式：DD/MM/YYYY | | 5 | CardID | | 身份证号 | Varchar | 32 | 非空 | 位数18位或15位 | | 6 | Skill | | 主要技能 | Varchar | 255 |  |  | | | |
| **二、参考实现步骤：** | | |
| 1 | 建立数据库 | |
|  | 建立数据库PerInfo，表结构见数据库设计； | |
| 2 | 创建Login.Aspx页面，如参考图１； | |
| 3 | 页面在IIS发布时，性别栏只允许选择”男或者”女”，允许输入的一个页面信息如参考图2； | |
| 4 | 用户输入数据完毕，点提交按钮时，进行页面校验，验证项包括： | |
|  | A、用户名、用户口令、身份证号三项不为空； | |
|  | B、出生日期格式：DD/MM/YYYY，并且DD从1-31，MM从1-12，YYYY从1900-2004； | |
|  | C、口令和确认口令两项数据相同，大小写敏感； | |
|  | D、身份证号位数等于15或18位，并且要求全部是数字； | |
|  | E、身份证上的出生日期必须和以上输入的出生日期符合，否则给出提示信息“非法的省份证号码或者出生日期！” | |
| 5 | 数据验证全部符合要求后，编码实现与数据库PerInfo连接、如果数据库中已存在此用户名，应给出提示信息，要求用户重新输入，如果是新用户，则把这些数据写入到数据库，并显示提交成功的提示页面Success.Aspx，如参考图3。 | |
| 提示：  1、使用Asp.net服务器校验控件进行校验；  2、页面背景和图片请自行设计，要求规范整洁； | | |
| **三、调试和运行页面；** | | |
| **四、注意事项** | | |
| 1. 编码规范作为考核标准之一； 2. 进行合理的页面设计； | | |

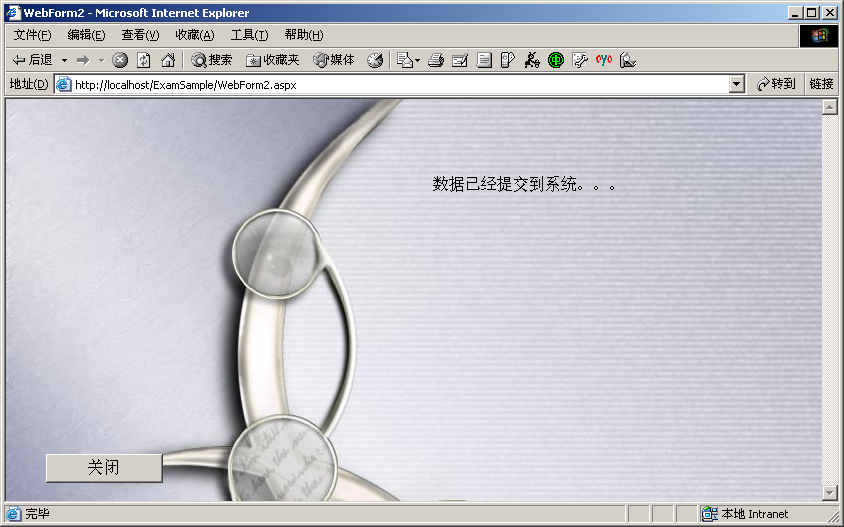
参考图１



参考图2



参考图3



深圳市福田区商报东路景明达酒店—莲兴苑8A（到北大医院或景新花园下）

龙控科技开发有限公司（主要做安防、网控等软、硬件）

以下试题有一定难度，做完1/3即可能被公司录用。所有题不要求完整，但要细量详细。（共10题）

（所有题不要求具体语言，vb,c,c++,java都可）

1．写一个输入整数（如123456789）转换为二进制的程序，如输入32则输出1000000；

2．写出对100个整数进行排序的程序，要求用两种不同的排序方法；

3．写出一个字符串替换程序如输入”Welcome to comp”,将“co”替换成”abc”，即为” Wel abcme to abcmp”，要求不能作任何系统提供的函数如strrepalce等。

4．设计数据库中有两个表，当主表中的一条记录删除时，副表中的相关记录全部自动删除。

5．写两个类，子类中的方法在父类中实现。

6．从数据库中查一个表，将表中的所有数据输出到磁般中的一个文件中。

7．用vb或其它写一个stock server和clinel通讯的例子；

8．一个线通讯的题，记不清了！

9．写一个父线程控制线程运行的例子。

10．谈谈你对龙控公司的认识和你想在龙控公司所从事的岗位和工作。

瑞科科技开发有限公司张经理

技术题

1 . **什么是CSS？它的能做些什么？**

CSS就是一种叫做样式表（stylesheet）的技术。也有的人称之为层叠样式表（Cascading Stylesheet）。在主页制作时采用CSS技术，可以有效地对页面的布局、字体、颜色、背景和其它效果实现更加精确的控制。 只要对相应的代码做一些简单的修改，就可以改变同一页面的不同部分，或者页数不同的网页的外观和格式。

它的作用可以达到：

（1）在几乎所有的浏览器上都可以使用。

（2）以前一些非得通过图片转换实现的功能，现在只要用CSS就可以轻松实现，从而更快地下载页面。

（3）使页面的字体变得更漂亮，更容易编排，使页面真正赏心悦目。

（4）你可以轻松地控制页面的布局 。

（5）你可以将许多网页的风格格式同时更新，不用再一页一页地更新了。你可以将站点上所有的网页风格都使用一个CSS文件进行控制，只要修改这个CSS文件中相应的行，那么整个站点的所有页面都会随之发生变动。

面试考题

**第一部分(每题10分)**

1. 请使用你的编码风格写出一个可以实现加，减，乘，除的类
2. 求表达式的值：1+3+5+7+…..+n
3. 部门表：

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| |  |  | | --- | --- | | 部门ID | 部门名称 | | 1 | 部门1 | | 2 | 部门2 | | 3 | 部门3 |   员工表   |  |  |  | | --- | --- | --- | | 员工ID | 员工姓名 | 部门ID | | 1 | 姓名1 | 1 | | 2 | 姓名2 | 5 | | 3 | 姓名3 | 2 | |  |  |  | |

问题：

1. 用一条SQL语句显示所有人员的姓名及所在部门，没有部门的用“未标志”代替
2. 使用触发器写出当删除部门表中的一个部门时。同时删除员工表中对应的记录
3. 请列举一些您用到过的设计模式以及在什么情况下使用该模式
4. 什么叫存储过程
5. 写出一个简单的委托
6. 如何处理几十万条并发数据？

可以采用分业存储的方法，尽量简化搜索条件，将数据分类存储！！！

1. 用什么方法比较2个字符串相似，且在系统运行时长驻内存？
2. SQL SERVER中的表存到多少记录时，速度明显减慢？
3. Session 有什么重大BUG，微软提出什么方法加以解决？
4. 找错题

试题1 ：

|  |
| --- |
| void test2()  {  char string[10]，str1[10];  char\*str1 = “0123456789”;  strcpy(string,str1);  } |

试题二：

|  |
| --- |
| void test2()  {  char string[10],str1[10];  int i;  for(i=0;i<10;i++)  {  Str1[i]=’a’;  }  Strcpy(string,str1);  } |

试题3：

|  |
| --- |
| Void test3(char\*str1)  {  char string[10];  if(strlen(str1)<=10)  {  Strcpy(string,str1);  }  } |

11〉使用Javascript删除下列表中值为“1223” 的节点

<select name=”select”>

<option>abcd</option>

<option>1223</option>

</select>

12〉请写出一个CSS，将表的背景色身子为蓝色

**第二部分（每题3分）**

1. 面向对象的语言具有\_\_\_\_\_\_\_\_\_性、 \_\_\_\_\_\_\_\_\_性、\_\_\_\_\_\_\_\_\_性
2. 能用foreach 遍历访问的对象需要实现\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_接口或声明\_\_\_\_\_\_\_\_\_\_\_方法的类型。
3. GC是什么？为什么要有GC？
4. String s = new String (“xyz”);创建了几个String Objict?
5. abstract class 和interface有什么区别？
6. 启动一个线程是用run()还是start()?
7. 接口是否可继承接口？抽象类是否可实现（implements）接口？抽象类是否可继承实体类（concrete class）?
8. 构造器Constructor是否可被override?
9. 是否可可继承String类？
10. try{}里有一个return语句，那么紧跟在这个try后的finally{}里的Code会不会被执行，什么时候被执行，在 retuan前还是后？
11. 两个对象值相同（x.equalsy(y)==true）,但却可有不同的hash code.，这句话对不对？
12. switch是否作用在byte上，是否能作用在long上。是否能作用在string上？
13. 当一个 线程进入一个对象的一个 synchronized方法后，其它线程是否可进入此对象的其他方法？
14. abstract的method是否可同时是static,是否可同时是native,是否可同时是synchronized ?
15. List,Set.Map是否继承自Collection接口？
16. Set里元素 是不能重复的，那么用什么方法来区分重复与否呢？是用==还是equals()?它们有何区别？
17. 数组有没有length()这个方法？String有没有length()这个方法?
18. sleep()和wait有什么区别?
19. short s1 = 1;s1=s1+I；有什么错？short s1 =1;s1+=1;有什么错？
20. 谈谈final,finalize的区别。
21. 如何处理几十万条并发数据？
22. Session 有什么重大BUG，微软提出什么方法加以解决？
23. 进程和线程的区别？
24. 堆和栈的区别？
25. 成员变量和成员函数前加static的作用？
26. ASP.NET与 ASP 相比，主要有那些进步？
27. 公司要求开发一个继承 System.Windows.Forms.ListView类的组件，要求达到以下的特殊功能：点击ListView 各列列头时，能按照点击列的每行值进行重排视图中的所有行（排序的方式如DataGrid相似）。根据您的知识，请简要谈一下您的思路。
28. 给定以下 XML文件，完成算法流程图。

<FileSystem>

<DriverC>

<Dir DirName=”MSDOS622”>

<File FileName=”Command.com”></File>

</Dir>

<File FileName=”MSDOS.SYS”></File>

<File FileName=”IO.SYS”></File>

</DriverC>

</FileSystem>

请画出遍历所有文件名（FileName）流程图（请使用递归算法）。

1. 根据委托（dalegate）的知识，请完成以下用户控件中代码片段的填写；

namespace test

{

public delegate void OnDBOperate{}

public class UserControlBase:System.Windows.Forms.UserControl

{

public event OnDBOperate OnNew;

privatevoidtoolBar\_ButtonClick(Objectsender:,System.Windows.Forms.ToolBarButtonClickEvent Argse

{

if(e.Button.Equals(BtnNew))

{

//请在以下补齐代码用来调用OnDBOperate委托签名的OnNew事件。

}

}

}

}

30．根据线程安全的相关知识，分析以下代码，当调用test方法时I >10 时是否会引起死锁？并简要说明理由。

Public void test(int i)

{

lock(this)

{

if(i>10)

{

i--;

test(i)

}

}

}

**第三部分逻辑分析（不记分）**

房间里有三盏灯，房外有三个开关，在房外看不见房内的情况，但可以任意拌动开关，你只能进门观察一次，你用什么方法区分哪个开关控制那一盏灯？

1 ASP题目：用ACCESS数据库建表如下：

|  |  |  |
| --- | --- | --- |
| 自动编码 | ID | PWD |
|  |  |  |
|  |  |  |

编写COM.asp，建立数据库连接

1. 创建index.asp

|  |  |
| --- | --- |
| **ID** |  |
| **PWD** |  | **确 定** |

1. 如果ID又重复，则提示，不允许重复
2. 客户端验证，ID不能为空

3 应用数据库显示所有记录

|  |  |  |
| --- | --- | --- |
| 自动编码 | ID | PWD |
|  |  |  |
|  |  |  |
|  |  |  |

1. 查询：

|  |  |  |
| --- | --- | --- |
| **ID** |  | **查询** |

1. 点击查询，显示相应对应的ID和PASSWORD；
2. ID不能为空（客户端验证）
3. 制作一个简单的界面设计，将上述功能实现，并尽最大可能美化界面；
4. SQL server 2000创建一个存储过程：

输出参数（省份编号）

查出相对应的城市

一、选择题

1. Oracle存储结构正确的是（）

A. tablespace--segment--osblock--block

B. tablespace--segment--extent--block

C. tablespace-- extent -- segment--block

D. tablespace-- extent -- block -- segment

1. 表EMPLOYEES结构为

(

EMPLOYEE\_ID NUMBER(6) not null primary key,

FIRST\_NAME VARCHAR2(20) unique,

LAST\_NAME VARCHAR2(25),

EMAIL VARCHAR2(25) not null,

PHONE\_NUMBER VARCHAR2(20),

HIRE\_DATE DATE,

JOB\_ID VARCHAR2(10),

SALARY NUMBER(8,2),

COMMISSION\_PCT NUMBER(2,2),

MANAGER\_ID NUMBER(6),

DEPARTMENT\_ID NUMBER(4)

)

以下哪几个语句可算出表中记录，多选 （）

A.select count(\*) from EMPLOYEES

B.select count(EMPLOYEE\_ID) from employees

C.select count(FIRST\_NAME) from employees

D.select count(EMAIL) from employees

E.select count(JOB\_ID) from employees

1. 表EMPLOYEES的结构为:

(

EMP\_ID NUMBER(4) NOT NULL

LAST\_NAME VARCHAR2(30) NOT NULL

FIRST\_NAME VARCHAR2(30)

DEPT\_ID NUMBER(2)

JOB\_CAT VARCHAR2(30)

SALARY NUMBER(8,2)

)

下面哪条语句可以查出各个dept\_id的最大salary值和最小salary值

并且它的最值小于5000最大值大于15000？（）

A. SELECT dept\_id, MIN(salary), MAX(salary)

FROM employees

WHERE MIN(salary) < 5000 AND MAX(salary) > 15000;

B. SELECT dept\_id, MIN(salary), MAX(salary)

FROM employees

WHERE MIN(salary) < 5000 AND MAX(salary) > 15000

GROUP BY dept\_id;

C. SELECT dept\_id, MIN(salary), MAX(salary)

FROM employees

HAVING MIN(salary) < 5000 AND MAX(salary) > 15000;

D. SELECT dept\_id, MIN(salary), MAX(salary)

FROM employees

GROUP BY dept\_id

HAVING MIN(salary) < 5000 AND MAX(salary) > 15000;

E. SELECT dept\_id, MIN(salary), MAX(salary)

FROM employees

GROUP BY dept\_id, salary

HAVING MIN(salary) < 5000 AND MAX(salary) > 15000;

1. SQL Server中在A数据库下如何访问B数据库中的C表：（）

A．在A数据库中通过建立数据库链接指向B，然后直接访问select \* from c

B．在B数据库中授权给A,然后直接访问select \* from c

C．直接访问select \* from c

D．直接访问select \* from B..c

1. Oracle数据库中用户A的一个存储过程用到用户B下的表Tab\_B及函数Fun\_B,正确的赋权语句是( )

A.在A用户模式执行 grant select on Tab\_B to A; grant execute on Fun\_B to A;

B.在B用户模式执行 grant select on Tab\_B to A; grant execute on Fun\_B to A;

C.在A用户模式执行 grant select on Tab\_B to B; grant execute on Fun\_B to B;

D.在B用户模式执行 grant select on Tab\_B to B; grant execute on Fun\_B to B;

1. 以下数据库是Oracle语法的是：（）

A.select sysdate into v\_time from dual;

B.select v\_time = sysdate;

C.set v\_time:=sysdate;(v\_time为已定义的时间类型变量)

D. v\_time=sysdate

1. 分别用 1.Insert, 2.delete, 3.update处理1000条数据，对数据库资源(回滚段)使用率由高到低的顺序是().

A.123

B.132

C.231

D.312

1. 请看以下oracle中存储过程

begin

begin

<语句1>

exception

when others then

dbms\_output.put\_line('错误1');

end;

begin

<语句2>

dbms\_output.putline('错误2');

end;

exception

when others then

dbms\_output.put\_line("错误3");

end;

请问,当<语句1>,<语句2>出错时,该存储过程输出字符串分别是什么?( )

A.错误1,错误3

B.错误1,错误2

C.错误1,错误2,错误3

D.错误2,错误3

1. SQL Server数据库中获取系统时间的内部函数是:()

A.sysdate

B.getdate（）

C.currentdate

D.date()

10.在Oracle数据库中，请描述语句**SELECT** decode(sign(-1),1,'a','b') **FROM** dual返回结果：（）

A.1

B.-1

C.a

D.b

二、简答题

1. Oracle数据库中，如何增加表空间userdata的大小，请写出具体数据库语句？（以windows操作系统为例说明）
2. 简要叙述TRUNCATE, DROP, DELETE之间的关系。

3、举例说明while..loop/for..loop循环语句,如插入1..10的值到一个单字段数据表(结构自定义)。

4、解释data block , extent 和 segment的区别（这里建议用英文术语）

5、已知表T1中A字段含有重复的数据（A字段非空），请在空白处完成SQL语句，其中第一条语句查询T1标准的记录总数，第二条语句查询T1标准A字段不重复记录总数

三、编程题（10个空格，每个空格2分）：

1. 作家信息表Auths包含两个字段

|  |  |
| --- | --- |
| Author\_Code | VARCHAR2(50) |
| Salary | NUMBER(10) |

其中Author\_Code为作家代码，主键。Salary为作家工资额，非空。

现由于工资调整，对于作家代码在'A00001'和'A00006';之间的作家工资少于或等于五百，则给该作家增加15%的工资，对于大于500到1000之间则增加12%，其它则增加10%。如下程序是通过游标方式实现的，请填写程序中相应的空格。

DECLARE

-- 声明两个变量，用来接收游标返回的工资和作家代码

v\_Salary （）

v\_Code （）

v\_ErrText VARHCAR(200);

-- 声明游标c\_Salary，该游标的结果集是Auths表中作家代码(Author\_Code)值是

-- 'A00001'到'A00006'的工资值(Salary)和作家代码值(Author\_Code)

CURSOR c\_Salary IS

SELECT Salary,

Author\_Code

FROM Auths

WHERE Author\_Code >='A00001'

AND Author\_Code <='A00006';

BEGIN

-- 打开游标

LOOP

--游标循环，将游标查询结果集中的一行保存到两个临时变量中

（） c\_Salary INTO v\_Salary, v\_Code;

-- 当结果集中没有行是退出

EXIT WHEN （）;

-- 如果该作家的工作少于或等于五百，则给该作家增加15%的工资

-- 500到1000之间则增加12%,其他增加10%

IF v\_Salary <= 500 THEN

UPDATE Auths

SET Salary = （）

WHERE Author\_Code = v\_Code;

ELSIF v\_Salary <= 1000 THEN

UPDATE Auths

SET Salary = Salary + Salary \* 0.12

WHERE Author\_Code = v\_Code;

（）

UPDATE Auths

SET Salary = Salary + Salary \* 0.1

WHERE Author\_Code = v\_Code;

END IF;

END LOOP;

（）;

（）;

EXCEPTION

WHEN OTHERS THEN

CLOSE c\_Salary;

ROLLBACK;

v\_ErrText := （）;

DBMS\_OUTPUT.PUT\_LINE('程序异常终止，出现一下错误:'||v\_ErrText);

END;

1. 填空题：
   1. 类的三大特性是（）（）（）。
   2. 在SELECT语句的FROM子句中最多可以指定（）个表或视图，相互之间要用（）分隔，当所查询的表不在当前数据库时，可用（）格式来指出表或视图对象。
   3. 创建存储过程的命令是（）删除表的命令是（）。
   4. 计算字段的累加和的函数是：（），统计项目数的函数是：（**）**。
   5. 关系数据库中，主键是（）。
   6. 声明抽象类的关键字是（），抽象类不能（）。
   7. .NET运行库支持被称为（）的引用类型，其作用类似于C++中函数指针的用途。
   8. XML的全称是（）。
   9. C#中所有的类型实质上都是从（）类派生而来的。
   10. 声明静态变量的关键字是（），声明常量的关键字是（）。
   11. C#中处理内存管理功能的是（）。
   12. 在C#中用于显式地将一个值或引用转换为另一种不同的类型称为（）。
   13. 在ADO。NET中，command对象是用（）（）（）方法执行命令。
   14. B/S通常使用的结构设计模型分为（）（）（）。
   15. SQL SERVER中索引类型包括的三种类型分别是（）（）（）。
2. 简答题：

1．什么是中间语言（IL）？它的作用？

2．什么是装箱和拆箱？

3．出ADO。NET中读写数据库的主要的几个类？它们的作用？

1. C#中接口和类有什么异同？

5. 解释ASP。NET中的code-behind?

1. 什么是ASP。NET中的用户控件？
2. 在C＃中，string str = null 与 string str = “” 请尽量说明其中的区别。

8. 什么是受管制代码？

9 C#中常用的访问形式？它们各自的作用域？

10. SQL SERVER 中的存储过程的理解？  
11.CTS，CLS，CLR分别作何解释？

12. ASP。NET和ASP的主要优势？

13. NET的错误处理机制是什么？

14. 解释Page.IsPostBack属性？

15．.net framework包含哪几种技术？

16．触发器分为事前触发和事后触发，有何区别？

17．解释SQL SERVER2000中HAVING的用法？

18．解释SQL SERVER2000中GROUP BY的用法？

19．解释下C#中的方法的重载？

20．事务是什么？

21．描述下ASP。NET中使用DataReader对象 和DataSet对象的区别？

22．分析类和结构的异同？

23．解释ADO。NET中的使用Connection 类的CLOSE（）和DISPOSE（）的区别？

24．叙述ADO。NET中的DataAdapte类的作用？

25．什么是WEBSERVICE ？

26．**行下面代码后：**

String strTemp ="abcdefg 某某某"；

Int i System.Text.Encoding.Default.GetBytes(strTemp).Length;

Int j = strTemp.Length;

问：i=( ) ；j=( )

原因是什么？

27．．叙述ASP。NET中的几种验证控件(equiredFieldValidato, CompareValidator, RangeValidator, RegularExpressionValitor, CustomValidator, ValidationSummary)及作用？

1. 编程题：

1．有三个表，结构如下：

STUDENT（学生证号，姓名，性别，出生年月）

COURSE（课程号，课程名，授课教师）

SC（学生证号，课程号，成绩，授课教师）

用标准SQL 语句完成下列操作

1. 查询女生的基本情况，并按照学生证号降序排列
2. 查询成绩在90分以上的学生的姓名
3. 查询数学类课程的课程号和课程名（模糊查询，以S开头）
4. 查询选课学生所选课程的课程号和成绩，以及该学生的姓名和性别
5. 查询03号课程的平均成绩
6. 删除成绩不及格的学生姓名
7. 把学生证号为90876的学生所选的01号课程成绩改为80分

2．SQLSERVER服务器中，给定表 table1 中有两个字段 ID、LastUpdateDate，ID表示更新的事务号， LastUpdateDate表示更新时的服务器时间，请使用一句SQL语句获得最后更新的事务号。(10)

1. 填空题

1．.NET Framework包括（）（）（）三种技术

2．运算符++X表示（），X++表示（）。

3．软件测试方法分为（）（）。

4．.NET通过对各语言先编译成（ ），然后再执行时用（ ）将其编译成本地平台代码，来实现异构平台下对象的互操作

5．在SQL SERVER 中游标的声明的语句是（）释放的语句是（）。

6．SQL SERVER的编程语言是（）语言

7．委托声明的关键字是（）。

8．在MS SQL Server中，用来显示数据库信息的系统存储过程是**（）**

9．SQL语言中，用于事务回滚的语句是（**）**。

10．如果两个实体之间具有M：N联系，则将它们转换为关系模型的结果是（）个表。

11．Codebehind主要是通过把（）和（）放在不同的文件中来实现代码分离的。

12．在C#中由值类型转换为引用类型称为（），有引用类型转换为值类型称为（）

13．SQL SERVER中索引类型包括的三种类型分别是（）（）（）。

14．软件工程的三大文档（）（）（）。

15.在SQL SERVER2000中的四中约束（）（）（）（）。

二．简答题

1．什么是中间语言（IL）？它的作用？

2．C#中与基类通信的关键字什么？作用是什么？

3．什么是元数据？

4．ASP。NET和ASP的主要优势？

5．Override与重载有什么区别？

6．SQL SERVER 中的触发器的理解？

7． ref与out有什么区别？

8．大概描述下ASP。NET中服务器控件的生命周期？

9．什么是强类型系统？

10．触发器分为事前触发和事后触发，有何区别？

11．谈谈final, finally, finalize的区别。

12．在.net（C# or vb.net）中，Appplication.Exit 还是 Form.Close有什么不同？

13．C#中的委托是什么？事件是不是一种委托？

14．值类型和引用类型的区别？

15．解释下C#中的方法的重载？

16．分析类和结构的异同？

17． &和&&的区别。

18．描述怎样区分ExcuteNonQuery 和ExcuteScalar方法？

19．解释文件系统为什么会存在数据冗余？这种数据冗余会导致哪两类典型的问题？

20．什么是WEBSERVICE ？

21．接口和抽象类有什么区别？

22．描述类中的构造函数和析构函数？

23．解释下C#中的程序集？

24．try {}里有一个return语句，那么紧跟在这个try后的finally {}里的code会不会被执行，什么时候被执行，在return前还是后?

25．SQL SERVER的两种索引是何形式？索引的作用？索引的优缺点？

26．谈谈C/S和B/S的特点

三．编程题：

1．设有如下关系表

供应者 SUPPLIER（SNO，SNAME，CITY） SNO供应者编号，SNAME为供应者姓名，CITY所在城市

零件 PART（PNO，PNAME，WEIGHT）PNO零件号，PNAME零件名称，WEIGHT重量

工程 JOB（JNO，JNAME，CITY） JNO工程号，JNAME工程名，CITY所在城市

联系关系 SPJ（SNO，PNO，JNO，QTY） QTY为数量

1．查找给工程J1提供零件P1的供应者号SNO

2．查找在北京的供应者给武汉的工程提供零件的零件号

3．查找由供应者S1提供的零件名PNAME

4．查找CITY值为上海的工程号和名称

5，将工程J3的城市改为广州

6．将所有重20公斤的零件改为重10公斤

7．将给工程J1提供零件P1的供应者S1改为S2

8．将值（S3，凌涛，武汉）加到SUPPLIER中

9．删除所有上海工程的数据

2．有个表USERS如下  
Number(int) Name(char)  
 1 a  
 2 b  
 3 b  
 4 a  
 5 c  
 6 c  
要求:当Name列上有相同时,只保留Number这列上值小的那一行,结果应如下:  
Number(int) Name(char)  
1 a  
2 b  
5 c  
问:用SQL语句的Delete来实现.(记住,是一个SQL语句)

**C#试题(3)**

1. 填空题
   1. 类的三大特性是（）（）（）。
   2. SQL语言中，用于事务回滚的语句是（**）**。

3． 声明静态变量的关键字是（），声明常量的关键字是（）。

4． 软件工程的三大文档（）（）（）。

5． 为数据表创建索引的目的是（）。

6． B/S通常使用的结构设计模型分为（）（）（）。

7． SQL SERVER中索引类型包括的三种类型分别是（）（）（）。

8． 在SQL SERVER2000中的四中约束（）（）（）（）。

9． Codebehind主要是通过把（）和（）放在不同的文件中来实现代码分离的。

10．计算字段的平均值和最大值的函数是：（）（）

11．XML的全称是（），提供了快速和有效地读写XML的方式的类分别是（）（）。

12. SQL SERVER 中的存储过程是（）。

13．C#中所有的类型实质上都是从（）类派生而来的。

14．.NET运行库支持被称为（ ）的引用类型，其作用类似于C++中函数指针的用途。

15. 在流程控制中CONTINUE的作用（）

1. 简答题
2. 描述下C#中的程序集？程序集是有哪两部分组成？
3. 什么是元数据？
4. 请写出 BOOL flag 与“零值”比较的 if 语句？请写出 char p 与“零值”比较的 if 语句？
5. 说出下面几个函数的区别:   
   private void test(string str){…}   
   private void test(ref string str){…}   
   private void test(out string str){…}
6. 什么是强类型系统？
7. 值类型和引用类型的区别？
8. 如何理解委托？

8． 事务是什么？

9． 解释重载函数、虚拟函数的概念？

10．请简述以下两个for循环的优缺点

|  |  |
| --- | --- |
| for (i=0; i<N; i++)  {  if (condition)  DoSomething();  else  DoOtherthing();  } | if (condition)  {  for (i=0; i<N; i++)  DoSomething();  }  else  {  for (i=0; i<N; i++)  DoOtherthing();  } |

11．描述描述怎样区分ExcuteNonQuery ，ExcuteDataReader和ExcuteScalar方法？

12. 列出ADO。NET中读写数据库的主要的几个类？它们的作用？

13. 什么是中间语言（IL）？它的作用？

14. 在SQL SERVER2000中存储过程和触发器的区别？

15．在ASP.NET中页面之间传递值的方式？(尽可能叙述)

16．分析类和结构的异同？

17．short s1 = 1; s1 = s1 + 1;有什么错? short s1 = 1; s1 += 1;有什么错?

18．叙述ASP。NET中的equiredFieldValidator和RegularExpressionValitor验证控件及作用？

19．叙述下Using 的用法？

20．C#中接口和类有什么异同？

21．什么是装箱和拆箱？

22．什么是WEBSERVICE ？

23．软件开发过程一般有几个阶段？

24．在ASP。NET中自定义控件是什么？

25．ASP.NET的Application、Session、Cookie、ViewState和Cache等变量的区别是什么？

26．解释下在ASP.NET中Response的作用?

27．解释下在ASP.NET中Request的作用?

28. C#中能够使用指针吗?如果不能使用说明理由,如果能使用说出使用方法?

三．编程题：

1．为管理岗位业务培训信息，建立3个表:

　　 S (S#,SN,SD,SA) S#,SN,SD,SA 分别代表学号、学员姓名、所属单位、学员年龄

　　 C (C#，CN) C#,CN 分别代表课程编号、课程名称

　　 SC ( S#,C#,G ) S#,C#,G 分别代表学号、所选修的课程编号、学习成绩

　　 1. 使用标准SQL嵌套语句查询选修课程名称为’税收基础’的学员学号和姓名

　　 2. 使用标准SQL嵌套语句查询选修课程编号为’C2’的学员姓名和所属单位

　 　3. 使用标准SQL嵌套语句查询不选修课程编号为’C5’的学员姓名和所属单位

　　 4. 使用标准SQL嵌套语句查询选修全部课程的学员姓名和所属单位

　　 5. 查询选修了课程的学员人数

　　 6. 查询选修课程超过5门的学员学号和所属单位

2．SQLSERVER服务器中，给定表 table1 中有两个字段 ID、LastUpdateDate，ID表示更新的事务号， LastUpdateDate表示更新时的服务器时间，请使用一句SQL语句获得最后更新的事务号。

第一题:完成下面的分段函数的计算,程序运行时用户从键盘输入一个数值X,在屏幕上输出Y值.   
Y= (这里有个大括号) X的2次方 X>0   
0 X=0   
-X的2次方 X<0   
2.通过键盘输入一串字符,统计并输出其中英文字母的个数.   
3.已有按从小到大的顺序排列的10个数,现输入一个数,要求插入到数列中.插入后数列仍然按从小到大的顺序排列.

第一题   
main( ){   
int x,y;   
scanf("%d",&x);   
if(x>0)   
y=x\*x;   
else if(x==O)   
y=0;   
else y=(-x)\*(-x);   
printf("%d",y);} 第二题   
#include "string.h"   
main( )   
{char str[30];   
int i,k=0;   
scanf("%s",str);   
for(i=0; str[i]!='\0';i++)   
if(str[i]>=a&&str[i]<=z||str[i]>=A&&str[i]<=Z)   
k++;   
prinf("%d",k);}

第三题：   
#include<stdio.h>   
main()   
{ int a[11]={1，4，6，9，13，16，19，28，40，100}；   
int temp1,temp2,number,end,i,j;   
printf("原数字串为：\n");   
for(i=0;i<10;i++)   
printf("%5d",a[i]);   
printf("\n");   
printf("insert data:");   
scanf("%d",&number);   
end=a[9];   
if(number>end)   
a[10]=number;   
else   
{ for(i=o;i<10;i++)   
{ if(a[i]>number)   
{temp1=a[i]; a[i]=number;   
for(j=i+1;j<11;j++)   
{ temp2=a[j];   
a[j]=temp;   
temp1=temp2; }   
break; }}}   
printf("插入后的数字串为：\n");   
for(i=0;i<11;i++)   
printf("%6d",a[i]); }

第三题   
main()   
{int h,i,j,k,l,m,a[11];   
clrscr();   
printf("intput 10 number with the regular(min....max):\n");   
for(h=0;h<=9;h++)   
{ scanf("%d",&a[h]);}   
printf("please input one number:\n");   
scanf("%d",&k);   
for(i=0;i<=9;i++) {   
if(k<a[i])   
{for(j=i;j<=9;j++)   
{a[10-j+i]=a[9-j+i]; }   
a[i]=k; break; }   
else   
a[10]=k; }   
for(m=0;m<=10;m++)   
printf("%5d",a[m]);   
getch();}

1、C语言程序的基本单位是\_\_\_\_ A) 程序行 B) 语句 C) 函数 D) 字符、C、1  
2、C语言程序的三种基本结构是\_\_\_\_ A、顺序结构，选择结构，循环结构 B、递归结构，循环结构，转移结构 C、嵌套结构，递归结构，顺序结构 D、循环结构，转移结构，顺序结构、A、1  
3、C语言规定，程序中各函数之间 A) 既允许直接递归调用也允许间接递归调用 B) 不允许直接递归调用也不允许间接递归调用 C) 允许直接递归调用不允许间接递归调用 D) 不允许直接递归调用允许间接递归调用、A、1  
4、C语言中可处理的文件类型是（ ） A) 文本文件和数据文件 B)文本文件和二进制文件 C) 数据文件和二进制文件 D)数据代码文件、B、1  
5、C语言可执行程序的开始执行点是（ ） A) 程序中第一条可执行语句 B) 程序中第一个函数 C) 程序中的main函数 D) 包含文件中的第一个函数、C、1  
6、C语言提供的合法的数据类型关键字是 A)double B) short C) integer D) char、B、1  
7、C语言中,运算对象必须是整型数的运算符是 A) % B) \ C) %和\ D) \* \*、A、1  
8、C语言中函数返回值的类型是由（ ）决定。 A) return语句中的表达式类型 B) 调用函数的主调函数类型 C) 调用函数时临时 D) 定义函数时所指定的函数类型、D、1  
9、C语言中数组名作为参数传递给函数，作为实在参数的数组名被处理为\_\_\_\_\_。 A、 该数组的长度。 B、 该数组的元素个数。 C、 该数组中各元素的值。 D、 该数组的首地址。、D、1  
10、C语言中数组下标的下限是\_\_\_\_\_\_\_\_。 A、1 B、0 C、视具体情况 D、无固定下限、B、1  
11、C语言中提供的合法关键字是\_\_\_\_ A、swith B、cher C、case D、default、D、1  
12、C语言中文件的存取方式是\_\_\_\_\_\_\_\_。 A、顺序存取 B、随机存取 C、顺序存取、随机存取均可 D、顺序存取、随机存取均不可、C、1  
13、C语言中最简单的数据类型包括 A) 整型、实型、逻辑型 B) 整型、实型、字符型 C) 整型、字符型、逻辑型 D) 整型、实型、逻辑型、字符型、B、1  
14、sizeof(float)是（ ） A) 一种函数调用 B) 一个不合法的表示形式 C) 一个整型表达式 D) 一个浮点表达式、C、1  
15、x、y、z被定义为int型变量，若从键盘给x、y、z输入数据，正确的输入语句是 A) input x,y,z; B) scanf("%d%d%d",&x,&y,&z); C) scanf("%d%d%d",x,y,z); D) read("%d%d%d",&x,&y,&z);、B、1  
16、表达式:10!=9的值是 A) true B) 非零值 C) 0 D) 1、D、1  
17、表示关系x<=y<=z的c语言表达式为 A) (X<=Y)&&(Y<=Z) B) (X<=Y)AND(Y<=Z) C) (X<=Y<=Z) D) (X<=Y)&(Y<=Z)、A、1  
18、程序片段：在TC20中, int i=65536; printf("%d",i);的输出结果是\_\_\_\_ A) 65536 B) 0 C) 有语法错误，无输出结果 D) -1、B、1  
19、当调用函数时，实参是一个数组名，则向函数传送的是\_\_\_\_ A) 数组的长度 B) 数组的首地址 C) 数组每一个元素的地址 D) 数组每个元素中的值、B、1  
20、对嵌套子程序调用说法正确的是\_\_\_\_。 A、 外层子程序可以调用所有的内层子程序 B、 内层了程序只可以调用包含本身的外层子程序，不可以隔层调用 C、 外分程序必须能完全套住内分程序 D、 以上说法均不正确、C、1  
21、对于基类型相同的两个指针变量之间，不能进行的运算是 A) < B) = C) + D) -、C、1  
22、合法的C语言中，合法的长整型常数是 A) '\t' B) "A" C) 65 D) A、A、1  
23、假定int类型变量占用两个字节，其有定义：int x[10]={0,2,4};，则数组x在内存中所占字节数是 A) 3 B) 6 C) 10 D) 20、D、1  
24、假定有以下变量定义: int k=7 ,x=12; 则能使值为3的表达式是 A x%=(k%=5) B x%=(k-k%5) C x%=k-k%5 D (x%=k)-(k%=5)、D、1  
25、请读程序片段(字符串内没有空格): printf("%d",strlen("ATS012\1\\"));的输出结果是 A) 11 B) 10 C) 9 D) 8 、C、1  
26、请选出合法的C语言赋值语句 A) a=b=58 B) i++; C) a=58,b=58 D) k=int(a+b);、B、1  
27、若a为int类型，且其值为3，则执行完表达式a+=a-=a\*a后，a的值是 A) -3 B) 9 C) -12 D) 6、C、1  
28、若int类型占两个字节，则以下语句输出为\_\_\_\_\_\_\_\_。 int k=-1; printf("%d,u%",k,k); A、-1,-1 B、-1,65536 C、-1,32768 D、-1,65535、D、1  
29、若变量a是int类型，并执行了语句：a=‘A'+1、6；，则正确的叙述是 A) a的值是字符C B) a的值是浮点型 C) 不允许字符型和浮点型相加 D) a的值是字符‘A’的ASCII值加上1。、D、1  
30、若变量已正确说明为float类型，要通过语句scanf("%f %f %f ",&a,&b,&c);给a赋于10、0，b赋予22、0，c赋予33、0，不正确的输入形式是： A) 10 B) 10、0,22、0,33、0 22 33 C) 10、0 D) 10 22 22、0 33、0 33、B、1  
31、若要求在if后一对圆括号中表示a不等于0的关系,则能正确表示这一关系的表达式为 A) a<>0 B) !a C) a=0 D) a、B、1  
32、若已定义x和y为double类型，则表达式:x=1，y=x+3/2的值是 A)1 B)2 C)2、0 D)2、5、C、1  
33、若有定义：int a=8，b=5，c；，执行语句c=a/b+0、4;后，c的值为 A) 1、4 B) 1 C) 2、0 D) 2、B、1  
34、若有定义:int x,y;char a,b,c;并有以下输入数据(此处< CR> 代表换行符，/u代表空格): 1u2 AuBuC 则能给x赋整数1，给y赋整数2，给a赋字符A，给b赋字符B，给c赋字符C 的正确程序段是 A) scanf("x=%d y+%d",&x,&y);a=getchar();b=getchar();c=getchar(); B) scanf("%d %d",&x,&y);a=getchar();b=getchar();c=getchar(); C) scanf("%d%d%c%c%c,&x,&y,&a,&b,&c); D) scanf("%d%d%c%c%c%c%c%c:&x,&y,&a,&a,&b,&b,&c,&c);、D、1  
35、若有定义和语句: char s[10]:s="abcd";printf("%s",s); 则结果是(以下u代表空格) A) 输出abcd B) 输出a C) 输出abcduuuuu D) 编译不通过、D、1  
36、若有以下程序段， int c1=1,c2=2,c3; c3=1、0/c2\*c1; 则执行后，c3中的值是 A) 0 B) 0、5 C) 1 D) 2、A、1  
37、若有以下定义: char a; int b; float c; double d; 则表达式a\*b+d-c值的类型为 A) float B) int C) char D) double、D、1  
38、若有以下定义和语句 char c1='b',c2='e'; printf("%d,%c",c2-c1,c2-'a'+"A"); 则输出结果是: A)2,M B)3,E C)2,E D)输出项与对应的格式控制不一致，输出结果不确定、B、1  
39、若有以下函数调用语句： fun(a+b,(x,y),fun(n+k,d,(a,b)));在此函数调用语句中实参的个数是 A）3 B）4 C）5 D）6、A、1  
40、若执行下面的程序时从键盘上输入3和4，则输出是\_\_\_\_ A) 14 B) 16 C) 18 D) 20 main( ) { int a,b,s; scanf("%d %d",&a,&b); s=a; if(a、B、1  
41、若执行下面的程序时从键盘上输入5，则输出是 A) 7 B) 6 C) 5 D) 4 main() { int x; scanf("%d",&x); if(x++>5) printf("%d",x); else printf("%d",x--);}、B、1  
42、若执行以下程序时从键盘上输入9，则输出结果是 A) 11 B) 10 C) 9 D) 8 main( ) { int n; scanf("%d",&n); if(n++、B、1  
43、设 a、b、c、d、m、n均为 int型变量，且 a=5、b=6、c=7、d=8、m=2、n=2，则逻辑表达式 (m=a>b)&&(n=c>d)运算后，n的值为\_\_\_\_ A) 0 B) 1 C) 2 D) 3、C、1  
44、设 int a=12，则执行完语句 a+=a-=a\*a后，a的值是 A) 552 B) 264 C) 144 D) -264、D、1  
45、设 int b=2;表达式(b>>2)/(b>>1)的值是 A) 0 B) 2 C) 4 D) 8、A、1  
46、设a,b和c都是int型变量，且a=3,b=4,c=5,则下列表达式中，值为0的表达式是\_\_\_\_\_\_\_\_ A)'a'&&'b' B)a<=b C)a||b+c&&b-c D)!((a<b)&&!c||1)、D、1  
47、设x、y、z和k都是int型变量,则执行表达式:x=(y=4,z=16,k=32)后,x的值为 A 4 B 16 C 32 D 52、C、1  
48、设x=3,y=-4,z=6，写出表达式的结果。 !(x>y)+(y!=z)||(x+y)&&(y-z) A、0 B、1 C、-1 D、6、B、1  
49、设x和y均为int 型变量,则以下语句:x+=y;y=x-y;y=x-y;x-=y;的功能是 A、把x 和y按从大到小排列 B、把x和y按从小到大排列 C、无确定结果 D、交换x和y中的值、D、1  
50、设有 int x=11; 则表达式 (x++ \* 1/3) 的值是 A) 3 B) 4 C) 11 D) 12、A、1

武汉北大高科开发人员测试题

感谢参加武汉北大高科软件有限公司程序编写能力考试，请按照本文档要求编写相关的应用程序，并在５个工作日内将程序源代码、编译结果、相关文档（如果有）发送到 [hr@pkusoft.net](mailto:hr@pkusoft.net)（邓小姐）收。

|  |  |  |  |
| --- | --- | --- | --- |
| 试题编码 | PKUHRTS012 | 测试内容 | 页面开发人员 |
| 标准时间 | ５个工作日 |  |  |

## 基本要求

1. 可以选择ASP或者JSP其中的一种实现程序功能
2. 客户端脚本要求使用JAVASCRIPT
3. 本测试将同时考察界面编写能力和用户交互设计能力，请在编写程序时考虑此方面

## 功能要求

要求通过分别编写一段客户端代码和服务端代码实现如下功能：

1. 正常解析一段XML代码（附后）
2. 将XML的内容显示在页面上

## XML结构

|  |
| --- |
| <?xml version=”1.0”?>  <PKUWH version=”2.0”>  <QUERYINFO>  <ROW>  <DEPT>  <DEPTNAME>行政部</DEPTNAME>  <ROW>  <NAME>张平</NAME>  <SEX>男</SEX>  <BIRTHDAY>1974-03-02</BIRTHDAY>  </ROW>  </DEPT>  </ROW>  <ROW>  <DEPT>  <DEPTNAME>开发部</DEPTNAME>  <ROW>  <NAME>李小辉</NAME>  <SEX>男</SEX>  <BIRTHDAY>1976-04-06</BIRTHDAY>  </ROW>  <ROW>  <NAME>张明</NAME>  <SEX>男</SEX>  <BIRTHDAY>1981-07-08</BIRTHDAY>  </ROW>  </DEPT>  </ROW>  </QUERYINFO>  </PKUWH> |

## 显示结果示例

|  |  |  |  |
| --- | --- | --- | --- |
| 部门 | 姓名 | 性别 | 出生日期 |
| 行政部 | 张平 | 男 | 1974-03-02 |
| 开发部 | 李小辉 | 男 | 1976-04-06 |
| 张明 | 男 | 1981-07-08 |

说明：

该显示结构仅为示例，具体的页面显示方案请自行确定。

武汉北大高科软件有限公司

1. 有一个正方提的盒子，里面有些水，怎么样判断水是过一半还是少一半

答：再装同样多的水，看是溢出还是没有溢出。

1. 面向对象的特点

答：继承，多态，封装

1. C#中的验证控件有几种

答：

* 1. RequiredFieldValidator（必须字段验证）；
  2. CompareValidator（比较验证）；
  3. RangeValidator （范围验证）；
  4. RegularExpresionValidator（正则表达式）；
  5. ValidationSummary（验证总结）；
  6. CustomValidator（自定义验证）

1. 页面跳转的几个方式

答：post，get

1. 页面间的传值有哪几种方式

答：session(viewstate) 简单，但易丢失

application 全局

cookie 简单，但可能不支持，可能被伪造

input ttype="hidden" 简单，可能被伪造

url参数 简单，显示于地址栏，长度有限

数据库 稳定，安全，但性能相对弱

武汉世纪畅想数字传播有限公司 .NET笔试题 2006.12.9

公司在武昌雄楚大街图书出版城C座908室

1. 填空：
2. 面向对象的语言有\_\_\_\_\_\_\_特性、\_\_\_\_\_\_\_特性、\_\_\_\_\_\_\_特性。

答：封装，继承，多态

1. ADO.NET有哪五大对象：

答：Connection，Command，DataSet，DataReader，DataAdapter

1. 能用foreach遍历访问的对象需要实现（ ）接口或声明（ ）方法的类型。

答：IEnumerable，GetEnumerable

2．多选：

1. 正确的是：BC

A．接口中可以有虚方法 B.一个类可实现多个接口

C.接口不能被实例化 D.接口中可以包含已实现的方法。

1. 读取数据库里的内容会用到以下那些：BCD

A Execute NonQuery B Execute Scalor

C fill D EXECUTE READER

3．简答：

private public internal protected 权限各自的调用方法。

答：public：不受限制的访问。

private：限定此类内部使用。

potected：除了类本身可以用外，任何继承此类的类都可以使用。

internal：表示在同一个应用程序（application）或类库（library）中都能使用。

4．写SQL语句

在SQL里取出A表中第31—40条记录。（SQLServer,以自动增长的ID做主键，注意：ID可能不是 连续的。）

答：select top 10 from a where id not in（select top 30 from a）

或 select top 10 from a where id > (select max(id) from (select top 30 from a) as b)

5、列举ASP.NET页面之间传递值的几种方式？

答：session(viewstate) 简单，但易丢失

application 全局

cookie 简单，但可能不支持，可能被伪造

input type="hidden" 简单，可能被伪造

url参数 简单，显示于地址栏，长度有限

数据库 稳定，安全，但性能相对弱

6、 看程序写结果

Class class1

{

private string str = “class1.str”;

private int i = 0 ;

static void stringconvert　(string str )

｛

　　str = “string being converted”;

｝

static void stringconvert　(class1 c )

{

c. str = “string being converted”;

}

static void Add (int i)

{

i++;

}

static void AddwithRef (ref int i)

{

i++;

}

static void Main()

{

int i1 = 10;

int i2 = 20;

　sting str ＝ “str”;

classｃ＝new class1();

Add(i1);

AddwithRef(ref i2）；

Add(c.i）；

stringconvert(str)；

stringconvert(c)；

console．WriteLine(i1)；

console．WriteLine(i2)；

console．WriteLine(c.i)；

console．WriteLine(str)；

console．WriteLine(c.str)；

}

}

答：10，21，0，“str”，“string being converted”

7看程序写结果

public abstract class A

{

public A ()

{

console.writeline('A');

}

public virtual void Fun()

{

console.writeline("A.Fun()" );

}

}

public class B : A

{

public B ()

{

console.writeline('B');

}

public new void Fun()

{

console.writeline(" B.Fun()" );

}

public static void Main()

{

A a = new B();

a.Fun();

}

}

答：’A’,’B’,A.fun()

8看程序写结果

public class A

{

public virtual void Fun1( int i)

{

console.writeline(i);

}

public void Fun2 ( A a )

{

a.Fun1(1);

Fun1(5);

}

}

public class B : A

{

public override void Fun1(int i)

{

base.Fun1( i+1 );

}

public static void Main()

{

B b = new B();

A a = new A();

a.Fun2(b);

b.Fun2(a);

}

}

答：2，5，1，6

9 C#设计程序： 有这样规则的几个数： 1,1,2,3,5,8,13,21,34,55...... 用递归的方法算出第30数是多少？

参考答案：

public static int add(int i)

{

if(i<=0)

{

return 0;

}

else if(i == 1 || i ==2)

{

return 1;

}

else

return add(i-1)+add(i-2);

}

10 C#设计程序：

猫大叫一声。所有的老鼠都开始逃跑，主人被惊醒。

要求 1 要有联动性，老鼠和主人的行为是被动的。

2 考虑可扩展性，猫的叫声可能引起其它联动效应。

设计方法一：

public interface Observer

{

void Response(); //观察者的响应，如是老鼠见到猫的反映

}

public interface Subject

{

void AimAt(Observer obs); //针对哪些观察者，这里指猫的要扑捉的对象---老鼠

}

public class Mouse : Observer

{

private string name;

public Mouse(string name, Subject subj)

{

this.name = name;

subj.AimAt(this);

}

public void Response()

{

Console.WriteLine(name + “attempt to escape!”);

}

}

public class Master : Observer

{

public Master(Subject subj)

{

subj.AimAt(this);

}

public void Response()

{

Console.WriteLine(“Host waken!”);

}

}

public class Cat : Subject

{

private ArrayList observers;

public Cat()

{

this.observers = new ArrayList();

}

public void AimAt(Observer obs)

{

this.observers.Add(obs);

}

public void Cry()

{

Console.WriteLine(“Cat cryed!”);

foreach (Observer obs in this.observers)

{

obs.Response();

}

}

}

class MainClass

{

static void Main(string[] args)

{

Cat cat = new Cat();

Mouse mouse1 = new Mouse(mouse1, cat);

Mouse mouse2 = new Mouse(mouse2, cat);

Master master = new Master(cat);

cat.Cry();

}

}

设计方法二: 使用event -- delegate设计..

public delegate void SubEventHandler();

public abstract class Subject

{

public event SubEventHandler SubEvent;

protected void FireAway()

{

if (this.SubEvent != null)

this.SubEvent();

}

}

public class Cat : Subject

{

public void Cry()

{

Console.WriteLine(“cat cryed”);

this.FireAway();

}

}

public abstract class Observer

{

public Observer(Subject sub)

{

sub.SubEvent += new SubEventHandler(Response);

}

public abstract void Response();

}

public class Mouse : Observer

{

private string name;

public Mouse(string name, Subject sub) : base(sub)

{

this.name = name;

}

public override void Response()

{

Console.WriteLine(name +“attempt to escape!”);

}

}

public class Master : Observer

{

public Master(Subject sub) : base(sub){}

public override void Response()

{

Console.WriteLine(host waken);

}

}

class Class1

{

static void Main(string[] args)

{

Cat cat = new Cat();

Mouse mouse1 = new Mouse(mouse1, cat);

Mouse mouse2 = new Mouse(mouse2, cat);

Master master = new Master(cat);

cat.Cry();

}

}

**亿阳信通股份有限公司笔试题**

**C#部分：**

1．  面向对象的方法有哪些特性（3种）：

2．  1。1。2。3。5。8。13。21。34。55 用递归的方法算出第30数是多少？

3．  4种修饰符的访问权限 public private protected internal

郭炜：主要考查两个方面：

              1、面向对象思想的理解深度；

                  （一）foreach循环所用到的接口及方法；

                  （二）对C#中的类封闭性、继承性、多态性的使用及理解程度；

                  （三）对接口的使用和理解。

              2、.net开发中的两个常用的东东：

                   （一）ADO.NET常对象；

                   （二）ASP.NET不同页面如何传值。

1.数字排列 1，1，2，3，5，8，13，21，34......  
用递归的算法求出第30位的数字。（用c#）  
  
2.写逻辑关系：猫叫一声老鼠就会跑，人回醒  
（1）要考虑连动性  
（2）要考虑扩展性  
  
还有3个读程序写结果的题和4个数据库的题。

C#：

1．翻译题：

什么是蓝牙？（概念，发展）

2．问答题：

* 不用SESSION，COOKIES，APPLICATION 怎么向其他页面传数据
* 遍历将页面为TEXTBOX的控件付值为空
* 谈谈你做项目的感想。
* PRIVATE，PROTECT，PUBLIC ，INTNER的区别。
* SESSION和COOKIES的区别
* 用户控件和自定义控件的区别

3数据库

FUNCTION和DELETE的区别

不用自增怎么解决客户抢号问题（主键）

表和视图的区别。

DataReader与Dataset有什么区别？

向服务器发送请求有几种方式？

如果在一个B/S结构的系统中需要传递变量值，但是又不能使用Session、Cookie、Application，有几种方法进行处理？

请编程遍历页面上所有TextBox控件并给它赋值为string.Empty

一列数的规则如下: 1、1、2、3、5、8、13、21、34......

求第30位数是多少， 用递归算法实现。(C#语言)

**Oracle测试试题**

1. 解释FUNCTION，PROCEDURE和PACKAGE区别

答：function 和procedure是PL/SQL代码的集合，通常为了完成一个任务。procedure 不需要返回任何值而function将返回一个值在另一方面，Package是为了完成一个商业功能的一组function和procedure的集合。

1. 取某个序列的当前值的PL/SQL语句怎么写？

答：SELECT 序列名.CURRVAL FROM DUAL；

1. 说明ORACLE数据库实例与ORACLE用户的关系？

答：实例可以包含多个用户，一个用户只能在一个实例下

1. 创建数据库时，自动建立的tablespace名称？

答：SYSTEM tablespace

1. 创建用户时，需要赋予新用户什么权限才能使它连上数据库？

答：CONNECT

1. IMPORT和SQL\*LOADER这2个工具的不同点？

答：这两个ORACLE工具都是用来将数据导入数据库的。

区别是：IMPORT工具只能处理由另一个ORACLE工具EXPORT生成的数据。而SQL\*LOADER可以导入不同的ASCII格式的数据源。

7．解释冷备份和热备份的不同点以及各自的优点？

答：热备份针对归档模式的数据库，在数据库仍旧处于工作状态时进行备份。而冷备份指在数据库关闭后，进行备份，适用于所有模式的数据库。热备份的优点在于当备份时，数据库仍旧可以被使用并且可以将数据库恢复到任意一个时间点。冷备份的优点在于它的备份和恢复操作相当简单，并且由于冷备份的数据库可以工作在非归档模式下,数据库性能会比归档模式稍好。（因为不必将archive log写入硬盘）

8．比较truncate和delete命令？

答：两者都可以用来删除表中所有的记录。区别在于：truncate是DDL（data defining language数据定义语言），它移动HWK，不需要rollback segment（处理事务回滚操作）而Delete是DML（data manufacturing language数据操作语言）操作，需要rollback segment（处理事务回滚操作）且花费较长时间。

9．给出数据的相关约束类型？

答：主键约束，外键约束，非空约束，唯一约束，检查约束。

10．说明索引的类型与作用？

答：索引类型上分为聚集索引，非聚集索引其作用是加快查询速度。

11．解释归档和非归档模式之间的不同和它们各自的优缺点

答：归档模式是指你可以备份所有的数据库 transactions并恢复到任意一个时间点。非归档模式则相反，不能恢复到任意一个时间点。但是非归档模式可以带来数据库性能上的少许提高。

12．解释$ORACLE\_HOME和$ORACLE\_BASE的区别？

答：ORACLE\_BASE是oracle的根目录，ORACLE\_HOME是oracle产品的目录。

13．获取某个字符字段的最后3个字符的函数是什么？

答：select substr (字段,(length(字段)-3)) from 表

14．取当前系统时间点日期（不包括年月）的SQL写法是怎样的？

答：Select substr (to\_char(sysdate,’YYYYMMDDh24hh:MM:SS’),5) from dual;

15．返回大于等于N的最小整数值？

答：select ceil(N) from dual;

16．将根据字符查询转换后结果，规则为：’A’转换为’男’，’B’转换为’女’，其他字符转换为’未知’，请用一个SQL语句写出。

答：select decode(字符,’A’,’男’,’B’,’女’,’未知’) from dual;

17．如何搜索出前N条记录？

答：select \* from 表 where Rownum <= N;

18．如何搜索出第N～M条记录？

答：select \* from 表 where Rownum <= M

Minus

select \* from 表 where Rownum <= N;

19．有一个数据表（TEST），字段如下：

|  |  |
| --- | --- |
| ID | number |
| PARENT\_ID | number |
| NAME | Varchar(20) |

请使用PL/SQL来按父子层次关系查询出该表的所有数据

答：Select \* from test a, test b Where a.parent\_id = b.id;

20．怎样用SQL语句实现查找一列中的第N大值？

答：select \* from (select \* from 表 order by 列名 Desc) where Rownum <= N

Minus

select \* from (select \* from 表 order by 列名 Desc) where Rownum <= N-1;

**西安博达软件有限公司笔试试题**

（按下面要求手写代码）



任务：

1. 按界面所示进行数据录入
2. 输入信息后点击“提交”按钮，开始验证。
3. 性别、年龄、学历均为必填内容。
4. 年龄必须为数字、且不得小于18岁，大于58岁。
5. 自行建表，对表中的多条记录列表显示。
6. 在列表中可以删除某条记录
7. （可选）实现上下翻页功能。
8. 页面的操作方式自行设计（方便用户操作者优先）

时限：

两小时

1. String s = new String(‘xyz’); 请问产生几个String Object?
2. 软件工程可以分为几个阶段？请问各个阶段的作用是什么？
3. 值类型与引用类型的区别？用C#事例说明？
4. 编写一个冒泡排序的程序。
5. 举例说明你所熟悉几种设计模式，说明各种模式在何种情况下使用？

7 你如何理解委托，一般在什么情况下用?

8 你做过的C/S结构程序中你用的是几层架构，为什么要用，各层间的关系是什么？

9：请问此程序的输出结果：

using System;

using System.Collections.Generic;

using System.Text;

namespace ConsoleApplication1

{

public class A

{

public A()

{

Fun();

}

public virtual void Fun()

{

}

}

class B : A

{

int x = 1;

int y;

public B()

{

y = 1;

}

public override void Fun()

{

Console.WriteLine("x={0},y={1}", x, y);

}

}

class Program

{

static void Main(string[] args)

{

B a = new B();

Console.ReadKey();

}

}

}

10.

Student学生信息表

Id(primary key) name sex class

Sch科目表

ID(primary key) name

Score成绩表

ID Uid(foreign key references Student(Id)) Sid(foreign key references Sch (ID)) num

A求各个班的总人数

B求1班的男生和2班的女生的平均成绩

C求各个班“数据类型”课不及格的学生数量

应用存储过程写个交叉表。

**西安夏日科技有限公司笔试题**

1. 简答与选择题：
2. 简述抽象类与接口的异同点？
3. Map与HashMap的关系是什么？
4. error和exception有什么区别?
5. Jsdk1.5与jsdk5.0有什么区别与联系？
6. 什么是java中的垃圾回收机制？什么时候回调用垃圾回收机制？
7. System.out.print(“3”+1+”/0”)输出结果是什么？

Public classText{

 public static void main(String args[]){

int a=4;

int b=3+1;

if(a=b){

system.out.print(“yes”);

}else

{

System.out.print(“no”);

}

}

}选择输出结果：

A,yes

B,no

C，程序编译出错！

D,程序可正常运行

8．**请说出你所知道的线程同步的方法？**

9．**Overload和Override的区别？**

10．System.out.print(),与字节输出流，字符输出流有什么关系？

二．问答题；

1．详细描述oracle9i的数据库标识，表空间，用户，存储过程，函数，序列的作用？

2，简单描述spring框架体系结构？

3，描述spring+hibernate+struts三者合用的流程？

4．Ocp是什么？简单描述其规则？

5．描述http,tcp，以及邮件协议？

.net笔试题

姓名： 日期：

1. 填空: (1)面向对象的语言具有\_\_\_\_封装\_\_\_\_性、\_\_\_\_\_继承\_\_\_\_性、\_\_\_\_多态\_\_\_\_性。

(2)能用foreach遍历访问的对象需要实现 \_\_\_\_\_\_\_IEnumerable\_\_\_\_\_\_\_\_\_接口或声明\_\_\_\_\_\_\_GetEnumerable\_\_\_\_\_\_\_\_\_方法的类型。

(3)列举ADO.net中的五个主要对象\_\_\_\_\_\_\_Command\_\_\_\_\_\_\_\_、\_\_\_\_\_\_Connection\_\_\_\_\_\_\_、\_\_\_\_\_\_\_DataAdapter\_\_\_\_\_\_\_\_、\_\_\_\_\_\_\_DataSet\_\_\_\_\_\_\_\_、\_\_\_\_\_\_\_DataReader\_\_\_\_\_\_\_\_\_\_。

2. 不定项选择：

(1) 以下叙述正确的是：bc

A. 接口中可以有虚方法。 B. 一个类可以实现多个接口。

C. 接口不能被实例化。 D. 接口中可以包含已实现的方法。

(2) 从数据库读取记录，你可能用到的方法有：bcd

A. ExecuteNonQuery B. ExecuteScalar

C. Fill D. ExecuteReader

3. 简述 private、 protected、 public、 internal 修饰符的访问权限。

答：public 不受限制的访问

Private 只能在本类中访问

Protected 本类和继承与他的类中的都能访问

Internal 同一个应用程序或类库中

4. 写出一条Sql语句： 取出表A中第31到第40记录（SQLServer, 以自动增长的ID作为主键, 注意：ID可能不是连续的。）

答：select top 10 from a where id not in（select top 30 from a）

或：select top 10 from a where id > (select max(id) from (select top 30 from a) as b)

5 .列举ASP.NET 页面之间传递值的几种方式。

答：

session(viewstate) 简单，但易丢失

application 全局

cookie 简单，但可能不支持，可能被伪造

input type="hidden" 简单，可能被伪造

url参数 简单，显示于地址栏，长度有限

数据库 稳定，安全，但性能相对弱

6. 写出程序的输出结果

class Class1 {

private string str = "Class1.str";

private int i = 0;

static void StringConvert(string str) {

str = "string being converted.";

}

static void StringConvert(Class1 c) {

c.str = "string being converted.";

}

static void Add(int i) {

i++;

}

static void AddWithRef(ref int i) {

i++;

}

static void Main() {

int i1 = 10;

int i2 = 20;

string str = "str";

Class1 c = new Class1();

Add(i1);

AddWithRef(ref i2);

Add(c.i);

StringConvert(str);

StringConvert(c);

Console.WriteLine(i1);

Console.WriteLine(i2);

Console.WriteLine(c.i);

Console.WriteLine(str);

Console.WriteLine(c.str);

}

}

答：10，21，0，str，string being converted.

7.写出程序的输出结果

public abstract class A

{

public A()

{

Console.WriteLine('A');

}

public virtual void Fun()

{

Console.WriteLine("A.Fun()");

}

}

public class B: A

{

public B()

{

Console.WriteLine('B');

}

public new void Fun()

{

Console.WriteLine("B.Fun()");

}

public static void Main()

{

A a = new B();

a.Fun();

}

}

答：A，B，A.Fun()

8. 写出程序的输出结果：

public class A

{

public virtual void Fun1(int i)

{

Console.WriteLine(i);

}

public void Fun2(A a)

{

a.Fun1(1);

Fun1(5);

}

}

public class B : A

{

public override void Fun1(int i)

{

base.Fun1 (i + 1);

}

public static void Main()

{

B b = new B();

A a = new A();

a.Fun2(b);

b.Fun2(a);

}

}

答：2，5，1，6

9. 一列数的规则如下: 1、1、2、3、5、8、13、21、34......

求第30位数是多少， 用递归算法实现。(C#语言)

答：private int Add(int i)

{

if (i <= 0)

{

return 0;

}

else if (i == 1 || i == 2)

{

return 1;

}

else

{

return Add(i - 1) + Add(i - 2);

}

}

10. 程序设计: 猫大叫一声，所有的老鼠都开始逃跑，主人被惊醒。(C#语言)

要求： 1.要有联动性，老鼠和主人的行为是被动的。

2.考虑可扩展性，猫的叫声可能引起其他联动效应。

public interface Observer

{

void Response(); //观察者的响应，如是老鼠见到猫的反映

}

public interface Subject

{

void AimAt(Observer obs); //针对哪些观察者，这里指猫的要扑捉的对象---老鼠

}

public class Mouse : Observer

{

private string name;

public Mouse(string name, Subject subj)

{

this.name = name;

subj.AimAt(this);

}

public void Response()

{

Console.WriteLine(name + " attempt to escape!");

}

}

public class Master : Observer

{

public Master(Subject subj)

{

subj.AimAt(this);

}

public void Response()

{

Console.WriteLine("Host waken!");

}

}

public class Cat : Subject

{

private ArrayList observers;

public Cat()

{

this.observers = new ArrayList();

}

public void AimAt(Observer obs)

{

this.observers.Add(obs);

}

public void Cry()

{

Console.WriteLine("Cat cryed!");

foreach (Observer obs in this.observers)

{

obs.Response();

}

}

}

class MainClass

{

static void Main(string[] args)

{

Cat cat = new Cat();

Mouse mouse1 = new Mouse("mouse1", cat);

Mouse mouse2 = new Mouse("mouse2", cat);

Master master = new Master(cat);

cat.Cry();

}

}

//---------------------------------------------------------------------------------------------

设计方法二: 使用event -- delegate设计..

public delegate void SubEventHandler();

public abstract class Subject

{

public event SubEventHandler SubEvent;

protected void FireAway()

{

if (this.SubEvent != null)

this.SubEvent();

}

}

public class Cat : Subject

{

public void Cry()

{

Console.WriteLine("cat cryed.");

this.FireAway();

}

}

public abstract class Observer

{

public Observer(Subject sub)

{

sub.SubEvent += new SubEventHandler(Response);

}

public abstract void Response();

}

public class Mouse : Observer

{

private string name;

public Mouse(string name, Subject sub) : base(sub)

{

this.name = name;

}

public override void Response()

{

Console.WriteLine(name + " attempt to escape!");

}

}

public class Master : Observer

{

public Master(Subject sub) : base(sub){}

public override void Response()

{

Console.WriteLine("host waken");

}

}

class Class1

{

static void Main(string[] args)

{

Cat cat = new Cat();

Mouse mouse1 = new Mouse("mouse1", cat);

Mouse mouse2 = new Mouse("mouse2", cat);

Master master = new Master(cat);

cat.Cry();

}

}

1. C#中 property 与 attribute的区别，他们各有什么用处，这种机制的好处在哪里？

答：一个是属性，用于存取类的字段，一个是特性，用来标识类，方法等的附加性质

2. 讲一讲你理解的web service,在dot net framework中，怎么很好的结合xml?（讲概念就行了）

答：web服务，是利用SOAP（简单对象访问协议，Simple Object Access Protocol）在HTTP上执行远程方法调用的一种方法，也可以使用WSDL（Web Service Description Language，Web服务描述语言）来完整的描述Web服务，基于XML标准。

3. C#, Java 和 c++的特点，有什么相同的地方，不同的地方，C#分别从c++和java中吸取了他们那些优点？

答：都是面向对象的语言，C#和java都是建立在虚拟机上的。从java中吸取了跨语言。

4. C#可否对内存进行直接的操作？(这可是个难点哦？要注意！）

答：可以，只要标记为unsafe

5. 用Visual C++ 6.0编写的代码(unmanaged code)，如何在CLR下和其他dot net component结合？

答：被编译成dll

6. 以前作过的一些项目？介绍一下自己的编程经验。(这是蒙混不过去的，基本一问就知道是真是假)

7. 你对公司有什么期望？你的薪水期望值？为什么学计算机？以前的工作经验（这非常重要）？

Q3： 维护数据库的完整性、一致性、你喜欢用触发器还是自写业务逻辑？为什么

答：使用数据库关系:这样可以尽可能的规范编码、对比较大型的开发来说也容易维护数据的一致性和完整性，也让数据库的性能尽可能的发挥出来。

实在是不能使用数据库的关系的时候才使用触发器，这让数据逻辑简单，但这时候要注意记录文档，不然，触发器多了，维护难度会相当大。

最后是自己编写逻辑：这样会造成数据的一致性，完整性的维护逻辑相当大，也十分分散，没有良好的文档记录习惯的话，一旦任务出现交接，其他人很难继续你未完成的工作，甚至于过一段时间以后，自己要完全理请逻辑都是一个麻烦的事！因为这些逻辑相当于隐藏起来了的。

Q4：ADO。NET相对于ADO等主要有什么改进？

答：ADO以Recordset存储，而ADO.NET则以DataSet表示。Recordset看起来更像单表，如果让Recordset以多表的方式表示就必须在SQL中进行多表连接。反之，DataSet可以是多个表的集合。ADO 的运作是一种在线方式，这意味着不论是浏览或更新数据都必须是实时的。ADO.NET则使用离线方式，在访问数据的时候ADO.NET会利用XML制作数据的一份幅本，ADO.NET的数据库连接也只有在这段时间需要在线。由于ADO使用COM技术，这就要求所使用的数据类型必须符合COM规范，而ADO.NET基于XML格式，数据类型更为丰富并且不需要再做COM编排导致的数据类型转换，从而提高了整体性能。

Q5：ASP。NET与ASP相比，主要有哪些进步？

答：ASP.Net摆脱了以前ASP使用脚本语言来编程的缺点，理论上可以使用任何编程语言包括C++ , VB , JS等等，当然，最合适的编程语言还是MS为.Net Frmaework专门推出的C(读c sharp)，它可以看作是VC和Java的混合体吧，尽管MS自己讲C#内核中更多的象VC，但实际上我还是认为它和Java更象一些吧。首先它是面向对象的编程语言，而不是一种脚本，所以它具有面向对象编程语言的一切特性，比如封装性、继承性、多态性等等，这就解决了刚才谈到的ASP的那些弱点。封装性使得代码逻辑清晰，易于管理，并且应用到ASP.Net上就可以使业务逻辑和Html页面分离，这样无论页面原型如何改变，业务逻辑代码都不必做任何改动；继承性和多态性使得代码的可重用性大大提高，你可以通过继承已有的对象最大限度保护你以前的投资。并且C#和C++、Java一样提供了完善的调试/纠错体系。

Q6：C#中的委托是什么？事件是不是一种委托？

答：委托是一种安全的函数指针，事件是一种消息机制

Q7：描述一下C#中索引器的实现过程，是否只能根据数字进行索引？

Q8：C#中要使一个类支持FOREACH遍历，实现过程怎样？

答：实现接口IEnumerable

Q10：写一个HTML页面，实现以下功能，左键点击页面时显示“您好”，右键点击时显示“禁止右键”。并在2分钟后自动关闭页面。

Q11：你对XMLHTTP、WEBSERVICE 了解吗？简单描述其特点、作用

答：XMLHTTP可以主动获取远端web代码，类似HttpWebRequest

Q12：接口和抽象类有什么区别？你选择使用接口和抽象类的依据是什么？

答：接口用于规范，抽象类用于共性。接口中只能声明方法，属性，事件，索引器。而抽象类中可以有方法的实现，也可以定义非静态的类变量。抽象类是类，所以只能被单继承，但是接口却可以一次实现多个。抽象类可以提供某些方法的部分实现，接口不可以.抽象类的实例是它的子类给出的。接口的实例是实现接口的类给出的。再抽象类中加入一个方法，那么它的子类就同时有了这个方法。而在接口中加入新的方法，那么实现它的类就要重新编写（这就是为什么说接口是一个类的规范了）。接口成员被定义为公共的，但抽象类的成员也可以是私有的、受保护的、内部的或受保护的内部成员（其中受保护的内部成员只能在应用程序的代码或派生类中访问）。此外接口不能包含字段、构造函数、析构函数、静态成员或常量。

Q13：自定义控件和一般用户控件的异同？如果要用这两者之一，你会选择哪种？为什么

答：用户控件(UserControl): 扩展名为\*.ascx,跟\*.aspx在结构上相似，是指页面中

加载的功能块,只是用户控件不能单独作为页面运行,必须嵌入到\*.aspx页面或其它用

户控件中使用 。

自定义控件，跟HtmlControl或WebControl相似，编译后可以添加引用到工具栏里

面，直接用鼠标拖动使用。

看情况使用，如果功能比较多，会用自定义控件。

Q14：大概描述一下ASP。NET服务器控件的生命周期

答：（1）初始化----在此阶段中，主要完成两项工作：一、初始化在传入Web请求生命周期内所需的设置；二、跟踪视图状态。首先，页面框架通过默认方式引发Init事件，并调用OnInit()方法，控件开发人员可以重写该方法为控件提供初始化逻辑。此后，页面框架将调用TrackViewState方法来跟踪视图状态。需要注意的是：多数情况下，Control基类提供的TrackViewState方法实现已经足够了。只有在控件定义了复杂属性时，开发人员才可能需要重写TrackViewState方法。

　　（2）加载视图状态----此阶段的主要任务是检查服务器控件是否存在以及是否需要将其状态恢复到它在处理之前的请求结束的状态。因此该过程发生在页面回传过程中，而不是初始化请求过程。在此阶段，页面框架将自动恢复ViewState字典。如果服务器控件不维持其状态，或者它有能力通过默认方式保存其所有状态而使用ViewState字典，那么开发人员则不必实现任何逻辑。针对那些无法在ViewState字典中存储的数据类型或者需要自定义状态管理的情况，开发人员可以通过重写LoadViewState方法来自定义状态的恢复和管理。

（3）处理回发数据----若要使控件能够检查客户端发回的窗体数据，那么必须实现System.Web.UI.IPostBackDataHandler接口的　LoadPostData()方法。因此只有处理回发数据的控件参与此阶段。

（4）加载----至此阶段开始，控件树中的服务器控件已创建并初始化、状态已还原并且窗体控件反映了客户端的数据。此时，开发人员可以通过重写OnLoad()方法来实现每个请求共同的逻辑。

（5）发送回发更改通知----在此阶段，服务器控件通过引发事件作为一种信号，表明由于回发而发生的控件状态变化（因此该阶段仅用于回发过程）。为了建立这种信号，开发人员必须再次使用System.Web.UI.IPostBackDataHandler接口，并实现另一方法-RaisePostBackChangedEvent()。其判断过程为：如果控件状态因回发而更改，则LoadPostData()返回true；否则返回false。页面框架跟踪所有返回true的控件并在这些控件上调用RaisePostDataChangedEvent()。

（6）处理回发事件----该阶段处理引起回发的客户端事件。为了便于将客户端事件映射到服务器端事件上进行处理，开发人员在此阶段可以通过实现System.Web.UI.IPostBackEventHandler接口的RaisePostBackEvent()方法来实现该逻辑。由此途径，服务器控件将成功捕获回发的客户端事件进行服务器端的相应处理。

（7）预呈现----该阶段完成在生成控件之前所需要的任何工作。通常情况下是通过重写OnPreRender()方法完成该工作。需要注意的是：在该阶段，可以保存在预呈现阶段对控件状态所做的更改，而在呈现阶段进行的更改则会丢失。

（8）保存状态----如果服务器控件不维持状态，或者它有能力通过默认方式保存其所有状态而使用ViewState字典，那么开发人员不必在该阶段实现任何逻辑。因为这个保存状态的过程是自动的。如果服务器控件需要自定义状态保存，或者控件无法在ViewState字典中存储特殊的数据类型，则需要通过重写SaveViewState()方法来实现状态保存。

（9）呈现----表示向HTTP输出流中写入标记文本的过程。开发人员通过重写Render()方法使其在输出流上自定义标记文本。

（10）处置----在此阶段中，通过重写Dispose ()方法完成释放对昂贵资源的引用，如数据库链接等。

（11）卸载----完成的工作与"处置"阶段相同，但是，开发人员通常在Dispose()方法中执行清除，而不处理Unload事件。

Q15：UML

答：统一建模语言。

Q16：面向对象的概念，主要特点

答：封装，继承，多态

Q17：类划分的依据。类粒度问题

Q18：事物管理问题

第一， 谈谈final, finally, finalize的区别。

答：final 用于声明属性，方法和类，分别表示属性不可变，方法不可覆盖，类不可继承。

finally是异常处理语句结构的一部分，表示总是执行。

finalize是Object类的一个方法，在垃圾收集器执行的时候会调用被回收对象的此方法，可以覆盖此方法提供垃圾收集时的其他资源回收，例如关闭文件等。

第二，Anonymous Inner Class (匿名内部类) 是否可以extends(继承)其它类，是否可以implements(实现)interface(接口)?

第三，Static Nested Class 和 Inner Class的不同，说得越多越好(面试题有的很笼统)。

答：Nested Class （一般是C++的说法），Inner Class (一般是JAVA的说法)。Java内部类与C++嵌套类最大的不

同就在于是否有指向外部的引用上

第四，&和&&的区别。

答：&是位运算符，表示与；&&是逻辑运算符，表示”and”

第五，HashMap和Hashtable的区别。

答：Hashtable和HashMap类有三个重要的不同之处。第一个不同主要是历史原因。Hashtable是基于陈旧的Dictionary类的，HashMap是Java 1.2引进的Map接口的一个实现。



也许最重要的不同是Hashtable的方法是同步的，而HashMap的方法不是。这就意味着，虽然你可以不用采取任何特殊的行为就可以在一个多线程的应用程序中用一个Hashtable，但你必须同样地为一个HashMap提供外同步。一个方便的方法就是利用Collections类的静态的synchronizedMap()方法，它创建一个线程安全的Map对象，并把它作为一个封装的对象来返回。这个对象的方法可以让你同步访问潜在的HashMap。这么做的结果就是当你不需要同步时，你不能切断Hashtable中的同步（比如在一个单线程的应用程序中），而且同步增加了很多处理费用。



第三点不同是，只有HashMap可以让你将空值作为一个表的条目的key或value。HashMap中只有一条记录可以是一个空的key，但任意数量的条目可以是空的value。这就是说，如果在表中没有发现搜索键，或者如果发现了搜索键，但它是一个空的值，那么get()将返回null。如果有必要，用containKey()方法来区别这两种情况。



一些资料建议，当需要同步时，用Hashtable，反之用HashMap。但是，因为在需要时，HashMap可以被同步，HashMap的功能比Hashtable的功能更多，而且它不是基于一个陈旧的类的，所以有人认为，在各种情况下，HashMap都优先于Hashtable。



关于Properties

有时侯，你可能想用一个hashtable来映射key的字符串到value的字符串。DOS、Windows和Unix中的环境字符串就有一些例子，如key的字符串PATH被映射到value的字符串C:\WINDOWS;C:\WINDOWS\SYSTEM。Hashtables是表示这些的一个简单的方法，但Java提供了另外一种方法。



Java.util.Properties类是Hashtable的一个子类，设计用于String keys和values。Properties对象的用法同Hashtable的用法相象，但是类增加了两个节省时间的方法，你应该知道。



Store()方法把一个Properties对象的内容以一种可读的形式保存到一个文件中。Load()方法正好相反，用来读取文件，并设定Properties对象来包含keys和values。



注意，因为Properties扩展了Hashtable，你可以用超类的put()方法来添加不是String对象的keys和values。这是不可取的。另外，如果你将store()用于一个不包含String对象的Properties对象，store()将失败。作为put()和get()的替代，你应该用setProperty()和getProperty()，它们用String参数。

第六，Collection 和 Collections的区别。

答：Collection是集合类的上级接口，继承与他的接口主要有Set 和List.

Collections是针对集合类的一个帮助类，他提供一系列静态方法实现对各种集合的搜索、排序、线程安全化等操作

第七，什么时候用assert。

答：断言是一个包含布尔表达式的语句，在执行这个语句时假定该表达式为 true。如果表达式计算为 false，那么系统会报告一个 Assertionerror。它用于调试目的：

assert(a > 0); // throws an Assertionerror if a <= 0

断言可以有两种形式：

assert Expression1 ;

assert Expression1 : Expression2 ;

Expression1 应该总是产生一个布尔值。

Expression2 可以是得出一个值的任意表达式。这个值用于生成显示更多调试信息的 String 消息。

断言在默认情况下是禁用的。要在编译时启用断言，需要使用 source 1.4 标记：

javac -source 1.4 Test.java

要在运行时启用断言，可使用 -enableassertions 或者 -ea 标记。

要在运行时选择禁用断言，可使用 -da 或者 -disableassertions 标记。

要系统类中启用断言，可使用 -esa 或者 -dsa 标记。还可以在包的基础上启用或者禁用断言。

可以在预计正常情况下不会到达的任何位置上放置断言。断言可以用于验证传递给私有方法的参数。不过，断言不应该用于验证传递给公有方法的参数，因为不管是否启用了断言，公有方法都必须检查其参数。不过，既可以在公有方法中，也可以在非公有方法中利用断言测试后置条件。另外，断言不应该以任何方式改变程序的状态。

第八，GC是什么? 为什么要有GC?

答：垃圾回收器。释放被视为垃圾的内存

第九，String s = new String("xyz");创建了几个String Object?

答：创建了一个对象,一个引用．对象new String("xyz")呆在堆里，那个引用s在栈中

第十，Math.round(11.5)等於多少? Math.round(-11.5)等於多少?

答：12.0；-12;

第十一，short s1 = 1; s1 = s1 + 1;有什么错? short s1 = 1; s1 += 1;有什么错?

答：s1＋1是int型，所以改成s1 ＝ (short)(s1＋1)；后面的没错。

第十二，sleep() 和 wait() 有什么区别?

答：sleep()方法是使线程停止一段时间的方法。在sleep 时间间隔期满后，线程不一定立即恢复执行。这是因为在那个时刻，其它线程可能正在运行而且没有被调度为放弃执行，除非(a)"醒来"的线程具有更高的优先级 (b)正在运行的线程因为其它原因而阻塞。wait()是线程交互时，如果线程对一个同步对象x 发出一个wait()调用，该线程会暂停执行，被调对象进入等待状态，直到被唤醒或等待时间到。

第十三，Java有没有goto?

答：没有

第十四，数组有没有length()这个方法? String有没有length()这个方法?

答：数组有length属性，没有length()方法

1 String有length()方法，没有length属性

第十五，Overload和Override的区别。Overloaded的方法是否可以改变返回值的类型?

答：重载和覆盖。覆写（Override）的两个函数的函数特征相同，重载（Overload）的两个函数的函数名虽然相同，但函数特征不同。 函数特征包括函数名，参数的类型和个数。Override 是在继承的时候，如果你写的函数与要继承的函数函数特征相同，那么，加上这个关键字，在使用这个子类的这个函数的时候就看不见父类（或超类）的函数了，它被覆盖掉了。

第十六，Set里的元素是不能重复的，那么用什么方法来区分重复与否呢? 是用==还是equals()? 它们有何区别?

答：Set里的元素是不能重复的，那么用iterator()方法来区分重复与否。equals()是判读两个Set是否相等。equals()和==方法决定引用值是否指向同一对象equals()在类中被覆盖

第十七，给我一个你最常见到的runtime exception。

答：NullPointException

第十八，error和exception有什么区别?

答：error 表示恢复不是不可能但很困难的情况下的一种严重问题。比如说内存溢出。不可能指望程序能处理这样的情况。

exception 表示一种设计或实现问题。也就是说，它表示如果程序运行正常，从不会发生的情况

第十九，List, Set, Map是否继承自Collection接口?

答：List和Set是，Map不是。

第二十，abstract class和interface有什么区别?

答：接口用于规范，抽象类用于共性。接口中只能声明方法，属性，事件，索引器。而抽象类中可以有方法的实现，也可以定义非静态的类变量。抽象类是类，所以只能被单继承，但是接口却可以一次实现多个。抽象类可以提供某些方法的部分实现，接口不可以.抽象类的实例是它的子类给出的。接口的实例是实现接口的类给出的。再抽象类中加入一个方法，那么它的子类就同时有了这个方法。而在接口中加入新的方法，那么实现它的类就要重新编写（这就是为什么说接口是一个类的规范了）。接口成员被定义为公共的，但抽象类的成员也可以是私有的、受保护的、内部的或受保护的内部成员（其中受保护的内部成员只能在应用程序的代码或派生类中访问）。此外接口不能包含字段、构造函数、析构函数、静态成员或常量。

第二十一，abstract的method是否可同时是static,是否可同时是native，是否可同时是synchronized?

答：不可以将abstract与static合用。不可以将abstract与native合用

第二十二，接口是否可继承接口? 抽象类是否可实现(implements)接口? 抽象类是否可继承实体类(concrete class)?

答：可以，可以，可以

第二十三，启动一个线程是用run()还是start()?

答：start（）；

第二十四，构造器Constructor是否可被override?

答：构造器Constructor不能被继承，因此不能重写Overriding，但可以被重载Overloading。当两个分离的对象的内容和类型相配的话，返回真值

第二十五，是否可以继承String类?

答：不能

第二十六，当一个线程进入一个对象的一个synchronized方法后，其它线程是否可进入此对象的其它方法?

答：不能，一个对象的一个synchronized方法只能由一个线程访问。

第二十七，try {}里有一个return语句，那么紧跟在这个try后的finally {}里的code会不会被执行，什么时候被执行，在return前还是后?

答：会，之前。

第二十八，编程题: 用最有效率的方法算出2乘以8等於几?

答：2<<3;

第二十九，两个对象值相同(x.equals(y) == true)，但却可有不同的hash code，这句话对不对?

答：不对，其hashcode应当是相等。因为hashcode用的是同一个类的方法，同时其每一个属性值是相等的。

第三十，当一个对象被当作参数传递到一个方法后，此方法可改变这个对象的属性，并可返回变化后的结果，那么这里到底是值传递还是引用传递?

答：引用ref

第三十一，swtich是否能作用在byte上，是否能作用在long上，是否能作用在String上?

答：必须为byte，short，int或char，不可以是其它类型

第三十二，编程题: 写一个Singleton出来。

答：48、编程题: 写一个Singleton出来。

一个类Class只有一个实例存在。

一般Singleton模式通常有几种种形式:

第一种形式: 定义一个类，它的构造函数为private的，它有一个static的private的该类变量，在类初始化时实例话，通过一个public的getInstance方法获取对它的引用,继而调用其中的方法。

public class Singleton {

private Singleton(){ }

　　 //在自己内部定义自己一个实例，是不是很奇怪？

　　 //注意这是private 只供内部调用

　　 private static Singleton instance = new Singleton();

　　 //这里提供了一个供外部访问本class的静态方法，可以直接访问

　　 public static Singleton getInstance() {

　　　　 return instance;

　　 }

}

第二种形式:

public class Singleton {

　　private static Singleton instance = null;

　　public static synchronized Singleton getInstance() {

　　//这个方法比上面有所改进，不用每次都进行生成对象，只是第一次

　　//使用时生成实例，提高了效率！

　　if (instance==null)

　　　　instance＝new Singleton();

return instance; 　　 }

}

其他形式:

定义一个类，它的构造函数为private的，所有方法为static的。

一般认为第一种形式要更加安全些

数据库方面：

1.存储过程和函数的区别

答：存储过程是编译好的存储在数据库的操作，函数中不能使用临时表,和动态SQL

2.事务是什么？

答：数据库事务是指作为单个逻辑工作单元执行的一系列操作。

数据库事务的ACID属性

事务处理可以确保除非事务性单元内的所有操作都成功完成，否则不会永久更新面向数据的资源。通过将一组相关操作组合为一个要么全部成功要么全部失败的单元，可以简化错误恢复并使应用程序更加可靠。一个逻辑工作单元要成为事务，必须满足所谓的ACID(原子性、一致性、隔离性和持久性)属性：

　　• 原子性

事务必须是原子工作单元；对于其数据修改，要么全都执行，要么全都不执行。通常，与某个事务关联的操作具有共同的目标，并且是相互依赖的。如果系统只执行这些操作的一个子集，则可能会破坏事务的总体目标。原子性消除了系统处理操作子集的可能性。

　　• 一致性

事务在完成时，必须使所有的数据都保持一致状态。在相关数据库中，所有规则都必须应用于事务的修改，以保持所有数据的完整性。事务结束时，所有的内部数据结构（如 B 树索引或双向链表）都必须是正确的。某些维护一致性的责任由应用程序开发人员承担，他们必须确保应用程序已强制所有已知的完整性约束。例如，当开发用于转帐的应用程序时，应避免在转帐过程中任意移动小数点。

　　• 隔离性

由并发事务所作的修改必须与任何其它并发事务所作的修改隔离。事务查看数据时数据所处的状态，要么是另一并发事务修改它之前的状态，要么是另一事务修改它之后的状态，事务不会查看中间状态的数据。这称为可串行性，因为它能够重新装载起始数据，并且重播一系列事务，以使数据结束时的状态与原始事务执行的状态相同。当事务可序列化时将获得最高的隔离级别。在此级别上，从一组可并行执行的事务获得的结果与通过连续运行每个事务所获得的结果相同。由于高度隔离会限制可并行执行的事务数，所以一些应用程序降低隔离级别以换取更大的吞吐量。

　　• 持久性

事务完成之后，它对于系统的影响是永久性的。该修改即使出现致命的系统故障也将一直保持。

DBMS的责任和我们的任务

企业级的数据库管理系统（DBMS）都有责任提供一种保证事务的物理完整性的机制。就常用的SQL Server2000系统而言，它具备锁定设备隔离事务、记录设备保证事务持久性等机制。因此，我们不必关心数据库事务的物理完整性，而应该关注在什么情况下使用数据库事务、事务对性能的影响，如何使用事务等等。

3.游标的作用？如何知道游标已经到了最后？

答：指示当前记录的位置,检查NULL

4.触发器分为事前触发和事后触发，这两种触发有和区别。语句级触发和行级触发有何区别。

答：一个事前，一个事后。通常事前触发器可以获取事件之前和新的字段值

1。用C＃实现以下功能

a 产生一个int数组，长度为100，并向其中随机插入1-100，并且不能重复。

b 对上面生成的数组排序，需要支持升序、降序两种顺序

答：ArrayList ar=new ArrayList();

int [] list=new int[100];

for(int i=1;i<=100;i++)

{

ar.Add(i);

}

Random rm=new Random();

for(int i=0;i<100;i++)

{

int index=rm.Next(99-i);

list[i]=(int)ar[index];

ar.RemoveAt(index);

}

升序：private int min;

public void xuanZhe(int[] list)//选择排序

{

for (int i = 0; i < list.Length - 1; i++)

{

min = i;

for (int j = i + 1; j < list.Length; j++)

{

if (list[j] < list[min])

min = j;

}

int t = list[min];

list[min] = list[i];

list[i] = t;

}

}

降序：private int max;

public void xuanZhe(int[] list)//选择排序

{

for (int i = 0; i < list.Length - 1; i++)

{

max = i;

for (int j = i + 1; j < list.Length; j++)

{

if (list[j] < list[min])

max = j;

}

int t = list[max];

list[max] = list[i];

list[i] = t;

}

}

2。请说明在.net中常用的几种页面间传递参数的方法，并说出他们的优缺点。

答：

session(viewstate) 简单，但易丢失

application 全局

cookie 简单，但可能不支持，可能被伪造

input ttype="hidden" 简单，可能被伪造

url参数 简单，显示于地址栏，长度有限

数据库 稳定，安全，但性能相对弱

3。请说明.net中的错误处理机制，并举例

答：try{可能要出错的代码}catch{出错后如何处理}finally{不论是否出错都要执行}

4。请说出强名的含义

答：具有自己的key，可以在GAC为公用

5。请列出c＃中几种循环的方法，并指出他们的不同

答：for while foreach

6。请指出.net中所有类型的基类

答：object

7。请指出GAC的含义

答：全局程序集缓存

8。SQL SREVER中，向一个表中插入了新数据，如何快捷的得到自增量字段的当前值

答：用max聚合函数

您在什么情况下会用到虚方法？它与接口有什么不同？

Q：Override与重载有什么区别？

答：其实很本质的区别就是看函数特征：覆写（Override）的两个函数的函数特征相同，重载（Overload）的两个函数的函数名虽然相同，但函数特征不同。 函数特征包括函数名，参数的类型和个数。Override 是在继承的时候，如果你写的函数与要继承的函数函数特征相同，那么，加上这个关键字，在使用这个子类的这个函数的时候就看不见父类（或超类）的函数了，它被覆盖掉了。

Q：值类型与引用类型有什么区别？

答：值类型直接存储其值，引用类型存储对值的引用。值类型存储在堆栈中，而引用类型存储在托管堆上。值类型包括：简单类型、结构类型、枚举类型；引用类型包括：类、数组、接口、委托、字符串、数组。

Q：怎样理解静态变量？

答：所有实例公用一个的变量

Q：向服务器发送请求有几种方式？

答：post，get

Q：DataReader与Dataset有什么区别？

答：dataset是保存数据的数据结构，断开模式，而DataReader不承担保存数据的责任，需要自己手动关闭连接数据库，它只负责从数据源读取数据到本地而已，是只读向前的，它不是数据结构，而是网络通讯组件的高层封装。

Q：如果在一个B/S结构的系统中需要传递变量值，但是又不能使用Session、Cookie、Application，您有几种方法进行处理？

答：input type＝hidden；url；数据库；

Q：用.net做B/S结构的系统，您是用几层结构来开发，每一层之间的关系以及为什么要这样分层？

答：3层，表示层，逻辑层，数据层；安全，可维护性高

Q：软件开发过程一般有几个阶段？每个阶段的作用？

答：分析，开发，测试，维护

Q：微软推出了一系列的Application Block，请举出您所知道的Application Block并说明其作用？

Q：请列举一些您用到过的设计模式以及在什么情况下使用该模式？

Q：您对WebService的体会？

答：在网络上共享

Q：您对编程的兴趣如何？工作中遇到不懂的问题是怎样去解决的？您一般怎样去提高自己的编程水平？

答：很高；问，查；多看，多动手。

Q：您离职的原因是什么？

Q：通过超链接怎样传递中文参数？

答：URLEncode，URLDecode；

Q：请编程遍历页面上所有TextBox控件并给它赋值为string.Empty？

答：c＃方法：

public void FindTextBox(Control ctr)

{

foreach(Control parent in ctr.Controls)

{

foreach(Control child in parent.Controls)

{

If(child.GetType().ToString() == “System.Web.UI.WebControls.TextBox”)

{

((TextBox)child).Text = string.Empty;

}

FindTextBox(child);

}

}

}

Javascript方法：

<script type="text/javascript">

Function FindTextBox()

{

var controls = document.getElementsByTagName(‘input’);

for(var i=0;i<controls.length;i++)

{

if(controls[i].type == ‘text’)

{

Controls[i].value = ‘’;

}

}

}

Q：请编程实现一个冒泡排序算法？

答：public int[] maoPao(int[] list)//冒泡排序

{

int i, j, temp;

bool done = false;

j = 1;

while ((j < list.Length) && (!done))

{

done = true;

for (i = 0; i < list.Length - j; i++)

{

if (list[i] > list[i + 1])

{

done = false;

temp = list[i];

list[i] = list[i + 1];

list[i + 1] = temp;

}

}

j++;

}

return list;

}

1、override与重载的区别

答：其实很本质的区别就是看函数特征：覆写（Override）的两个函数的函数特征相同，重载（Overload）的两个函数的函数名虽然相同，但函数特征不同。 函数特征包括函数名，参数的类型和个数。Override 是在继承的时候，如果你写的函数与要继承的函数函数特征相同，那么，加上这个关键字，在使用这个子类的这个函数的时候就看不见父类（或超类）的函数了，它被覆盖掉了。

2、.net的错误处理机制是什么

答：try{可能要出错的代码}catch{出错后如何处理}finally{不论是否出错都要执行}

3、C＃中接口和类的异同

答：类是引用类型，可以继承类、接口和被继承，有默认的构造函数，有析构函数，可以使用abstract和sealed，有protected修饰符，必须使用new初始化。

接口可以说是比抽象类更抽象的类，只能继承接口，是一种规范

4、DataReader和DataSet的异同

答：dataset是保存数据的数据结构，断开模式，而DataReader不承担保存数据的责任，需要自己手动关闭连接数据库，它只负责从数据源读取数据到本地而已，是只读向前的，它不是数据结构，而是网络通讯组件的高层封装。

1.有哪几种方法可以实现一个类存取另外一个类的成员函数及属性,并请举列来加以说明和分析.

答：反射，base

2.如果需记录类的实例个数,该如何实现,请写一个简单的类于以证明.

答：static int num ＝ 0；在构造函数里num++。

3.A类是B类的基类,并且都有自己的构造,析构函数,请举例证明B类从实例化到消亡过程中构造,析构函数的执行过程.

答：A构造 →B构造 →B析构 →A析构

4.需要实现对一个字符串的处理,首先将该字符串首尾的空格去掉,如果字符串中间还有连续空格的话,仅保留一个空格,即允许字符串中间有多个空格,但连续的空格数不可超过一个.

答：string a = " a b c d e c d ";

string d = a.Replace(" ", " ");

while (true)

{

d = d.Replace(" "," ");

if (d.IndexOf(" ") == -1)

{

break;

}

}

一.填空题

1.c#中的三元运算符是\_\_?:\_\_\_?

2.当整数a赋值给一个object对象时，整数a将会被\_\_装箱(封装)\_\_\_?

3.类成员有\_\_\_\_\_种可访问形式？ （public,private,protected,internal）

4.public static const int A=1;这段代码有错误么？是什么？ const不能用static修饰

5.float f=-123.567F;

int i=(int)f;

i的值现在是\_\_\_\_\_?

答：123

6.利用operator声明且仅声明了“==”，有什么错误么?

答： 还要声明“!=”

7.委托声明的关键字是\_\_\_\_\_\_?

答：delegate

8.用sealed修饰的类有什么特点？

答：密封，不能继承

9.在Asp.net中所有的自定义用户控件都必须继承自\_\_\_\_\_\_\_\_?

答：Control

10.在.Net中所有可序列化的类都被标记为\_\_ serializable \_\_?

11.在.Net托管代码中我们不用担心内存漏洞，这是因为有了\_\_\_\_\_\_?垃圾回收器

12.下面的代码中有什么错误吗？\_\_\_\_\_\_\_

using System;

class A

{

public virtual void F(){

Console.WriteLine("A.F");

}

}

abstract class B:A

{

public abstract override void F(); // new public abstract void F();

}

13.当类T只声明了私有实例构造函数时，则在T的程序文本外部，\_\_\_不可以\_\_\_（可以 or 不可以）从T派生出新的类，不可以\_\_\_\_（可以 or 不可以）直接创建T的任何实例。

14.下面这段代码有错误么？

switch (i){

case():

CaseZero();

break;

case 1:

CaseOne();

break;

case 2:

dufault; //错误

CaseTwo();

break;

}

15.在.Net中，类System.Web.UI.Page 可以被继承么？可以

二.简答题

1.在c#中using和new这两个关键字有什么意义，请写出你所知道的意义？

答：1）. using指令。using + 命名空间名字，这样可以在程序中直接用命令空间中的类型，而不必指定类型的详细命名空间。

2）.using别名。using + 别名 = 包括详细命名空间信息的具体的类型。当一个cs引用多个命名空间时，而有相同的类型时可以这样做。比如namespace1 和namespace2下面都有myClass类时，可以using class1 ＝ namespace1.myClass;using class2 = namespace2.myClass。

3）.using语句，定义一个范围，在范围结束时处理对象。

(1)new 运算符

用于创建对象和调用构造函数。

(2)new 修饰符

用于向基类成员隐藏继承成员。

(3)new 约束

用于在泛型声明中约束可能用作类型参数的参数的类型。

4.谈谈类和结构的区别？

答：类是引用类型，可以继承类、接口和被继承，有默认的构造函数，有析构函数，可以使用abstract和sealed，有protected修饰符，必须使用new初始化。

结构是值类型，只能继承接口，不能被继承，没有默认的构造函数，可以创建，没有析构函数，不可以用abstract和sealed，没有protected修饰符，可以不用new初始化。

如何选择使用结构还是类：

1）．堆栈的空间有限，对于大量的逻辑的对象，创建类要比创建结构好一些

2）．结构表示如点、矩形和颜色这样的轻量对象，例如，如果声明一个含有 1000 个点对象的数组，则将为引用每个对象分配附加的内存。在此情况下，结构的成本较低。

3）．在表现抽象和多级别的对象层次时，类是最好的选择

4）．大多数情况下该类型只是一些数据时，结构是最佳的选择

5.一个长度为10000的字符串，通过随机从a-z中抽取10000个字符组成。请用c＃语言编写主要程序来实现。

答： using System.Text;

StringBuilder sb = new StringBuilder(0, 10000);

string strABC = "a,b,c,d,e,f,g,h,i,j,k,l,m,n,o,p,q,r,s,t,u,v,w,x,y,z";

string[] ABC = strABC.Split(',');

int len = ABC.Length;

Random rd = new Random();

for (int i = 0; i < 10000; i++)

{

sb.Append(ABC[rd.Next(len)]);

}

6.对于这样的一个枚举类型：

enum Color:byte{

Red,

Green,

Blue,

Orange

}

试写一段程序显示出枚举类型中定义的所有符号名称以及它们对应的数值。

答：string[] ss=Enum.GetNames(typeof(Color));

byte[] bb=Enum.GetValues(typeof(Color));

7.您了解设计模式么？请列出您所知道的设计模式的名称。 //

答：设计模式是一套被反复使用、多数人知晓的、经过分类编目的、代码设计经验的总结。使用设计模式是为了可重用代码、让代码更容易被他人理解、保证代码可靠性。设计模式使人们可以更加简单方便地复用成功的设计和体系结构。将已证实的技术表述成设计模式也会使新系统开发者更加容易理解其设计思路。

一个模式的四个基本要素

1. 模式名称（pattern name）

一个助记名，它用一两个词来描述模式的问题、解决方案和效果。命名一个新的模式增加了我们的设计词汇。设计模式允许我们在较高的抽象层次上进行设计。基于一个模式词汇表，我们自己以及同事之间就可以讨论模式并在编写文档时使用它们。模式名可以帮助我们思考，便于我们与其他人交流设计思想及设计结果。找到恰当的模式名也是我们设计模式编目工作的难点之一。

2. 问题(problem)

描述了应该在何时使用模式。它解释了设计问题和问题存在的前因后果，它可能描述了特定的设计问题，如怎样用对象表示算法等。也可能描述了导致不灵活设计的类或对象结构。有时候，问题部分会包括使用模式必须满足的一系列先决条件。

3. 解决方案(solution)

描述了设计的组成成分，它们之间的相互关系及各自的职责和协作方式。因为模式就像一个模板，可应用于多种不同场合，所以解决方案并不描述一个特定而具体的设计或实现，而是提供设计问题的抽象描述和怎样用一个具有一般意义的元素组合（类或对象组合）来解决这个问题。

4. 效果(consequences)

描述了模式应用的效果及使用模式应权衡的问题。尽管我们描述设计决策时，并不总提到模式效果，但它们对于评价设计选择和理解使用模式的代价及好处具有重要意义。软件效果大多关注对时间和空间的衡量，它们也表述了语言和实现问题。因为复用是面向对象设计的要素之一，所以模式效果包括它对系统的灵活性、扩充性或可移植性的影响，显式地列出这些效果对理解和评价这些模式很有帮助。

MVC，factory，singleton

8.请在SQL Server中设计表来保存一个树状结构的组织结构图（假设结构图中只有名称这一项内容需要保存），如果我想查询某一职位下的所有职位，用一个存储过程来实现，你有什么思路？

答：parent表（字段：pid，pname），child表（字段：cid，cname，pid）

递归

9.什么叫做SQL注入，如何防止？请举例说明。

答：利用sql关键字对网站进行攻击。过滤关键字'等

10.下面这段代码输出什么？为什么？

int i=5;

int j=5;

if (Object.ReferenceEquals(i,j))

Console.WriteLine("Equal");

else

Console.WriteLine("Not Equal");

答：不相等，因为比较的是对象

1.写一个实现对一段字符串翻转的方法，附加一些条件，如其中包括“，”、“.”，对其设计测试用 例 。

答： using System.Text;

string inputStr = "a,b,c,d,e,f,g,h,i,j,k,l";

char[] inputChar = inputStr.ToCharArray();

int count = inputChar.Length;

StringBuilder sb = new StringBuilder();

for (int m = count - 1; m >= 0; m--)

{

sb.Append(inputChar[m]);

}

2.对一支纸杯设计测试用例（可以是广义的杯，不一定是某一支特定功能的杯）

开发语言概念题

3.什么是反射？

答：动态获取程序集信息。

反射

公共语言运行库加载器管理应用程序域。这种管理包括将每个程序集加载到相应的应用程序域以及控制每个程序集中类型层次结构的内存布局。

程序集包含模块，而模块包含类型，类型又包含成员。反射则提供了封装程序集、模块和类型的对象。您可以使用反射动态地创建类型的实例，将类型绑定到现有对象，或从现有对象中获取类型。然后，可以调用类型的方法或访问其字段和属性。

4.用Singleton如何写设计模式

答：public class Singleton

{

private static Singleton instance;

public string name;

private Singleton()//将singleton函数变为private,防止直接调用new来创建它.

{

name="guqi";

Console.WriteLine("你只可以看到我一次哦");

}

public static Singleton GetInstance()

{

if (instance==null)

{

instance=new Singleton();

return instance;

}

else return instance;

}

}

5.C#中的垃圾回收机制是怎样的？

答：.NET Framework 的垃圾回收器管理应用程序的内存分配和释放。每次您使用 new 运算符创建对象时，运行库都从托管堆为该对象分配内存。只要托管堆中有地址空间可用，运行库就会继续为新对象分配空间。但是，内存不是无限大的。最终，垃圾回收器必须执行回收以释放一些内存。垃圾回收器优化引擎根据正在进行的分配情况确定执行回收的最佳时间。当垃圾回收器执行回收时，它检查托管堆中不再被应用程序使用的对象并执行必要的操作来回收它们占用的内存。

6.什么是Application Pool？

答：Web应用，类似Thread Pool，提高并发性能

7.链表和数组的区别，各有什么优缺点.

答：一个可以动态增长，一个固定,性能数组教好

8.Remoting在客户端服务器怎么实现 ?

答：创建一个TcpClientChannel对象，在ChannelServices中注册，然后使用Activator类把代理对象返回给远程对象。

9.什么是友元函数？

答：friendly声明，可以访问protect级别方法

10.用标准C如何实现多态？

答：虚方法

11.什么是虚函数？

答：可以被重写

12.什么是抽象函数？

答：必须被重写

13.什么是内存泄漏，怎样最简单的方法判断被存泄漏 ?

答：C++,C中忘了释放内存，内存不会再次分配

英语题

14.用英文介绍一下使用C#/C++做的项目，主要功能

15.如果要与美国开电话会议，会如何与美国的工程师沟通

16.如果老板认为你的技术落后，你会怎么回答

数据库知识题

17.使用什么工具来调用存储过程

答：ADO.Net

18.SQL Server的两种索引是何形式？索引的作用？索引的优缺点？

答：集聚索引，非聚集索引。提高查询速度。但是会过多的占用磁盘空间。

19.触发器的作用

答：保证数据库操作变更能接到通知

其它知识题及问题

20.什么是Web Service？

答：web服务，是利用SOAP（简单对象访问协议，Simple Object Access Protocol）在HTTP上执行远程方法调用的一种方法，也可以使用WSDL（Web Service Description Language，Web服务描述语言）来完整的描述Web服务，基于XML标准。

21.什么是XML？

答：可扩展标记语言，可以做配置文件，可以传输数据，可以存储数据。

22.Socket怎么实现？

答：Socket 类为网络通信提供了一套丰富的方法和属性。Socket 类允许您使用 ProtocolType 枚举中所列出的任何一种协议执行异步和同步数据传输。Socket 类遵循异步方法的 .NET Framework 命名模式；例如，同步 Receive 方法对应于异步 BeginReceive 和 EndReceive 方法。

如果应用程序在执行期间只需要一个线程，请使用下面的方法，这些方法适用于同步操作模式。

• 如果当前使用的是面向连接的协议（如 TCP），则服务器可以使用 Listen 方法侦听连接。Accept 方法处理任何传入的连接请求，并返回可用于与远程主机进行数据通信的 Socket。可以使用此返回的 Socket 来调用 Send 或 Receive 方法。如果要指定本地 IP 地址和端口号，请在调用 Listen 方法之前先调用 Bind 方法。如果您希望基础服务提供程序为您分配可用端口，请使用端口号 0。如果希望连接到侦听主机，请调用 Connect 方法。若要进行数据通信，请调用 Send 或 Receive 方法。

• 如果当前使用的是无连接协议（如 UDP），则根本不需要侦听连接。调用 ReceiveFrom 方法可接受任何传入的数据报。使用 SendTo 方法可将数据报发送到远程主机。

若要在执行过程中使用单独的线程处理通信，请使用下面的方法，这些方法适用于异步操作模式。

• 如果当前使用的是面向连接的协议（如 TCP），则可使用 Socket、BeginConnect 和 EndConnect 方法来连接侦听主机。通过使用 BeginSend 和 EndSend 方法，或者使用 BeginReceive 和 EndReceive 方法，可以进行异步数据通信。可以使用 BeginAccept 和 EndAccept 处理传入的连接请求。

• 如果您使用的是 UDP 等无连接协议，则可以使用 BeginSendTo 和 EndSendTo 来发送数据报，而使用 BeginReceiveFrom 和 EndReceiveFrom 来接收数据报。

如果对一个套接字执行多个异步操作，它们不一定按启动时的顺序完成。

当数据发送和数据接收完成之后，可使用 Shutdown 方法来禁用 Socket。在调用 Shutdown 之后，可调用 Close 方法来释放与 Socket 关联的所有资源。

通过 Socket 类，您可以使用 SetSocketOption 方法来配置 Socket。可以使用 GetSocketOption 方法来检索这些设置。

1.ADO.NET的四大对象是什么？

2.面向对象的三大特征？

3.net framwork两种配置方式？

4.public,private,protected区别？

5.读取水晶报表的两种方式？

6.函数实现：对学生成绩进行排序？

7.根据所给xml schemal结构写出相应的xml文件？

8.如何实现webservice动态配置？

9.你是如何理解软件开发中的多层结构?

10.下列那些属于引用类型，那些属于值类型？

数组，接口，抽象类，委托，结构

11.接口能否继承接口？

新北邦java笔试题

1题

string str1="1"+"2"+"3"+....+"1000"; 1式

string str2;

for(int i = 1; i <=1000 ;i++)

str2+=i; 2式

stringbuffer str3 = new stringbuffer();

for(int i = 1; i <=1000 ;i++)

str3 = str3.append(i); 3式

1式和2式那个执行速度快？

2式和3式那个执行速度快？

2题

public class Test{

private int num1;

private static int num2;

private int num3;

static {

num2=10;

system.put.println(" init num");

}

public Test(){

num1=10;

num3=10;

}

public void f1(int num1,int num3){

num1=20;

num2=20;

num3=20;

}

public void f2(int num4){

num1=num4;

num2=30;

num3=30;

}

public void print(){

system.put.println(" num1="+num1);

system.put.println(" num2="+num2);

system.put.println(" num3="+num3);

}

public static void main(){

Test test = new Test();

test.f1(20,20);

test.print(); 1式

test.f1(30,30);

test.print(); 2式

}

}

程序在1式处显示什么？

程序在2式处显示什么？

还有一道 输入输出流的题

还有一道 对象的构造，析构，和静态块的题

还有一道javaScript题，问settimeout()函数

**中海讯通笔试题**

1. asp.net页面提交过程（开始、初始化页面、提交、页面处理、呈现、卸载）
2. asp.net页面过程中存储用户数据的方法，各自特点。
3. 字符串的比较、删除、与给定字符串与比较字符串相一致的字符
4. 三个表的多表查询。

1.如何用Asp判断你的网站的虚拟物理路径   
答：使用Mappath方法   
< p align="center" >< font size="4" face="Arial" >< b >   
The Physical path to this virtual website is:   
< /b >< /font >   
< font color="#FF0000" size="6" face="Arial" >   
< %= Server.MapPath("\")% >   
< /font >< /p >   
2.我如何知道使用者所用的浏览器？   
答：使用the Request object方法   
strBrowser=Request.ServerVariables("HTTP\_USER\_AGENT")   
If Instr(strBrowser,"MSIE") < > 0 Then   
Response.redirect("ForMSIEOnly.htm")   
Else   
Response.redirect("ForAll.htm")   
End If   
  
3.如何计算每天的平均反复访问人数   
答：解决方法   
< % startdate=DateDiff("d",Now,"01/01/1990")   
if strdate< 0 then startdate=startdate\*-1   
avgvpd=Int((usercnt)/startdate) % >   
显示结果   
< % response.write(avgvpd) % >   
that is it.this page have been viewed since November 10,1998   
  
4.如何显示随机图象   
< % dim p,ppic,dpic   
ppic=12   
randomize   
p=Int((ppic\*rnd)+1)   
dpic="graphix/randompics/"&p&".gif"   
% >   
显示   
< img src="< %=dpic% >" >   
  
5.如何回到先前的页面   
答：< a href="< %=request.serverVariables("Http\_REFERER")% >" >preivous page< /a >   
或用图片如：< img src="arrowback.gif" alt="< %=request.serverVariables("HTTP\_REFERER")% >" >   
  
6.如何确定对方的IP地址   
答：< %=Request.serverVariables("REMOTE\_ADDR)% >   
  
7.如何链结到一副图片上   
答：< % @Languages=vbs cript % >   
< % response.expires=0   
strimagename="graphix/errors/erroriamge.gif"   
response.redirect(strimagename)   
% >   
  
8.强迫输入密码对话框   
答：把这句话放载页面的开头   
< % response.status="401 not Authorized"   
response.end   
% >   
  
9.如何传递变量从一页到另一页   
答：用 HIDDEN 类型来传递变量   
< % form method="post" action="mynextpage.asp" >   
< % for each item in request.form % >   
< input namee="< %=item% >" type="HIDDEN"   
value="< %=server.HTMLEncode(Request.form(item)) % >" >   
< % next % >   
< /form >   
  
10.为何我在 asp 程序内使用 msgbox，程序出错说没有权限   
答：由于 asp 是服务器运行的，如果可以在服务器显示一个对话框，那么你只好等有人按了确定之后，你的程序才能继续执行，而一般服务器不会有人守着，所以微软不得不禁止这个函数，并胡乱告诉你 (:) 呵呵) 没有权限。但是ASP和客户端脚本结合倒可以显示一个对话框，as follows:   
< % yourVar="测试对话框"% >   
< % s cript language='javas cript' >   
alert("< %=yourvar% >")   
< /s cript >   
  
11.有没有办法保护自己的源代码，不给人看到   
答：可以去下载一个微软的Windows s cript Encoder，它可以对asp的脚本和客户端javas cript/vbs cript脚本进行加密。。。不过客户端加密后，只有ie5才能执行，服务器端脚本加密后，只有服务器上安装有s cript engine 5（装一个ie5就有了）才能执行。   
  
12.怎样才能将 query string 从一个 asp 文件传送到另一个？   
答：前者文件加入下句： Response.Redirect("second.asp?" & Request.ServerVariables("QUERY\_STRING"))   
  
13.global.asa文件总是不起作用?   
答：只有web目录设置为web application, global.asa才有效，并且一个web application的根目录下 global.asa才有效。IIS4可以使用Internet Service Manager设置application setting 怎样才能使得htm文件如同asp文件一样可以执行脚本代码？   
  
14.怎样才能使得htm文件如同asp文件一样可以执行脚本代码？   
答：Internet Sevices Manager - > 选择default web site - >右鼠键- >菜单属性-〉主目录- > 应用程序设置（Application Setting）- > 点击按钮 "配置"- > app mapping - >点击按钮"Add" - > executable browse选择 \WINNT\SYSTEM32\INETSRV\ASP.DLL EXTENSION 输入 htm method exclusions 输入PUT.DELETE 全部确定即可。但是值得注意的是这样对htm也要由asp.dll处理，效率将降低。   
  
15.如何注册组件   
答：有两种方法。   
第一种方法：手工注册 DLL 这种方法从IIs 3.0一直使用到IIs 4.0和其它的Web Server。它需要你在命令行方式下来执行，进入到包含有DLL的目录，并输入：regsvr32 component\_name.dll 例如 c:\temp\regsvr32 AspEmail.dll 它会把dll的特定信息注册入服务器中的注册表中。然后这个组件就可以在服务器上使用了，但是这个方法有一个缺陷。当使用这种方法注册完毕组件后，该组件必须要相应的设置NT的匿名帐号有权限执行这个dll。特别是一些组件需要读取注册表，所以，这个注册组件的方法仅仅是使用在服务器上没有MTS的情况下，要取消注册这个dll，使用：regsvr32 /u aspobject.dll example c:\temp\regsvr32 /u aneiodbc.dll   
  
第二种方法：使用MTS(Microsoft Transaction Server) MTS是IIS 4新增特色，但是它提供了巨大的改进。MTS允许你指定只有有特权的用户才能够访问组件，大大提高了网站服务器上的安全性设置。在MTS上注册组件的步骤如下：   
1) 打开IIS管理控制台。   
2) 展开transaction server，右键单击"pkgs installed"然后选择"new package"。   
3) 单击"create an empty package"。   
4) 给该包命名。   
5) 指定administrator帐号或则使用"interactive"（如果服务器经常是使用administrator 登陆的话）。   
6) 现在使用右键单击你刚建立的那个包下面展开后的"components"。选择 "new then component"。   
7) 选择 "install new component" 。   
8) 找到你的.dll文件然后选择next到完成。   
要删除这个对象，只要选择它的图标，然后选择delete。   
附注：特别要注意第二种方法，它是用来调试自己编写组件的最好方法，而不必每次都需要重新启动机器了。   
  
16. ASP与Access数据库连接：   
  
<%@ language=VBs cript%>   
<%   
dim conn,mdbfile   
mdbfile=server.mappath("数据库名称.mdb")   
set conn=server.createobject("adodb.connection")   
conn.open "driver={microsoft access driver (\*.mdb)};uid=admin;pwd=数据库密码;dbq="&mdbfile   
  
%>   
  
17. ASP与SQL数据库连接：   
  
<%@ language=VBs cript%>   
<%   
dim conn   
set conn=server.createobject("ADODB.connection")   
con.open "PROVIDER=SQLOLEDB;DATA SOURCE=SQL服务器名称或IP地址;UID=sa;PWD=数据库密码;DATABASE=数据库名称   
%>   
  
建立记录集对象：   
  
set rs=server.createobject("adodb.recordset")   
rs.open SQL语句,conn,3,2   
  
18. SQL常用命令使用方法：   
  
(1) 数据记录筛选：   
  
sql="select \* from 数据表 where 字段名=字段值 order by 字段名 [desc]"   
  
sql="select \* from 数据表 where 字段名 like '%字段值%' order by 字段名 [desc]"   
  
sql="select top 10 \* from 数据表 where 字段名 order by 字段名 [desc]"   
  
sql="select \* from 数据表 where 字段名 in ('值1','值2','值3')"   
  
sql="select \* from 数据表 where 字段名 between 值1 and 值2"   
  
(2) 更新数据记录：   
  
sql="update 数据表 set 字段名=字段值 where 条件表达式"   
  
sql="update 数据表 set 字段1=值1,字段2=值2 …… 字段n=值n where 条件表达式"   
  
(3) 删除数据记录：   
  
sql="delete from 数据表 where 条件表达式"   
  
sql="delete from 数据表" (将数据表所有记录删除)   
  
(4) 添加数据记录：   
  
sql="insert into 数据表 (字段1,字段2,字段3 …) valuess (值1,值2,值3 …)"   
  
sql="insert into 目标数据表 select \* from 源数据表" (把源数据表的记录添加到目标数据表)   
  
(5) 数据记录统计函数：   
  
AVG(字段名) 得出一个表格栏平均值   
COUNT(\*|字段名) 对数据行数的统计或对某一栏有值的数据行数统计   
MAX(字段名) 取得一个表格栏最大的值   
MIN(字段名) 取得一个表格栏最小的值   
SUM(字段名) 把数据栏的值相加   
  
引用以上函数的方法：   
  
sql="select sum(字段名) as 别名 from 数据表 where 条件表达式"   
set rs=conn.excute(sql)   
  
用 rs("别名") 获取统的计值，其它函数运用同上。   
  
(5) 数据表的建立和删除：   
  
CREATE TABLE 数据表名称(字段1 类型1(长度),字段2 类型2(长度) …… )   
  
例：CREATE TABLE tab01(name varchar(50),datetime default now())   
  
DROP TABLE 数据表名称 (永久性删除一个数据表)   
  
19. 记录集对象的方法：   
  
rs.movenext 将记录指针从当前的位置向下移一行   
rs.moveprevious 将记录指针从当前的位置向上移一行   
rs.movefirst 将记录指针移到数据表第一行   
rs.movelast 将记录指针移到数据表最后一行   
rs.absoluteposition=N 将记录指针移到数据表第N行   
rs.absolutepage=N 将记录指针移到第N页的第一行   
rs.pagesize=N 设置每页为N条记录   
rs.pagecount 根据 pagesize 的设置返回总页数   
rs.recordcount 返回记录总数   
rs.bof 返回记录指针是否超出数据表首端，true表示是，false为否   
rs.eof 返回记录指针是否超出数据表末端，true表示是，false为否   
rs.delete 删除当前记录，但记录指针不会向下移动   
rs.addnew 添加记录到数据表末端   
rs.update 更新数据表记录   
  
－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－   
  
20 Recordset对象方法   
  
Open方法   
  
recordset.Open Source,ActiveConnection,CursorType,LockType,Options   
  
Source   
Recordset对象可以通过Source属性来连接Command对象。Source参数可以是一个Command对象名称、一段SQL命令、一个指定的数据表名称或是一个Stored Procedure。假如省略这个参数，系统则采用Recordset对象的Source属性。   
  
ActiveConnection   
Recordset对象可以通过ActiveConnection属性来连接Connection对象。这里的ActiveConnection可以是一个Connection对象或是一串包含数据库连接信息（ConnectionString）的字符串参数。   
  
CursorType   
Recordset对象Open方法的CursorType参数表示将以什么样的游标类型启动数据，包括adOpenForwardOnly、adOpenKeyset、adOpenDynamic及adOpenStatic，分述如下：   
--------------------------------------------------------------   
常数 常数值 说明   
-------------------------------------------------------------   
adOpenForwardOnly 0 缺省值，启动一个只能向前移动的游标（Forward Only）。   
adOpenKeyset 1 启动一个Keyset类型的游标。   
adOpenDynamic 2 启动一个Dynamic类型的游标。   
adOpenStatic 3 启动一个Static类型的游标。   
-------------------------------------------------------------   
以上几个游标类型将直接影响到Recordset对象所有的属性和方法，以下列表说明他们之间的区别。   
  
-------------------------------------------------------------   
Recordset属性 adOpenForwardOnly adOpenKeyset adOpenDynamic adOpenStatic   
-------------------------------------------------------------   
AbsolutePage 不支持 不支持 可读写 可读写   
AbsolutePosition 不支持 不支持 可读写 可读写   
ActiveConnection 可读写 可读写 可读写 可读写   
BOF 只读 只读 只读 只读   
Bookmark 不支持 不支持 可读写 可读写   
CacheSize 可读写 可读写 可读写 可读写   
CursorLocation 可读写 可读写 可读写 可读写   
CursorType 可读写 可读写 可读写 可读写   
EditMode 只读 只读 只读 只读   
EOF 只读 只读 只读 只读   
Filter 可读写 可读写 可读写 可读写   
LockType 可读写 可读写 可读写 可读写   
MarshalOptions 可读写 可读写 可读写 可读写   
MaxRecords 可读写 可读写 可读写 可读写   
PageCount 不支持 不支持 只读 只读   
PageSize 可读写 可读写 可读写 可读写   
RecordCount 不支持 不支持 只读 只读   
Source 可读写 可读写 可读写 可读写   
State 只读 只读 只读 只读   
Status 只读 只读 只读 只读   
AddNew 支持 支持 支持 支持   
CancelBatch 支持 支持 支持 支持   
CancelUpdate 支持 支持 支持 支持   
Clone 不支持 不支持   
Close 支持 支持 支持 支持   
Delete 支持 支持 支持 支持   
GetRows 支持 支持 支持 支持   
Move 不支持 支持 支持 支持   
MoveFirst 支持 支持 支持 支持   
MoveLast 不支持 支持 支持 支持   
MoveNext 支持 支持 支持 支持   
MovePrevious 不支持 支持 支持 支持   
NextRecordset 支持 支持 支持 支持   
Open 支持 支持 支持 支持   
Requery 支持 支持 支持 支持   
Resync 不支持 不支持 支持 支持   
Supports 支持 支持 支持 支持   
Update 支持 支持 支持 支持   
UpdateBatch 支持 支持 支持 支持   
--------------------------------------------------------------   
其中NextRecordset方法并不适用于Microsoft Access数据库。   
  
LockType   
Recordset对象Open方法的LockType参数表示要采用的Lock类型，如果忽略这个参数，那么系统会以Recordset对象的LockType属性为预设值。LockType参数包含adLockReadOnly、adLockPrssimistic、adLockOptimistic及adLockBatchOptimistic等，分述如下：   
  
-------------------------------------------------------------   
常数 常数值 说明   
--------------------------------------------------------------   
adLockReadOnly 1 缺省值，Recordset对象以只读方式启动，无法运行AddNew、Update及Delete等方法   
adLockPrssimistic 2 当数据源正在更新时，系统会暂时锁住其他用户的动作，以保持数据一致性。   
adLockOptimistic 3 当数据源正在更新时，系统并不会锁住其他用户的动作，其他用户可以对数据进行增、删、改的操作。   
adLockBatchOptimistic 4 当数据源正在更新时，其他用户必须将CursorLocation属性改为adUdeClientBatch才能对数据进行增、   
删、改的操作。

<%   
'Asp连SQLServer数据库(本地数据库)   
dim conn,connstr   
  
connstr="Provider = Sqloledb; User ID =sa; Password =123123; Initial Catalog =vote; Data Source =(local);"   
Set conn = Server.CreateObject("ADODB.Connection")   
conn.Open connstr   
If Err then   
Err.Clear   
Set conn = Nothing   
Response.Write "系统调整中"   
Response.End   
End If   
%>   
  
<%   
'Asp连SQLServer数据库(远程数据库)   
dim conn,connstr   
  
connstr="Provider = Sqloledb; User ID =hm; Password =newfeel; Initial Catalog =vote; Data Source =192.168.168.38;"   
Set conn = Server.CreateObject("ADODB.Connection")   
conn.Open connstr   
If Err then   
Err.Clear   
Set conn = Nothing   
Response.Write "系统调整中"   
Response.End   
End If   
%>   
  
  
<%   
dim MM\_gbook\_STRING,db,conn   
db="database/Shop.mdb" '数据库路径及名称   
Set conn = Server.CreateObject("ADODB.Connection")   
MM\_gbook\_STRING="Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" & Server.MapPath(""&db&"")   
conn.Open MM\_gbook\_STRING   
%>   
  
'带密码   
dim conn   
set conn=server.CreateObject("adodb.connection")   
conn.open "provider=microsoft.jet.oledb.4.0;data source="&Server.MapPath("mydata.mdb")&";uid=id;pwd=pwd"

c#笔试题总结（二）1. new 关键字用法(1)new 运算符

用于创建对象和调用构造函数。

(2)new 修饰符

用于向基类成员隐藏继承成员。

(3)new 约束

用于在泛型声明中约束可能用作类型参数的参数的类型。

2.如何把一个Array复制到ArrayList里

(1) 实现1

string[] s ={ "111", "22222" };

ArrayList list = new ArrayList();

list.AddRange(s);

(2)实现2

string[] s ={ "111", "22222" };

ArrayList list = new ArrayList(s);

3.DataGrid的Datasouse可以连接什么数据源

l DataTable

l DataView

l DataSet

l DataViewManager

l 任何实现IListSource接口的组件

l 任何实现IList接口的组件

4.概述反射和序列化

反射

公共语言运行库加载器管理应用程序域。这种管理包括将每个程序集加载到相应的应用程序域以及控制每个程序集中类型层次结构的内存布局。

程序集包含模块，而模块包含类型，类型又包含成员。反射则提供了封装程序集、模块和类型的对象。您可以使用反射动态地创建类型的实例，将类型绑定到现有对象，或从现有对象中获取类型。然后，可以调用类型的方法或访问其字段和属性。

序列化

序列化是将对象状态转换为可保持或传输的格式的过程。与序列化相对的是反序列化，它将流转换为对象。这两个过程结合起来，可以轻松地存储和传输数据。

5.概述O/R Mapping 的原理

利用反射，配置将对象和数据库表映射。

6. 可访问性级别有哪几种

l public 访问不受限制。

l protected 访问仅限于包含类或从包含类派生的类型。

l internal 访问仅限于当前程序集。

l protected internal 访问仅限于从包含类派生的当前程序集或类型。

l private 访问仅限于包含类型。

7. sealed 修饰符有什么特点

sealed 修饰符可以应用于类、实例方法和属性。密封类不能被继承。密封方法会重写基类中的方法，但其本身不能在任何派生类中进一步重写。当应用于方法或属性时，sealed 修饰符必须始终与 override一起使用。

8.列举ADO.NET中的共享类和数据库特定类

共享类

DataSet

DataTable

DataRow

DataColumn

DataRelation

Constraint

DataColumnMapping

DataTableMapping

特定类

(x)Connection

(x)Command

(x)CommandBuilder

(x)DataAdapter

(x)DataReader

(x)Parameter

(x)Transaction

9.执行下面代码后：

String strTemp ="abcdefg 某某某"；

Int i=System.Text.Encoding.Default.GetBytes(strTemp).Length;

Int j=strTemp.Length;

问：

i=?

j=?

i=(14 ) ；j=(11 ) 中文两个字节

10.C#中，string str = null 与 string str =""，请尽量用文字说明区别。

string str =""初始化对象分配空间

而string str=null初始化对象

11.详述.NET里class和struct的异同

结构与类共享几乎所有相同的语法，但结构比类受到的限制更多：

尽管结构的静态字段可以初始化，结构实例字段声明还是不能使用初始值设定项。

结构不能声明默认构造函数（没有参数的构造函数）或析构函数。

结构的副本由编译器自动创建和销毁，因此不需要使用默认构造函数和析构函数。实际上，编译器通过为所有字段赋予默认值（参见默认值表）来实现默认构造函数。结构不能从类或其他结构继承。

结构是值类型 -- 如果从结构创建一个对象并将该对象赋给某个变量，变量则包含结构的全部值。复制包含结构的变量时，将复制所有数据，对新副本所做的任何修改都不会改变旧副本的数据。由于结构不使用引用，因此结构没有标识 -- 具有相同数据的两个值类型实例是无法区分的。C# 中的所有值类型本质上都继承自 ValueType，后者继承自 Object。

编译器可以在一个称为装箱的过程中将值类型转换为引用类型。

结构具有以下特点：

l 结构是值类型，而类是引用类型。

l 向方法传递结构时，结构是通过传值方式传递的，而不是作为引用传递的。

l 与类不同，结构的实例化可以不使用 new 运算符。

l 结构可以声明构造函数，但它们必须带参数。

l 一个结构不能从另一个结构或类继承，而且不能作为一个类的基。所有结构都直接继承自 System.ValueType，后者继承自 System.Object。

l 结构可以实现接口。

l 在结构中初始化实例字段是错误的。

12.什么叫应用程序域？什么是托管代码？什么是强类型系统？什么是装箱和拆箱？什么是重载？CTS、CLS和CLR分别作何解释？

应用程序域

应用程序域为安全性、可靠性、版本控制以及卸载程序集提供了隔离边界。应用程序域通常由运行库宿主创建，运行库宿主负责在运行应用程序之前引导公共语言运行库。应用程序域提供了一个更安全、用途更广的处理单元，公共语言运行库可使用该单元提供应用程序之间的隔离。

托管代码

使用基于公共语言运行库的语言编译器开发的代码称为托管代码；托管代码具有许多优点，例如：跨语言集成、跨语言异常处理、增强的安全性、版本控制和部署支持、简化的组件交互模型、调试和分析服务等。

装箱和拆箱

装箱和拆箱使值类型能够被视为对象。对值类型装箱将把该值类型打包到 Object 引用类型的一个实例中。这使得值类型可以存储于垃圾回收堆中。拆箱将从对象中提取值类型。

重载

每个类型成员都有一个唯一的签名。方法签名由方法名称和一个参数列表（方法的参数的顺序和类型）组成。只要签名不同，就可以在一种类型内定义具有相同名称的多种方法。当定义两种或多种具有相同名称的方法时，就称作重载。

CTS通用类型系统 (common type system)

一种确定公共语言运行库如何定义、使用和管理类型的规范。

CLR公共语言运行库

.NET Framework 提供了一个称为公共语言运行库的运行时环境，它运行代码并提供使开发过程更轻松的服务。

CLS公共语言规范

要和其他对象完全交互，而不管这些对象是以何种语言实现的，对象必须只向调用方公开那些它们必须与之互用的所有语言的通用功能。为此定义了公共语言规范 (CLS)，它是许多应用程序所需的一套基本语言功能。

强类型

C# 是强类型语言；因此每个变量和对象都必须具有声明类型。

13.值类型和引用类型的区别？

基于值类型的变量直接包含值。将一个值类型变量赋给另一个值类型变量时，将复制包含的值。这与引用类型变量的赋值不同，引用类型变量的赋值只复制对对象的引用，而不复制对象本身。

所有的值类型均隐式派生自 System.ValueType。

与引用类型不同，从值类型不可能派生出新的类型。但与引用类型相同的是，结构也可以实现接口。

与引用类型不同，值类型不可能包含 null 值。然而，可空类型功能允许将 null 赋给值类型。

每种值类型均有一个隐式的默认构造函数来初始化该类型的默认值。

值类型主要由两类组成：结构、枚举

结构分为以下几类：Numeric（数值）类型、整型、浮点型、decimal、bool、用户定义的结构。

引用类型的变量又称为对象，可存储对实际数据的引用。声明引用类型的关键字：class、interface、delegate、内置引用类型： object、string

14.如何理解委托

委托类似于 C++ 函数指针，但它是类型安全的。

委托允许将方法作为参数进行传递。

委托可用于定义回调方法。

委托可以链接在一起；例如，可以对一个事件调用多个方法。

方法不需要与委托签名精确匹配。有关更多信息，请参见协变和逆变。

C# 2.0 版引入了匿名方法的概念，此类方法允许将代码块作为参数传递，以代替单独定义的方法。

15.C#中的接口和类有什么异同。

异：

不能直接实例化接口。

接口不包含方法的实现。

接口、类和结构可从多个接口继承。但是C# 只支持单继承：类只能从一个基类继承实现。

类定义可在不同的源文件之间进行拆分。

同：

接口、类和结构可从多个接口继承。

接口类似于抽象基类：继承接口的任何非抽象类型都必须实现接口的所有成员。

接口可以包含事件、索引器、方法和属性。

一个类可以实现多个接口。

16.ASP.net的身份验证方式有哪些

Windows 身份验证提供程序

提供有关如何将 Windows 身份验证与 Microsoft Internet 信息服务 (IIS) 身份验证结合使用来确保 ASP.NET 应用程序安全的信息。

Forms 身份验证提供程序

提供有关如何使用您自己的代码创建应用程序特定的登录窗体并执行身份验证的信息。使用 Forms 身份验证的一种简便方法是使用 ASP.NET 成员资格和 ASP.NET 登录控件，它们一起提供了一种只需少量或无需代码就可以收集、验证和管理用户凭据的方法。

Passport 身份验证提供程序

提供有关由 Microsoft 提供的集中身份验证服务的信息，该服务为成员站点提供单一登录和核心配置

17.活动目录的作用

Active Directory存储了有关网络对象的信息，并且让管理员和用户能够轻松地查找和使用这些信息。Active Directory使用了一种结构化的数据存储方式，并以此作为基础对目录信息进行合乎逻辑的分层组织。

18.解释一下UDDI、WSDL的意义及其作用

UDDI

统一描述、发现和集成协议(UDDI, Universal Description, Discovery and Integration)是一套基于Web的、分布式的、为Web服务提供的信息注册中心的实现标准规范，同时也包含一组使企业能将自身提供的Web服务注册以使得别的企业能够发现的访问协议的实现标准。UDDI 提供了一组基于标准的规范用于描述和发现服务，还提供了一组基于因特网的实现。

WSDL

WSDL描述Web服务的公共接口。这是一个基于XML的关于如何与Web服务通讯和使用的服务描述；

l 服务 URL 和命名空间

l 网络服务的类型（可能还包括 SOAP 的函数调用，正像我所说过的，WSDL 足够自如地去描述网络服务的广泛内容）

l 有效函数列表

l 每个函数的参数

l 每个参数的类型

l 每个函数的返回值及其数据类型

19.什么是SOAP

SOAP（Simple Object Access Protocol ）简单对象访问协议是在分散或分布式的环境中交换信息并执行远程过程调用的协议，是一个基于XML的协议。使用SOAP，不用考虑任何特定的传输协议（最常用的还是HTTP协议），可以允许任何类型的对象或代码，在任何平台上，以任何一直语言相互通信。

SOAP 是一种轻量级协议，用于在分散型、分布式环境中交换结构化信息。 SOAP 利用 XML 技术定义一种可扩展的消息处理框架，它提供了一种可通过多种底层协议进行交换的消息结构。 这种框架的设计思想是要独立于任何一种特定的编程模型和其他特定实现的语义。

SOAP 定义了一种方法以便将 XML 消息从 A 点传送到 B 点。 为此，它提供了一种基于 XML 且具有以下特性的消息处理框架：1) 可扩展，2) 可通过多种底层网络协议使用，3) 独立于编程模型。

20.如何部署一个ASP.net页面

VS 2005和VS 2003都有发布机制。2003可以发布然后再复制部署。

VS2005基本上可以直接部署到对应位置。

21.如何理解.net中的垃圾回收机制

.NET Framework 的垃圾回收器管理应用程序的内存分配和释放。每次您使用 new 运算符创建对象时，运行库都从托管堆为该对象分配内存。只要托管堆中有地址空间可用，运行库就会继续为新对象分配空间。但是，内存不是无限大的。最终，垃圾回收器必须执行回收以释放一些内存。垃圾回收器优化引擎根据正在进行的分配情况确定执行回收的最佳时间。当垃圾回收器执行回收时，它检查托管堆中不再被应用程序使用的对象并执行必要的操作来回收它们占用的内存。

第一部分：PB(PB6 ~ PB7)(每题10分)

1． 编写一个全局函数f\_center()，参数aw\_window 为window类型，使该窗口自动居中。

2． PB中如何存取Image或Text类型的数据，请举例说明。

3． 如何检测数据窗口是否被修改。

4． 编写一个全局函数f\_getparentwindow()，参数1 apo\_parm为PowerObject类型，参数2 aw\_parent 为window类型，以参考方式传递，抓取apo\_parm的父窗口（注意必须是父窗口，而非父对象）赋于aw\_parent。

第二部分：SQL（MS SQL SERVER / ORACLE / SYBASE）(每题10分)

1． 表名为tab\_1，其中有一个字段为code，写一条SELECT按code从小到大排序的前10条记录并按code从大到小排序（选所有字段）

2． 有一个表t\_customer，栏位code为Varchar(20)，name为Varchar(40)，Code为主键栏位，写一条SQL语句删除name相同但code不同且code不是相同name对应的最大值的记录。（限一条SQL语句）

3． 在基表inv\_trans\_detail上创建一个触发器，当插入一笔资料时更新库存inv\_current\_stock。Inv\_trans\_detail的交易数量栏位为qty\_trans，运算方式栏位为operation\_type，可取“+”、“-”、“U”三个值，如为“+”，则增加库存，如为“-”，则减少库存，如为“U”， 则增加库存（根据Qty\_trans的正负实际增加/减少库存）。Inv\_current\_stock的当前库存数量栏位为qty\_current。两个表以part\_no连接起来。

第三部分：上机操作(每题30分)

1． 在MSSQLSERVER7的MRPII数据库（或Sybase Adaptive Server Anywhere的EAS Demo DB V3数据库）建一个存储过程，抓取数据库中所有表和视图，设该存储过程名为usp\_getusertable，在PB中用usp\_getusertable作数据源建数据窗口对象并预览数据。

1. 磁盘柜上有14块73G的磁盘，  数据库为200G  大小包括日志文件，如何设置磁盘（要说明这14磁盘是怎么用的）？
2. 有两服务器群集，分别为node1和node2  现在要打win200系统补丁，打完后，要重新启动，如何打补丁，不能影响用户使用（要用群集的术语详细说明）。

3.有一个A  数据库，分别复制到B和C    B  要求  每次数据更新  也同时更新，C  每天更新一次就行，如何制定复制策略!

4.有一个order  表，有90个字段，20个索引，15个复合索引，其中有3个索引字段超过10个，如何进行优化

5.有一个数据库200G大小，每天增加50M  允许用户随时访问，制定备份策略（详细说明）。

6.管理50台数据库，日常工作是检查数据库作业是否完成，你该如何完成这项检查工作？

7.自定义函数和存储过程的区别是什么，什么情况下只能用自定义函数，什么情况下只能用存储过程

数据库面试测试题(SQL Server)  
题目1  
  
　　问题描述:  
  
　　为管理岗位业务培训信息，建立3个表:  
  
　　S (S#,SN,SD,SA) S#,SN,SD,SA 分别代表学号、学员姓名、所属单位、学员年龄  
  
　　C (C#,CN ) C#,CN 分别代表课程编号、课程名称  
  
　　SC ( S#,C#,G ) S#,C#,G 分别代表学号、所选修的课程编号、学习成绩  
  
　　1. 使用标准SQL嵌套语句查询选修课程名称为’税收基础’的学员学号和姓名  
  
　　--实现代码:  
  
　　Select SN,SD FROM S  
  
　　Where [S#] IN(  
  
　　Select [S#] FROM C,SC  
  
　　Where C.[C#]=SC.[C#]  
  
　　AND CN=N'税收基础')  
  
　　2. 使用标准SQL嵌套语句查询选修课程编号为’C2’的学员姓名和所属单位  
  
　　--实现代码:  
  
　　Select S.SN,S.SD FROM S,SC  
  
　　Where S.[S#]=SC.[S#]  
  
　　AND SC.[C#]='C2'  
  
　　3. 使用标准SQL嵌套语句查询不选修课程编号为’C5’的学员姓名和所属单位  
  
　　--实现代码:  
  
　　Select SN,SD FROM S  
  
　　Where [S#] NOT IN(  
  
　　Select [S#] FROM SC  
  
　　Where [C#]='C5')  
  
　　4. 使用标准SQL嵌套语句查询选修全部课程的学员姓名和所属单位  
  
　　--实现代码:  
  
　　Select SN,SD FROM S  
  
　　Where [S#] IN(  
  
　　Select [S#] FROM SC  
  
　　RIGHT JOIN  
  
　　C ON SC.[C#]=C.[C#] GROUP BY [S#]  
  
　　HAVING COUNT(\*)=COUNT([S#]))  
  
　　5. 查询选修了课程的学员人数  
  
　　--实现代码:  
  
　　Select 学员人数=COUNT(DISTINCT [S#]) FROM SC  
  
　　6. 查询选修课程超过5门的学员学号和所属单位  
  
　　--实现代码:  
  
　　Select SN,SD FROM S  
  
　　Where [S#] IN(  
  
　　Select [S#] FROM SC  
  
　　GROUP BY [S#]  
  
　　HAVING COUNT(DISTINCT [C#])>5)  
  
题目2  
  
　　问题描述:  
  
　　已知关系模式:  
  
　　S (SNO,SNAME) 学生关系。SNO 为学号，SNAME 为姓名  
  
　　C (CNO,CNAME,CTEACHER) 课程关系。CNO 为课程号，CNAME 为课程名，CTEACHER 为任课教师  
  
　　SC(SNO,CNO,SCGRADE) 选课关系。SCGRADE 为成绩  
  
　　1. 找出没有选修过“李明”老师讲授课程的所有学生姓名  
  
　　--实现代码:  
  
　　Select SNAME FROM S  
  
　　Where NOT EXISTS(  
  
　　Select \* FROM SC,C  
  
　　Where SC.CNO=C.CNO  
  
　　AND CNAME='李明'  
  
　　AND SC.SNO=S.SNO)  
  
　　2. 列出有二门以上(含两门)不及格课程的学生姓名及其平均成绩  
  
　　--实现代码:  
  
　　Select S.SNO,S.SNAME,AVG\_SCGRADE=AVG(SC.SCGRADE)  
  
　　FROM S,SC,(  
  
　　Select SNO  
  
　　FROM SC  
  
　　Where SCGRADE<60  
  
　　GROUP BY SNO  
  
　　HAVING COUNT(DISTINCT CNO)>=2  
  
　　)A Where S.SNO=A.SNO AND SC.SNO=A.SNO  
  
　　GROUP BY S.SNO,S.SNAME  
  
　　3. 列出既学过“1”号课程，又学过“2”号课程的所有学生姓名  
  
　　--实现代码:  
  
　　Select S.SNO,S.SNAME  
  
　　FROM S,(  
  
　　Select SC.SNO  
  
　　FROM SC,C  
  
　　Where SC.CNO=C.CNO  
  
　　AND C.CNAME IN('1','2')  
  
　　GROUP BY SNO  
  
　　HAVING COUNT(DISTINCT CNO)=2  
  
　　)SC Where S.SNO=SC.SNO  
  
　　4. 列出“1”号课成绩比“2”号同学该门课成绩高的所有学生的学号  
  
　　--实现代码:  
  
　　Select S.SNO,S.SNAME  
  
　　FROM S,(  
  
　　Select SC1.SNO  
  
　　FROM SC SC1,C C1,SC SC2,C C2  
  
　　Where SC1.CNO=C1.CNO AND C1.NAME='1'  
  
　　AND SC2.CNO=C2.CNO AND C2.NAME='2'  
  
　　AND SC1.SCGRADE>SC2.SCGRADE  
  
　　)SC Where S.SNO=SC.SNO  
  
　　5. 列出“1”号课成绩比“2”号课成绩高的所有学生的学号及其“1”号课和“2”号课的成绩  
  
　　--实现代码:  
  
　　Select S.SNO,S.SNAME,SC.[1号课成绩],SC.[2号课成绩]  
  
　　FROM S,(  
  
　　Select SC1.SNO,[1号课成绩]=SC1.SCGRADE,[2号课成绩]=SC2.SCGRADE  
  
　　FROM SC SC1,C C1,SC SC2,C C2  
  
　　Where SC1.CNO=C1.CNO AND C1.NAME='1'  
  
　　AND SC2.CNO=C2.CNO AND C2.NAME='2'  
  
　　AND SC1.SCGRADE>SC2.SCGRADE  
  
　　)SC Where S.SNO=SC.SNO

1．实现双向链表删除一个节点P，在节点P后插入一个节点，写出这两个函数。

2．写一个函数，将其中的\t都转换成4个空格。

3．Windows程序的入口是哪里？写出Windows消息机制的流程。

4．如何定义和实现一个类的成员函数为回调函数？

5．C++里面是不是所有的动作都是main()引起的？如果不是，请举例。

6．C++里面如何声明const void f(void)函数为C程序中的库函数？

7．下列哪两个是等同的

int b;

A const int\* a = &amp;b;

B const\* int a = &amp;b;

C const int\* const a = &amp;b;

D int const\* const a = &amp;b;

8．内联函数在编译时是否做参数类型检查？

一、请填写BOOL , float, 指针变量 与“零值”比较的 if 语句。（10分）

提示：这里“零值”可以是0, 0.0 , FALSE或者“空指针”。例如 int 变量 n

与“零值”比较的 if 语句为：

if ( n == 0 )

if ( n != 0 )

以此类推。

请写出 BOOL flag 与“零值”比较的 if 语句：

请写出 float x 与“零值”比较的 if 语句：

请写出 char \*p 与“零值”比较的 if 语句：

二、以下为Windows NT下的32位C++程序，请计算sizeof的值（10分）

char str[] = “Hello” ;

char \*p = str ;

int n = 10;

请计算

sizeof (str ) =

sizeof ( p ) =

sizeof ( n ) = void Func ( char str[100])

{

请计算

sizeof( str ) =

}

void \*p = malloc( 100 );

请计算

sizeof ( p ) =

三、简答题（25分）

1、头文件中的 ifndef/define/endif 干什么用？

2、#include 和 #include “filename.h” 有什么区别？

3、const 有什么用途？（请至少说明两种）

4、在C++ 程序中调用被 C编译器编译后的函数，为什么要加 extern “C”声明？

5、请简述以下两个for循环的优缺点

// 第一个

for (i=0; i

{

if (condition)

DoSomething();

else

DoOtherthing();

} // 第二个

if (condition)

{

for (i=0; i

DoSomething();

}

else

{

for (i=0; i

DoOtherthing();

}

优点：

缺点：

优点：

缺点：

四、有关内存的思考题（20分）

void GetMemory(char \*p)

{

p = (char \*)malloc(100);

}

void Test(void)

{

char \*str = NULL;

GetMemory(str);

strcpy(str, "hello world");

printf(str);

}

请问运行Test函数会有什么样的结果？

答：

char \*GetMemory(void)

{

char p[] = "hello world";

return p;

}

void Test(void)

{

char \*str = NULL;

str = GetMemory();

printf(str);

}

请问运行Test函数会有什么样的结果？

答：

Void GetMemory2(char \*\*p, int num)

{

\*p = (char \*)malloc(num);

}

void Test(void)

{

char \*str = NULL;

GetMemory(&str, 100);

strcpy(str, "hello");

printf(str);

}

请问运行Test函数会有什么样的结果？

答：

void Test(void)

{

char \*str = (char \*) malloc(100);

strcpy(str, “hello”);

free(str);

if(str != NULL)

{

strcpy(str, “world”);

printf(str);

}

}

请问运行Test函数会有什么样的结果？

答：

五、编写strcpy函数（10分）

已知strcpy函数的原型是

char \*strcpy(char \*strDest, const char \*strSrc);

其中strDest是目的字符串，strSrc是源字符串。

（1）不调用C++/C的字符串库函数，请编写函数 strcpy

（2）strcpy能把strSrc的内容复制到strDest，为什么还要char \* 类型的返回值？

六、编写类String的构造函数、析构函数和赋值函数（25分）

已知类String的原型为：

class String

{

public:

String(const char \*str = NULL); // 普通构造函数

String(const String &other); // 拷贝构造函数

~ String(void); // 析构函数

String & operate =(const String &other); // 赋值函数

private:

char \*m\_data; // 用于保存字符串

};

请编写String的上述4个函数。

Sony笔试题

　　1．完成下列程序

　　\*

　　\*.\*.

　　\*..\*..\*..

　　\*...\*...\*...\*...

　　\*....\*....\*....\*....\*....

　　\*.....\*.....\*.....\*.....\*.....\*.....

　　\*......\*......\*......\*......\*......\*......\*......

　　\*.......\*.......\*.......\*.......\*.......\*.......\*.......\*.......

　　#include <stdio.h>

　　#define N 8

　　int main()

　　{

　　 int i;

　　 int j;

　　 int k;

　　 ---------------------------------------------------------

||

　　 ---------------------------------------------------------

　　 return 0;

　　}

　　2．完成程序，实现对数组的降序排序

　　#include <stdio.h>

　　void sort( );

　　int main()

　　{

　　 int array[]={45，56，76，234，1，34，23，2，3}； //数字任//意给出

　　 sort( );

　　 return 0;

　　}

　　void sort( )

　　{

　　 ------------------------------------------------------

　　 ||

　　 ------------------------------------------------------

　　}

　　3．费波那其数列，1，1，2，3，5……编写程序求第十项。可以用递归，也可以用其他方法，但要说明你选择的理由。

　　#include <stdio.h>

　　int Pheponatch(int);

　　int main()

　　{

　　 printf("The 10th is %d",Pheponatch(10));

　　 return 0;

　　}

　　int Pheponatch(int N)

　　{

　　--------------------------------

　　||

　　--------------------------------

　　}

　　4．下列程序运行时会崩溃，请找出错误并改正，并且说明原因。

　　#include <stdio.h>

　　#include <malloc.h>

　　typedef struct{

　　 TNode\* left;

　　 TNode\* right;

　　 int value;

　　} TNode;

　　TNode\* root=NULL;

　　void append(int N);

　　int main()

　　{

　　 append(63);

　　 append(45);

　　 append(32);

　　 append(77);

　　 append(96);

　　 append(21);

　　 append(17); // Again, 数字任意给出

　　}

　　void append(int N)

　　{

　　 TNode\* NewNode=(TNode \*)malloc(sizeof(TNode));

　　 NewNode->value=N;

　　 if(root==NULL)

　　 {

　　 root=NewNode;

　　 return;

　　 }

　　 else

　　 {

　　 TNode\* temp;

　　 temp=root;

　　 while((N>=temp.value && temp.left!=NULL) || (N<temp. value && temp. right !=NULL))

　　 {

　　 while(N>=temp.value && temp.left!=NULL)

　　 temp=temp.left;

　　 while(N<temp.value && temp.right!=NULL)

　　 temp=temp.right;

　　 }

　　 if(N>=temp.value)

　　 temp.left=NewNode;

　　 else

　　 temp.right=NewNode;

　　 return;

　　 }

　　}

华为笔试题

　　1．请你分别画出OSI的七层网络结构图和TCP/IP的五层结构图。

　　2．请你详细地解释一下IP协议的定义，在哪个层上面？主要有什么作用？TCP与UDP呢 ？

　　3．请问交换机和路由器各自的实现原理是什么？分别在哪个层次上面实现的？

　　4．请问C++的类和C里面的struct有什么区别？

　　5．请讲一讲析构函数和虚函数的用法和作用。

　　6．全局变量和局部变量有什么区别？是怎么实现的？操作系统和编译器是怎么知道的 ？

　　7．8086是多少位的系统？在数据总线上是怎么实现的？

联想笔试题

　　1．设计函数 int atoi(char \*s)。

　　2．int i=(j=4,k=8,l=16,m=32); printf(“%d”, i); 输出是多少？

　　3．解释局部变量、全局变量和静态变量的含义。

　　4．解释堆和栈的区别。

　　5．论述含参数的宏与函数的优缺点。

普天C++笔试题

　　1．实现双向链表删除一个节点P，在节点P后插入一个节点，写出这两个函数。

　　2．写一个函数，将其中的\t都转换成4个空格。

　　3．Windows程序的入口是哪里？写出Windows消息机制的流程。

　　4．如何定义和实现一个类的成员函数为回调函数？

　　5．C++里面是不是所有的动作都是main()引起的？如果不是，请举例。

　　6．C++里面如何声明const void f(void)函数为C程序中的库函数？

　　7．下列哪两个是等同的

　　int b;

　　A const int\* a = &b;

　　B const\* int a = &b;

　　C const int\* const a = &b;

　　D int const\* const a = &b;

　　8．内联函数在编译时是否做参数类型检查？

　　void g(base & b){

　　 b.play;

　　}

　　void main(){

　　 son s;

　　 g(s);

　　 return;

　　}

大唐电信-DTT笔试题

　　考试时间一小时，第一部分是填空和选择：

　　1．数列6，10，18，32，“？”，问“？”是几？

　　2．某人出70买进一个x，80卖出，90买回，100卖出，这桩买卖怎么样？

　　3．月球绕地球一圈，至少要多少时间？

　　4．7个人用7小时挖了7米的沟，以同样的速度在50小时挖50米的沟要多少人？

　　5．鱼头长9，鱼尾等于鱼头加半个鱼身，鱼身等于鱼头加鱼尾，问鱼全长多少？

　　6．一个小姐买了一块手表，回家发现手表比她家的表慢了两分钟，晚上看新闻的时候又发现她家的表比新闻里的时间慢了两分钟，则 。

　　A 手表和新闻里的时间一样

　　B 手表比新闻里的时间慢

　　C 手表比新闻里的时间快

　　7．王先生看到一则招聘启事，发现两个公司除了以下条件不同外，其他条件都相同

　　A 半年年薪50万，每半年涨5万

　　B 一年年薪100万，每一年涨20万

　　王先生想去一家待遇比较优厚的公司，他会去哪家？

　　10．问哪个袋子里有金子？

　　A袋子上的标签是这样写的：B袋子上的话是对的，金子在A袋子。

　　B袋子上的标签是这样写的：A袋子上的话是错的，金子在A袋子里。

　　11．3个人住酒店30块钱，经理找回5块钱，服务生从中藏了2块钱，找给每人1块钱，3×（10&#61485;1）+2=29，问这是怎么回事？

　　12．三篇写作，均为书信形式。

　　（1）一片中文的祝贺信，祝贺某男当了某公司xx

　　（2）两篇英文的，一是说有事不能应邀，派别人去；另一篇是讨债的，7天不给钱就走人（主要考business letter格式）。

大唐面试试题

　　1．什么是中断？中断发生时CPU做什么工作？

　　2．CPU在上电后，进入操作系统的main()之前必须做什么工作？

　　3．简述ISO OSI的物理层Layer1，链路层Layer2，网络层Layer3的任务。

　　4．有线电话和无线电话有何区别？无线电话特别需要注意的是什么？

　　5．软件开发五个主要step是什么？

　　6．你在开发软件的时候，这5个step分别占用的时间百分比是多少？

　　7．makefile文件的作用是什么？

　　8．UNIX显示文件夹中，文件名的命令是什么？能使文件内容显示在屏幕的命令是什么 ？

　　9．（选做）手机用户在从一个基站漫游到另一个基站的过程中，都会发生什么?

网通笔试题

　　选择题（每题5分，只有一个正确答案）

　　1．中国1号信令协议属于 的协议。

　　A ccs B cas C ip D atm

　　2．isdnpri协议全称是 。

　　A 综合业务模拟网基速协议

　　B 综合业务模拟网模拟协议

　　C 综合业务数字网基率协议

　　D 综合业务数字网基次协议

　　3．路由协议中， 协议是用距离作为向量的。

　　A ospf B bgp C is-is D rip

　　4．中国智能网中，ssp与scp间最上层的ss7协议是 。

　　A incs B is41b C is41c D inap

　　5．dtmf全称是 。

　　A 双音多频 B多音双频 C多音三频 D三音多频

　　6．计算机的基本组成部分中，不包含下面设备的是 。

　　A cpu B输入设备 C存储器 D接口

　　7．脉冲编码调制的简称是 。

　　A pcm B pam C (delta)M D atm

　　8．普通电话线接口专业称呼是 。

　　A rj11 B rj45 C rs232 D bnc

　　9．现有的公共数据网都采用 。

　　A电路交换技术 B报文交换技术

　　C语音插空 D分组交换

　　10．ss7协议中的制止市忙消息简写为 。

　　A stb B slb C sub D spb

　　简答题（每题10分）

　　1．简述普通电话与IP电话的区别。

　　2．简述随路信令与公路信令的根本区别。

　　3．说明掩码的主要作用。

　　4．ss7协议中，有三大要素决定其具体定位，哪三大要素？

　　5．描述ss7的基本通话过程。

　　6．简述通信网的组成结构。

　　7．面向连接与面向非连接各有何利弊？

　　8．写出爱尔兰的基本计算公式。

　　9．数据网主要有哪些设备？

　　10．中国一号协议是如何在被叫号码中插入主叫号码的？

东信笔试题目

　　笔试：30分钟。

　　1．压控振荡器的英文缩写。

　　2．动态随机存储器的英文缩写。

　　3．选择电阻时要考虑什么？

　　4．单片机上电后没有运转，首先要检查什么？

　　5．计算机的基本组成部分及其各自的作用。

　　6．怎样用D触发器、与或非门组成二分频电路？

中软融鑫笔试题

　　1．关于工作

　　（1） 你对未来的工作生活是怎样憧憬的？为何选择我公司作为求职公司？

　　（2）请用不超过30个字给出一个最能让我们录用你的理由。

　　（3）你认为比较理想的工作环境是怎样的？

　　（4）你个人的中长期的职业发展目标是怎样的？

　　2．关于社会

　　（1）如果你是杨利伟，你在太空中向祖国人民说的第一句话是什么？

　　（2）宋美龄女士于2003年10月谢世，对这位著名人士在西安事变中的态度和作用，你是如何看待的？（不超过300字）

　　（3）北京政府颁布的对拾金不昧者，失主要奖励相当于财产20%奖金的公告，你是如何看的？

　　（4）如果给你50万元人民币，你将会用这些钱做什么？

　　（5）在美国，男、女卫生间（厕所）的正确称呼为什么？请用英语写出答案。

　　（6）你认为麦当劳是世界最大的汉堡生产商吗？如果不是，请说出你的观点。

　　3．教育背景

　　（1）你受过哪些正规的教育或培训？（自高中毕业起）

　　（2）在校期间进行过哪些社会活动？

Delphi笔试题目

　　机械类笔试试题

　　1. Briefly describe what is blanking(cutting), forming, coining and embossing in stamping process.

　　2. What is metal clading?

　　3. What is the purpose of adding glass fiber to thermoplastic material?

　　4. In contrast with metal and thermoplastic material,which has a higher coefficient of thermal expansion(CTE).

　　5. The most suitable material for a integral hinge design (typical plastic thickness=0.25 to 0.5mm at hinge)

　　6. Can a bending load makes both compressive and tensile stress in a member?

　　7. What is the design criteria used in plastics catch/snap?

　　8. What is FEA?

　　9. Why is natural frequency important in vibration analysis?

　　10. What is the deflection equation of a cantilever beam fixed at one edge?

　　EE笔试试题

　　1. Name 3 Vehicle Buses.

　　2. Name 2 possible sources of Electromagnetic interference on Electronics Circuit ASM.

　　3. Wavelength for 12MHz frequency signal is\_\_\_\_

　　4. Name 2 important considerations for car radio performance related to audio signal processing under multipath condition

　　5. What is the typical FM receiver RF signal strength to achieve 30dB S/N for car radio?

　　6. When a radio is tuned to 98.1 MHz & with a LO of 108.8 MHz, what is the image frequency?

　　7. For a system with a matched impedance, what is the Reflection Coefficient and SWR?

　　8. Which property of the output capacitor is the primary cause of Low Drop Out(LDO) regulator loop instability?

　　（1）Equivalent series resistance(ESR)

　　（2）Effective series inductance(ESL)

　　（3）Capacitance value

　　（4）Dielectric material

　　9. The switching regulator is capable of:

　　（1）Higher power conversion efficiency

　　（2）Providing an output voltage that is higher than the input

　　（3）Generating an output boltage oppsite in polarity to the input

　　（4）All of the above

　　10. A linear regulator op Vin(max) = 10v, Vout(min) = 4.8v, Iout(max) = 2.5mA, Iq(max) = 2.5mA, Ta(max) = 8.5摄氏度，The regulator is available in 3 packages.Each package has the following thermal characteristics:

　　Package Rja（摄氏度/W） Rjc（摄氏度/W）

　　SO14 125 30

　　D1P8 100 52

　　Choose the most suitable package to handle the power dissipation requirement without a heat sink and why.

　　软件笔试题

　　1. How do you code an infinite loop in C?

　　2. Volatile:

　　（1）What does the keyword volatile mean? Give an example

　　（2）Can a parameter be both const and volatile? Give an example

　　（3）Can a pointer be volatile? Give an example

　　3. What are the values of a, b, and c after the following instructions:

　　int a=5, b=7, c;

　　c = a+++b;

　　4. What do the following declarations mean?

　　（1）const int a;

　　（2）int const a;

　　（3）const int \*a;

　　（4）int \* const a;

　　（5）int const \* a const;

　　5. Which of the following statements describe the use of the keyword static?

　　（1）Within the body of a function: A static variable maintains its value between function revocations

　　（2）Within a module: A static variable is accessible by all functions within that module

　　（3）Within a module: A static function can only be called by other functins within that module

　　6. Embedded systems always require the user to manipulate bits in registers or variables. Given an integer variable a, write two code fragments.

　　The first should set bit 5 of a. The second shnuld clear bit 5 of a. In both cases, the remaining bits should be unmodified.

　　7. What does the following function return?

　　char foo(void)

　　{

　　 unsigned int a = 6;

　　 iht b = -20;

　　 char c;

　　 (a+b > 6) ? (c=1): (c=0);

　　 return c;

　　}

　　8. What will be the output of the following C code?

　　main()

　　{

　　 int k, num= 30;

　　 k =(num > 5 ? (num <=10 ? 100:200): 500);

　　 printf(“%d”, k);

　　}

　　9. What will the following C code do?

　　int \*ptr;

　　ptr =(int \*)Ox67a9;

　　\*ptr = Oxaa55;

　　10. What will be the output of the follow C code?

　　#define product(x) (x\*x)

　　main()

　　{

　　 int i = 3, j, k;

　　 j = product(i++);

　　 k = product(++i);

　　 printf(“%d %d”,j,k);

　　}

　　11. Simplify the following Boolean expression

　　!((i ==12) || (j > 15))

　　12. How many flip-flop circuits are needed to divide by 16?

　　13. Provides 3 properties that make an OS, a RTOS?

　　14. What is pre-emption?

　　15. Assume the BC register value is 8538H, and the DE register value is 62A5H.Find the value of register BC after the following assembly operations:

　　MOV A,C

　　SUB E

　　MOV C,A

　　MOV A,B

　　SBB D

　　MOV B,A

　　16. In the Assembly code shown below

　　LOOP: MVI C,78H

　　 DCR C

　　 JNZ LOOP

　　 HLT

　　How many times is the DCR C Operation executed?

　　17. Describe the most efficient way (in term of execution time and code size) to divide a number by 4 in assembly language

　　18. what value is stored in m in the following assembly language code fragment if n=7?

　　 LDAA #n

　　LABEL1: CMPA #5

　　 BHI L3

　　 BEQ L2

　　 DECA

　　 BRA L1

　　LABEL2: CLRA

　　LABEL3: STAA #m

　　19. What is the state of a process if a resource is not available?

　　#define a 365\*24\*60\*60

　　20. Using the #define statement, how would you declare a manifest constant that returns the number of seconds in a year? Disregard leap years in your

answer.

　　21. Interrupts are an important part of embedded systems. Consequently, many compiler vendors offer an extension to standard C to support interrupts. Typically, the keyword is \_\_interrupt. The following routine (ISR). Point out

problems in the code.

　　\_\_interrupt double compute\_area (double radius)

　　{

　　 double area = PI \* radius \* radius;

　　 printf(“\nArea = %f”, area);

　　 return area;

　　}

Hongkong Bank笔试题

　　1. Please state why you chose to follow these activities and how they have contributed to your personal development. You may wish to give details of

your role whether anyone else was involved and any difficulties you encountered.

　　2. Please state how you have benefited from your work experience.

　　3. How much is your present monthly salary including allowances.

　　4. Do you need to compensate your present employer if you resign? If so, please give details.

　　5. Other than academic success, what has been your greatest achievement to date? What do you see as your personal strength, why?

　　6. Please state why the position you have applied for is appropriate for you; Why you have selected HongKong Bank and what your career objectives are.

A.T. Keaney笔试题

　　1. Describe your greatest achievement in the past 4-5 years?

　　2. What are your short-term and long-term career objectives? What do you think is the most ideal job for you?

　　3. Why do you want to join A.T kearney? What do you think you can contribute to A.T kearney?

　　4. Why are you applying for a position at Arthur Anderson?

　　5. What are your expectations of our firm.

　　6. Describe your hobbies and interests.

Shell company笔试题

　　1. How wold your colleagues/classmates describe you in five words? On what evidence would they base this assessment.

　　2. If you are asked to recruit the best graduates for shell, what would you do to attract them? What would you do to select them?

　　3. Please describe a new activity that you have initiated and implemented. Please highlight your role out.

　　4. Please describe your outstanding non-academic achieve- ments.

　　5. Please describe any other significant activities you have been involved in including organizing people.

　　6. Imagine that Shell has found oil in an inland province of China, near a large river. You are responsible for planning how to transport the oil to

the coast thousands of miles away. What are the main issue you would consider, and what would you do?

KPMG笔试题

　　“The big economic difference between nuclear and fossil-fuelled power stations is that nuclear reactors are more expensive to build and decommission, but cheaper to sun. So disputes over the relative efficiency of the two

systems revolve not just around prices of coal and uranium today and tomorrow, but also around the way in which future income should be compared with current

income.”

　　1. The main difference between nuclear and fossil-fuelled power stations is an economic one.

　　TRUE

　　UNTRUE

　　CANNOT SAY

　　2. The price of coal is not relevant to discussions about the relative efficiency of nuclear reactors.

　　TRUE

　　UNTRUE

　　CANNOT SAY

　　3. If nuclear reactors were cheaper to build and decommission than fossil-fuelled power stations, they would definitely have the economic advantage.

　　TRUE

　　UNTRUE

　　CANNOT SAY　“At any given moment we are being bombarded by physical and psychological stimuli competing for our attention. Although our eyes are capable of

handling more than 5 million bits of data per second, our brain are capable of interpreting only about 500 bits per second. With similar disparities between each of the other senses and the brain, it is easy to see that we must select the vi sual, auditory, or tactile stimuli that we wish to compute at any specific time.”

　　4. Physical stimuli usually win in the competition for our attention.

　　TRUE

　　UNTRUE

　　CANNOT SAY

　　5. The capacity of the human brain is sufficient to interpret nearly all the stimuli the senses can register under optimum conditions.

　　TRUE

　　UNTRUE

　　CANNOT SAY

　　6. Eyes are able to cope with a greater input of information than ears.

　　TRUE

　　UNTRUE

　　CANNOT SAY

　　VERBAL ANSWER：

　　（1）C CANNOT SAY

　　（2）B UNTRUE

　　（3）A TRUE

　　（4）C CANNOT SAY

　　（5）B UNTRUE

　　（6）C CANNOT SAY

　　PartII NUMERCAL TEST

　　1．Which country had the highest number of people aged 60 or over at the start of 1985?

　　A. UK

　　B. France

　　C. Italy

　　D. W.Germany

　　E. Spain

　　2．What percentage of the total 15mm button production was classed as sub-standard in September?

　　AA 10.5% BB 13% CC 15% DD 17.5% EE 20% AB 23.5% AC 25%

　　AD 27.5% AE 28% BC 30.5%

　　3. How many live births occurred in 1985 in Spain and Italy together (to the nearest 1000)?

　　A. 104 000

　　B. 840 000

　　C. 1 044 000

　　D. 8 400 000

　　E. 10 440 000

　　4. What was the net effect on the UK population of the live birth and death rates in 1985?

　　A. Decrease of 66 700

　　B. Increase of 752 780

　　C. Increase of 84 900

　　D. Cannot Say

　　E. Increase of 85 270

　　5. By how much did the total sales value of November‘s button production vary from October‘s?

　　A. 8.50 (Decrease)

　　B. 42.50 (Decrease)

　　C. 85.00 (Increase)

　　D. 27.50 (Decrease)

　　E. No change

　　6. What was the loss in potential sales revenue attributable to the production of sub-standard (as opposed to standard) buttons over the 6 month period?

　　A. 13.75

　　B. 27.50

　　C. 137.50

　　D. 280.00

　　E. 275.00

香港电信笔试题

　　1. Based on your understanding of the following java related technologies:

servlets, JavaServerPage, JavaBeans, Enterprise JavaBeans, how do you think these technologies are work together or are applied in the development of an

in ternet-based application (25marks).

　　2. In your opinion ,what do you think are the advantages or benefitsof

using an object-oriented approach to software development? how do you think those benefits can be achieved or realized? (15marks).

　　3. In designing your classes, given the choice between inheritance and aggregation which do you choose (15marks).

　　4. How would you work around the lack of multiple inheritance feature in Java (15marks).

　　5. What would you consider to be the hardest part of OO analysis and design and why (10marks).

　　6. How do you keep yourself up to date with the latest in software techonogy, especially in the field of software development (10marks).

　　7. What si your career aspiration? Why do you think this E-Commerce Development Center can help you in achieving your career goals (10marks) (1hr, answer in English).

　　笔试题

　　1. Would you please describe yourself in 3-4 lines? (limited in 500 words)

　　2. Could you tell us why we should choose you as a Loreal Person, and what makes you unique? (limited in 500 words)

　　3. What is your short-term and long-term career plan? (limited in 500 words)

　　4. What kind of group activities are you interested in and what type of role do you often play? (limited in 500 words)

　　5. Please use one sentence to give a definition of ‘Beauty’, and describe the most beautiful thing in your life. (limited in 500 words)

维尔VERITAS软件笔试题

　　1. A class B network on the internet has a subnet mask of 255.255.240.0, what is the maximum number of hosts per subnet .

　　a. 240 b. 255 c. 4094 d. 65534

　　2. What is the difference: between o(log n) and o(log n^2), where both log arithems have base 2 .

　　a. o(log n^2) is bigger b. o(log n) is bigger

　　c. no difference

　　3. For a class what would happen if we call a class’s constructor from with the same class’s constructor .

　　a. compilation error b. linking error

　　c. stack overflow d. none of the above

　　4. “new” in c++ is a: .

　　a. library function like malloc in c

　　b. key word c. operator

　　d. none of the above

　　5. Which of the following information is not contained in an inode .

　　a. file owner b. file size

　　c. file name d. disk address

　　6. What’s the number of comparisons in the worst case to merge two sorted lists containing n elements each .

　　a. 2n b.2n-1 c.2n+1 d.2n-2

　　7. Time complexity of n algorithm T(n), where n is the input size ,is T(n) =T(n-1)+1/n if n>1 otherwise 1the order of this algorithm is .

　　a. log (n) b. n c. n^2 d. n^n

　　8. The number of 1’s in the binary representation of 3\*4096+ 15\*256+5\*16+3 are .

　　a. 8 b. 9 c. 10 d. 12

百威啤酒(武汉公司)

　　1,为什么申请来百威?

　　2,将来有什么打算?

　　3,有没有社会活动经历?

　　4,有没有当众演讲的经历?

　　5,经常使用那些软件?

　　6,喜欢哪些课程?

　　7,你认为工作中的什么因素对你来说最重要?

　　8,什么时候可以来上班?可以在这里工作多久?

　　9,八点上班,要加班和出差,能不能做到?

星巴克

　　1、 您是一家咖啡店的店经理，你发现店内同时出现下列状况：

　　　1）许多张桌子桌面上有客人离去后留下的空杯未清理，桌面不干净待整理。

　　　2）有客人正在询问店内卖哪些品种，他不知如何点咖啡菜单。

　　　3）已有客人点完成咖啡，正在收银机旁等待结帐。

　　　4）有厂商正准备要进货，需要店经理签收。

　　　　请问，针对上述同时发生的情况，你要如何排定处理之先后顺序，为什么

　　2、 有一位甲员工脾气不好以致在前三家店因为与店内其他同事相处不佳而屡屡调动，现在甲被调到你的店里面来，请问身为店经理的你，将如何应对？？

　　3、 你是店经理，本周五结帐后，发现门市总销售额较上周五减少30%，请问可能原因会是哪几种，各原因如何应对？

凹凸电子软件笔试题

　　1. Select ONE of the following projects to discuss: 　　a. Signal Filtering: You are given a sampled realtime waveform consisting of a sensor reading mixed with highly periodic impulses and high frequency

noise.The desired output is the realtime filtered sensor signal with the impulses and noise removed, and a readout of the impulse period. The FFT may not be

used.

　　b. Interrupt Processing.A headware register consisting of eight independent edge triggered latches is used to record external asynchronous interrupt

requests. When any of the request bits are latched, a software interrupt is generated. The software may read the latch to see which interrupt(s) occurred.

Writing a one to any latch bit will clear the latch. How does that software assure that no interrupt request is ever missed?

　　c. User Interface: a prototype MP3 player interface consisting of a playlist display and a few control buttons is given to you. How would you make the interface “skinnable”,with user selected graphics, options, and control

button placement?

　　Each project description is incomplete. What questions would you ask to completely specify the project? What development tools would you prefer to use?

What algorithm /data structures/design would you use?

　　2. What program(s) have you coded for you own enjoyment (not part of a school project,not for pay). What type of software project would you most enjoy

working on?

　　3. Have you participated in a team programming project? What is the hardest part of programming as a team, as opposed to programming alone?

友立资讯笔试题目

　　1．一堆鸡蛋，3个3个数剩余2个，5个5个数剩余1个，7个7个数剩余3个，问这堆鸡蛋

最少有多少个？并给出通解。

　　2．列举五岳，及其所在省份。

　　3．何为四书。

　　4．按顺序默写24节气。

　　5．默写于谦的《吟石灰》。

　　6．英语翻译约300字。

　　7．作文一篇：求职有感。

普华永道PWC笔试题目（作文）

　　1．最近10年来中国媒体的变化。

　　2．你认为发展汽车产业和公共交通哪个更重要？

　　3．如何理解风险投资？

　　4．如何理解广告的消极作用和积极作用？

Avant! 微电子EE笔试题

　　1．名词解释：VLSI，CMOS，EDA，VHDL，Verilog，HDL，ROM，RAM，DRC，LVS。

　　2．简述CMOS工艺流程。

　　3．画出CMOS与非门的电路，并画出波形图简述其功能。

　　4．画出N沟道增强型MOSFET的剖面图。

　　5．简述ESD和latch-up的含义。

　　6．简述三极管与MOS管的区别。

　　7．简述MOORE模型和MEALY模型。

　　8．简述堆栈与队列的区别。

　　奇码数字信息有限公司笔试题

　　1．画出NMOS的特性曲线（指明饱和区，截至区，线性区，击穿区和C-V曲线）

　　2．2.2um工艺下，Kn＝3Kp，设计一个反相器，说出器件尺寸。

　　3．说出制作N-well的工艺流程。

　　4．雪崩击穿和齐纳击穿的机理和区别。

　　5．用CMOS画一个D触发器（clk，d，q，q-）。

德勤笔试题

　　五个人来自不同地方，住不同房子，养不同动物，吸不同牌子香烟，喝不同饮料，喜

欢不同食物。根据以下线索确定谁是养猫的人。

　　（1）红房子在蓝房子的右边，白房子的左边（不一定紧邻）

　　（2）黄房子的主人来自香港，而且他的房子不在最左边。

　　（3）爱吃比萨饼的人住在爱喝矿泉水的人的隔壁。

　　（4）来自北京的人爱喝茅台，住在来自上海的人的隔壁。

　　（5）吸希尔顿香烟的人住在养马的人右边隔壁。

　　（6）爱喝啤酒的人也爱吃鸡。

　　（7）绿房子的人养狗。

　　（8）爱吃面条的人住在养蛇的人的隔壁。

　　（9）来自天津的人的邻居（紧邻）一个爱吃牛肉，另一个来自 成都。

　　（10）养鱼的人住在最右边的房子里。

　　（11）吸万宝路香烟的人住在吸希尔顿香烟的人和吸“555”香烟的人的中间（紧邻）

　　（12）红房子的人爱喝茶。

　　（13）爱喝葡萄酒的人住在爱吃豆腐的人的右边隔壁。

　　（14）吸红塔山香烟的人既不住在吸健牌香烟的人的隔壁，也不与来自上海的人相邻 。

　　（15）来自上海的人住在左数第二间房子里。

　　（16）爱喝矿泉水的人住在最中间的房子里。

　　（17）爱吃面条的人也爱喝葡萄酒。

　　（18）吸“555”香烟的人比吸希尔顿香烟的人住的\*右。

扬智（科技）笔试题目

　　软件题目

　　1. Queue is a useful structure

　　\* What is a queue?

　　\* Write 5 operations or functions, without details, that can be done on a queue.

　　2. Insert a sequence fo keys(24,49,13,20,59,23,90,35) into a data structure, which has no keys initially. Depict the data structure after these insertions, if it is:

　　\* a heap tree

　　\* an AVL tree

　　3. \* What is a synchronous I/O bus?

　　\* What is an asnchronous I/O bus?

　　\* Compare the advantages and disadvantages of synchronous and a synchronous I/O bus.

　　4. Explain the following terminology:

　　\* Baud rate

　　\* Handshaking

　　\* Memory mapped I/O

　　5. Explain the key issues in supporting a real-time operation system for embedded system.

　　6. Explain the mapping of visual addresses to real addresses under paging by

　　\* direct mapping

　　\* associative mapping

　　\* combined direct/associated mapping

　　7. Please explain what is “write-back” and “write-through”, and discuss the advantage and disadvantage about these two methods.

　　8. Explain the concept and benefit of threads

　　9. What is hardware interrupt? What is software interrupt? What is exception? Please tell me all you know about interrupt.

　　10. Write a recursive function that tests wether a string is a palindrome. A palindrome is s string such as “abcba” or “otto” that reads the same in both directions.If you can write this function recursively,you can write an iterative version of this function instead.

　　11．什么是进程（Process）和线程（Thread）？有何区别？

　　12．MFC和SDK有何区别？

　　13．IRP是什么？有何作用？

　　14．Windows 2000操作系统下用户模式和内核模式下编程有何区别？

　　15．驱动程序的BUFFER能swap到磁盘上去吗？为什么？

　　16．试编写3个函数实现

　　（1）建立一个双向链表

　　（2）插入一个节点

　　（3）删除一个节点

　　17．简述Hardware interrupt和software中断的区别，简述其应用。

　　18．试编写一个函数，计算一个字符串中A的个数。

　　19．画出其相应流程图并编写一个函数实现一个整数到二进制数的转换，如输入6，输出110。

　　20．

　　（1）编写一个递归函数，删除一个目录。

　　（2）编写一个非递归函数，删除一个目录。

　　并比较其性能。

　　21．附加题：简单叙述编程经历

　　硬件题目

　　1．用mos管搭出一个二输入与非门。

　　2．集成电路前段设计流程，写出相关的工具。

　　3．解释名词IRQ，BIOS，USB，VHDL，SDR。

　　4．简述如下Unix命令cp -r, rm,uname。

　　5．用波形表示D触发器的功能。

　　6．写异步D触发器的verilog module。

　　7．What is PC Chipset？

　　8．用传输门和倒向器搭一个边沿触发器。

　　9．画状态机，接受1，2，5分钱的卖报机，每份报纸5分钱。

　　DSP题目

　　1．H（n）&#61501;&#61485;a\*h（n&#61485;1）+b\*δ（n）

　　（1）求h（n）的z变换

　　（2）该系统是否为稳定系统

　　（3）写出FIR数字滤波器的差分方程

　　2．写出下面模拟信号所需的最小采样带宽

　　（1）模拟信号的频率范围是0~4kHz

　　（2）模拟信号的频率范围是2~4kHz

　　3．名词解释

　　（1）量化误差

　　（2）直方图

　　（3）白平衡

　　（4）MMX

　　4．写出下面几种格式中用到的压缩技术

　　（1）JPEG

　　（2）MPEG2

　　（3）MP3

高通笔试题

　　1. Can you describe the trend of wireless mobile communication industry? ( 2000 letters)

　　2. Compare the major third generation technologies.(2000 letters)

　　3. Describe the characteristics of Walsh function. Explain how to generate Walsh Function. (2000 letters)

　　4. List factors that will affect the capacity of forward and reverse links of a CDMA system. (2000 letters)

　　5. What are the differences between IS-95 A/B and cdma2000 1X? (2000 letters)

威盛笔试试题

　　2002年软件笔试题

　　1．三组程序，找出你认为的错误。

　　（1）a.c long temp[255];

　　b.c extern \*temp;

　　（2）a.c long temp[255];

　　b.c extern temp[256];

　　（3）a.c long temp[255];

　　b.c extern temp[];

　　2．在第一个声明处编译出了奇怪的错误，为什么？

　　#include <stdio.h>

　　#include “myfun1.h”

　　#include “myfun2.h”

　　int myInt1;

　　int myInt2;

　　3．printf(“0x%x”, (&0)[&#61485;1]); 请问打印了什么？

　　4．汇编，用ax,bx,cx,dx，求1000×1000/30（四舍五入），结果放在ax中。

　　5．编最优化Bubble(int \*pIntArray,int L)，要求:交换元素不能用临时变量，如果有序需要最优。

　　6．用任意一种编程语言写n!的算法。

　　2003 Asic部分

　　1．一个四级的Mux，其中第二级信号为关键信号，如何改善timing？

　　2．一个状态机的题目用Verilog实现。

　　3．Asic中的design flow的实现。

　　4．用逻辑门画出D触发器。

　　5．给出某个一般时序电路的图，有Tsetup，Tdelay，Tck&#61485;>q还有clock的del

ay，写出决定最大时钟的因素，同时给出表达式。

　　6．用C语言实现统计某个cell在某.v文件调用的次数。

　　7．Cache的主要部分。

　　2003 EE笔试题目

　　1．写出电流公式。

　　2．写出平板电容公式。

　　3．电阻R和电容C串联，输入电压为R和C之间的电压，输出电压分别为C上电压和R上电压，要求绘制这两种电路输入电压的频谱，判断这两种电路何为高通滤波器，何为低通滤波器。当RC<<T时，给出输入电压波形图，绘制两种电路的输出波形图。

　　4．给出时域信号，求其直流分量。

　　5．给出一时域信号，要求写出频率分量，并写出其傅立叶变换级数。当波形经过低通滤波器滤掉高次谐波而只保留一次谐波时，画出滤波后的输出波形。

　　6．有一时域信号S=V0sin(2pif0t)+V1cos(2pif1t)+V2sin(2pif3t+90)，写出当其通过低通、带通、高通滤波器后的信号表示方式。

　　7．给出一差分电路，告诉其输出电压Y+和Y&#61485;，求共模分量和差模分量。

　　8．一电源和一段传输线相连（长度为L，传输时间为T），画出终端处波形，考虑传输线无损耗。给出电源电压波形图，要求绘制终端波形图。

　　9．求锁相环的输出频率，给了一个锁相环的结构图。

　　10．给出一个堆栈的结构，求中断后显示结果，主要是考堆栈压入返回地址存放在低端地址还是高端。

　　2003 Graphic笔试题目

　　1．问答题

　　（1）texture mapping是什么？为什么要用filter？

　　（2）用float和int表示一个数，比如2，说明优点和缺点。

　　（3）在MPEG哪部分可以加速硬件？

　　（4）解释cubic和B-spline的差别，写出各自函数。

　　（5）写出几个Win API中的OpenGL函数。

　　（6）说出固定小数表示和浮点小数表示的优缺点。

　　（7）说出显卡可以优化哪些MPEG中的计算？

　　（8）说出Bezier和B-Spline曲线的区别。

　　2．用最简单的方法判断一个数是否是2的指数次幂。

　　3．S23E8和S10E5两种浮点数表示方法分析，表示0.25写一个类S10E5，实现从S23E8转 换。

　　4．用模版的方式实现三个量取最大值。

　　5．题目告诉你IEEE 16和32浮点数表示的规范，要求将&#61485;0.25分别用IEEE 16和

32表示并写一个C++函数将输入的IEEE 16表示转化为IEEE 32的表示。

　　6．用C语言写一个函数f（x）&#61501; x \* 0.5要求只能用整数操作。

　　2003 Software Engineer笔试题

　　1. Describe x86 PC’s architecture in a diagram cpu,core chipset, Cache,DRAM, IO-subsystem, IO-Bus

　　2. SWI instruction is often called a “supervisor call”, describe the actions in detail

　　\* Save the address of the instruction after the SWI in rl4\_svc.

　　\* Save the CPSR in SPSR\_svc.

　　\* Enter supervisor mode and disable IRQs.

　　\* Set the PC to 08 and begin executing the instruction there.

　　3.

　　\* What is PIO operation? advantage and disadvantage?

　　\* DMA operation? advantage and disadvantage?

　　\* Scatter/Gather DMA engine? how does it operate?

　　4. MP3 decoder related. (a flow chart of decoding is presented)

　　\* advantages of Huffman encoding?

　　\* why the aliasing reduction is necessary?

　　\* analytical expression in mathematics of the IMDCT?

　　\* which block in the flow chart is suitable for the software implementation and which for the hardware? why?

　　5. Assembly codes -> C language (about 15 lines).

　　6. Graduation thesis description.

汉王笔试题

　　高级研究人员（模式识别、图像处理类）招聘试题

　　说明：

　　可能您的专业并不完全符合本试题所涉及的领域。因此，并非所有的问题都需要回答，您可以只回答你所熟悉和能够回答的问题。允许参考任意的资料，但请独立完成此试题，我们更欣赏您独立的思考和创新的精神。本试题并非我们录用或者不录用您的惟一依据。应聘高级研究人员者请回答这部分问题。

　　1．人工智能与模式识别的研究已有多年，但似乎公认的观点认为它仍然非常困难。试对你所熟悉的任一方向（如指纹识别、人像识别、语音识别、字符识别、自然语言理解等）的发展状况进行描述。并设想如果你将从事该方向的研究，你打算如何着手，以建立有效的识别理论和方法；或者你认为现在的理论和方法有何缺陷，有什么办法来进行改进？（500字以内即可，不要太长）

　　2．简述下面任一主题的主要理论框架或主要观点（500字以内即可，不要太长）

　　（1）David Marr的视觉计算理论框架

　　（2）格式塔（Gestalt）心理学派的主要观点

　　（3）Bayes决策理论

　　（4）人工神经网络中的BP网络、自组织网络和联想记忆网络的主要内容

　　（5）基因算法

　　（6）小波分析

　　（7）目前流行的有损静态图像压缩方法

　　3．设想你要设计一个算法，检测给定的图像中是否有矩形结构。所要检测的矩形可能有多种形态，试提出你的算法框架。要求你的算法至少能检测出样本中的矩形，而拒绝其他的任意非矩形结构。矩形的大小、位置和方向未知，要求你的算法能确定这些参数。

　　如果你认为这个问题太难而不能解决，请说明理由。

　　高级软件开发人员招聘试题

　　说明：

　　可能您的专业并不完全符合本试题所涉及的领域。因此，并非所有的问题都需要回答，您可以只回答你所熟悉和能够回答的问题。允许参考任意的资料，但请独立完成此试题，我们更欣赏您独立的思考和创新的精神。本试题并非我们录用或者不录用您的惟一依据 。

　　应聘高级软件开发人员者请回答这部分问题。

　　1．数据的逻辑存储结构（如数组，队列，树等）对于软件开发具有十分重要的影响，试对你所了解的各种存储结构从运行速度、存储效率和适用场合等方面进行简要地分析。

　　2．数据库技术是计算机系统中一个非常重要的领域，几乎所有的计算机应用中都或多或少地用到了数据库。试简要地谈谈数据库设计中应当注意哪些问题，以及如何解决？给出两种你所熟悉的DBMS，要求一种适用于小型应用，另一种适用于大型应用，给出你做出选择的理由。

　　3．某公司的主要业务是提供WWW和E-mail服务，出于安全考虑，该公司要求我公司提供一套网络指纹登录系统，该系统要求能够利用指纹替代E-mail中常用的密码，并对所提供的部分网页通过指纹认证后才能访问，请利用你所学过的知识对该系统进行分析设计，你可以指定网络的配置（包括协议），但必须保证邮件用户既可通过网页（http方式）收取信件，也可通过Outlook收取信件。请分析该系统的可行性，可行时给出系统结构和主要的存储结构，指出系统中的难点和解决方法。（假设指纹识别的问题已经解决）

　　高级硬件开发人员招聘试题

　　说明：

　　可能您的专业并不完全符合本试题所涉及的领域。因此，并非所有的问题都需要回答，您可以只回答你所熟悉和能够回答的问题。允许参考任意的资料，但请独立完成此试题，我们更欣赏您独立的思考和创新的精神。本试题并非我们录用或者不录用您的惟一依据 。

　　应聘高级硬件开发人员者请回答这部分问题。

　　1．下面是一些基本的数字电路知识问题，请简要回答：

　　（1）什么是Setup和Holdup时间？

　　（2）什么是竞争与冒险现象？怎样判断？如何消除？

　　（3）请画出用D触发器实现2倍分频的逻辑电路。

　　（4）什么是“线与”逻辑？要实现它，在硬件特性上有什么具体要求？

　　（5）什么是同步逻辑和异步逻辑？

　　（6）请画出微机接口电路中，典型的输入设备与微机接口逻辑示意图（数据接口、控制接口、所存器／缓冲器）。

　　（7）你知道哪些常用的逻辑电平？TTL与COMS电平可以直接互连吗？

　　2．可编程逻辑器件在现代电子设计中越来越重要，请问：

　　（1）你所知道的可编程逻辑器件有哪些？

　　（2）试用VHDL或Verilog，ABLE描述8位D触发器逻辑

　　3．设想你将设计完成一个电子电路方案。请简述用EDA软件（如PROTEL）进行设计（包括原理图和PCB图）到调试出样机的整个过程。在各个环节应注意哪些问题？

北京信威通信技术股份有限公司面试题

　　1．DSP和通用处理器在结构上有什么不同？请简要画出你熟悉的一种DSP结构图。

　　2．说说定点DSP和浮点DSP的定义（或者说出他们的区别）。

　　3．说说你对循环寻址和位反序寻址的理解。

　　4．请写出【&#61485;8，7】的二进制补码和二进制偏置码。用Q15表示出0.5和&#614; 85;0.5。

中国国际金融有限公司CICC笔试题

　　1. Please tell us about an achievement that you are especially proud of because it was difficult or demanding.

　　（1）What the objective was?

　　（2）Why it is important to you?

　　（3）How you achieved it and the obstacles that you had to overcome in order to do so?

　　2. What is your career plan? Three years after graduation, and five years after graduation?

　　3. Why are you interested in investment bank? What other industries do you also have interests?

　　4. Why do you think you can be a qualified investment banker? How can you contribute in this industry?

国泰君安笔试题

　　一列火车上有三个工人，史密斯、琼斯和罗伯特，三人工作为消防员、司闸员和机械师，有三个乘客与这三人的名字相同。罗伯特住在底特律；司闸员住在芝加哥和底特律中间的地方；琼斯一年赚2万美金；有一个乘客和司闸员住在一个地方，每年的薪水是司闸员的3倍整；史密斯台球打得比消防员好；和司闸员同名的乘客住在芝加哥。

　　请问谁是机械师？

Briny笔试题

　　1．说出RC振荡器的构成和工作原理。

　　2．什么是SDH?

　　3．什么是共模、差模?画出差分电路的结构。

　　4．a=5; b=6; a+=b++; 执行结果是什么？

　　5．什么是TDM？什么是CDMA？

　　6．什么是采样定理？

　　7．什么是香农定理？

　　8．计算机的中断有哪几类？

广东北电面试题目

　　英文笔试题

　　1. Tranlation (Mandatory)

　　CDMA venders have worked hard to give CDMA roaming capabilities via the development of RUIM-essentially, a SIM card for CDMA handsets currently being deployed in China for new CDMA operator China Unicom. Korean cellco KTF

demonstr ated earlier this year the ability to roam between GSM and CDMA using such cards.However,only the card containing the user’s service data can roam-not the CDMA handset or the user’s number (except via call forwarding).

　　2. Programming (Mandatory)

　　Linked list

　　a. Implement a linked list for integers,which supports the insertafter (in sert a node after a specified node) and removeafter (remove the node after as pecified node) methods;

　　b. Implement a method to sort the linked list to descending order.

　　3. Debugging (Mandatory)

　　a. For each of the following recursive methods,enter Y in the answer box

i

f themethod terminaters (assume i=5), Otherwise enter N.

　　static int f(int i){

　　 return f(i-1)\*f(i-1);

　　 }

　　Ansewr:

　　static int f(int i){

　　 if(i==0){return 1;}

　　 else {return f(i-1)\*f(i-1);}

　　 }

　　Ansewr:

　　static int f(int i){

　　 if(i==0){return 1;}

　　 else {return f(i-1)\*f(i-2);}

　　 }

　　Ansewr:

　　b. There are two errors in the following JAVA program:

　　static void g(int i){

　　 if(i==1){return;}

　　 if(i%2==0){g(i/2);return;}

　　 else {g(3\*i);return;}

　　 }

　　please correct them to make sure we can get the printed-out result as below:

　　3 10 5 16 8 4 2 1

　　中文笔试题

　　1．汉译英

　　北电网络的开发者计划使来自于不同组织的开发者，能够在北电网络的平台上开发圆满的补充业务。北电网络符合工业标准的开放接口，为补充业务的开展引入了无数商机，开发者计划为不同层面的开发者提供不同等级的资格，资格的划分还考虑到以下因素：补充业务与北电网络平台的集合程度，开发者团体与北电网络的合作关系，等等。

　　2．编程

　　将整数转换成字符串：void itoa(int,char);

　　例如itoa(-123,s[])则s=“-123”;

U2合唱团在17分钟内得赶到演唱会场，途中必需跨过一座桥，四个人从桥的同一端出发，你得帮助他们到达另一端，天色很暗，而他们只有一只手电筒。一次同时最多可以有两人一起过桥，而过桥的时候必须持有手电筒，所以就得有人把手电筒带来带去，来回桥两端。手电筒是不能用丢的方式来传递的。四个人的步行速度各不同，若两人同行则以较慢者的速度为准。Bono需花1分钟过桥，Edge需花2分钟过桥，Adam需花5分钟过桥，Larry需花 10分钟过桥。他们要如何在17分钟内过桥呢？（有个同济的学生写文章说他当时在微软面 试时就是碰到了这道题，最短只能做出在19分钟内过桥，微软的人对他讲这样的结果已经是不错的了！）

　　A点到B点

　　1和2过去 2分钟 2

　　2过来 4分钟 2+2=4

　　10和5过去 14分钟 4+10=14

　　1过来 15分钟 14+1=15

　　1和2过去 17分钟 15+2=17

　　19分钟还很不错？？？？

这是广州本田的试题，大家看一下

广州本田笔试题

　　1．排序s-m-t-w-t-f-？

　　2．如果六千，六百，六表示成6606，那么十一千，十一百，十一表示成什么？

　　3．grass后面加一个词，agent前面加一个单词，组成两个新词，这个词是什么？

　　4．农场不知道有多少鸡，现有一批饲料，如果卖掉75只鸡饲料够20天用，买进100只 鸡饲料够用15天，问原来有多少只鸡？

　　5．6个桶，装着两种液体，一种液体的价格是另外一种的double，桶容量为8，13，15，17，19，31，有一个美国人，各用了14美元买两种液体，剩下一个桶。问剩下哪个？

　　6．篮球场，还剩6秒，差对手4分，没可能追得上，现在有一个暂停，你会怎么指导球 员去做？

明基面试问题

　　1．自我介绍（2分钟）。

　　2．你大学期间最辉煌的一件事是什么？

　　3．如果你明天去火星呆上300年，今天晚上你最想做的一件事是什么?

网易

　　1、10个人分成4组 有几种分法？

　　2、如图：

　　　　 7 8 9 10

　　　　 6 1 2 11

　　　　 5 4 3 12

　　　　 16 15 14 13

　　　　设“1”的坐标为（0，0） “7”的坐标为（－1，－1） 编写一个小程序，使程序做到输入坐标（X,Y）之后显示出相应的数字。

　　3、#include<stdio.h>

　　　　//example input and output

　　　　//in 1 2 3 out 1 3 1

　　　　//in 123456789 2 100 out 123456789 100 21

　　　　long mex(long a,long b,long c)

　　　　{ long d;

　　　　 if(b==0) return 0;

　　　　 if(b==1) return a%c;

　　　　 d=mex(a,b/2,c)； d\*=d;这里忘了;d\*=mex(a,b%2,c);d%=c;

　　　　 return d;

　　　　}

　　　　int main(void)

　　　　{ long x,y,z;

　　　　 while(1)

　　　　 { if(scanf(%d %d %d,&x,&y,&z)>3) return 0;

　　　　 if(x<0) { printf("too small\n");continue;}

　　　　 if(y<0) { printf("too small\n");continue;}

　　　　 if(z<1) { printf("too small\n");continue;}

　　　　 if(y>z) { printf("too big\n");continue;}

　　　　 if(z>1000000010) {printf("too big\n");continue}

　　　　 printf(%d %d %d,x,z,mex(x,y,z);

　　　　}}

　　　　根据这个程序，当已知一个输入，算出输出，如：输入 1 3 1 则输出 1 2 3

　　　　输 入 123456789 100 21 输出 123456789 2 100

问答题：

1．实现双向链表删除一个节点P，在节点P后插入一个节点，写出这两个函数。

2．写一个函数，将其中的\t都转换成4个空格。

3．Windows程序的入口是哪里？写出Windows消息机制的流程。

4．如何定义和实现一个类的成员函数为回调函数？

5．C++里面是不是所有的动作都是main()引起的？如果不是，请举例。

6．C++里面如何声明const void f(void)函数为C程序中的库函数？

7．下列哪两个是等同的

int b;

A const int\* a = &b;

B const\* int a = &b;

C const int\* const a = &b;

D int const\* const a = &b;

8．内联函数在编译时是否做参数类型检查？

void g(base & b){

b.play;

}

void main(){

son s;

g(s);

return;

}

程序设计

1.（可以用自然语言来描述，不编程）：C/C++源代码中，检查花括弧（是"（"与

"）"，"{"与"}"）是否匹配，若不匹配，则输出不匹配花括弧所在的行与列。

2，巧排数字，将1,2,...,19,20这20个数字排成一排，使得相邻的两个数字之和为一个素数，且

首尾两数字之和也为一个素数。编程打印出所有的排法。

3，打印一个N\*N的方阵，N为每边字符的个数（ 3〈N〈20 ），要求最外层为"X"，第二层为"Y"，从第三层起每层依次打印数字0，1，2，3，...

例子：当N =5，打印出下面的图形：

X X X X X

X Y Y Y X

X Y 0 Y X

X Y Y Y X

X X X X X

1、局部变量能否和全局变量重名？

答：局部变量可以与全局变量同名，在函数内引用这个变量时，会用到同名的局部变量，而不会用到全局变量。对于有些编译器而言，在同一个函数内可以定义多个同名的局部变量，比如在两个循环体内都定义一个同名的局部变量，而那个局部变量的作用域就在那个循环体内



2、如何引用一个已经定义过的全局变量？

答：可以用引用头文件的方式，也可以用extern关键字，如果用引用头文件方式来引用某个在头文件中声明的全局变理，假定你将那个变写错了，那么在编译期间会报错，如果你用extern方式引用时，假定你犯了同样的错误，那么在编译期间不会报错，而在连接期间报错。（不要问我什么是编译期间，什么是连接期间，请自己查资料）





3、全局变量定义在可被多个.C文件包含的头文件中可不可以？为什么？

答：可以，但要加static



4、语句for( ；1 ；)有什么问题？它是什么意思？

答：死循环，同while(1)，这种写法并不规范，可读性差，一般不用



5、do……while和while……do有什么区别？

答：前一个循环一遍再判断，后一个判断以后再循环



6、请写出下列代码的输出内容

#include

main()

{

int a,b,c,d;

a=10;

b=a++;

c=++a;

d=10\*a++;

printf("b，c，d：%d，%d，%d"，b，c，d）;

return 0;

}

答：10，12，120



7.static函数与普通函数有什么区别？

只在当前源文件中使用的函数应该说明为内部函数(static)，内部函数应该在当前源文件中说明和定义。对于可在当前源文件以外使用的函数，应该在一个头文件中说明，要使用这些函数的源文件要包含这个头文件。



8.static全局变量与普通的全局变量有什么区别？

全局变量(外部变量)的说明之前再冠以static 就构成了静态的全局变量。全局变量本身就是静态存储方式， 静态全局变量当然也是静态存储方式。

这两者在存储方式上并无不同。这两者的区别虽在于非静态全局变量的作用域是整个源程序， 当一个源程序由多个源文件组成时，非静态的全局变量在各个源文件中都是有效的。 而静态全局变量则限制了其作用域， 即只在定义该变量的源文件内有效， 在同一源程序的其它源文件中不能使用它。由于静态全局变量的作用域局限于一个源文件内，只能为该源文件内的函数公用， 因此可以避免在其它源文件中引起错误。从以上分析可以看出， 把局部变量改变为静态变量后是改变了它的存储方式即改变了它的生存期。把全局变量改变为静态变量后是改变了它的作用域， 限制了它的使用范围。



9.程序的局部变量存在于（栈）中，全局变量存在于（静态区 ）中，动态申请数据存在于（堆 ）中。



10.设有以下说明和定义：

typedef union {long i; int k[5]; char c;} DATE;

struct data { int cat; DATE cow; double dog;} too;

DATE max;

则语句 printf("%d",sizeof(struct date)+sizeof(max));的执行结果是：\_\_\_52\_\_\_\_



11.队列和栈有什么区别？

队列先进先出，栈后进先出



12.写出下列代码的输出内容

#include

int inc(int a)

{

return(++a);

}

int multi(int\*a,int\*b,int\*c)

{

return(\*c=\*a\*\*b);

}

typedef int(FUNC1)(int in);

typedef int(FUNC2) (int\*,int\*,int\*);



void show(FUNC2 fun,int arg1, int\*arg2)

{

INCp=&inc;

int temp =p(arg1);

fun(&temp,&arg1, arg2);

printf("%d\n",\*arg2);

}



main()

{

int a;

show(multi,10,&a);

return 0;

}



翻译一下

==================================

#include

using namespace std;



// 返回a + 1

int inc(int a){

return a+1;

}

// out = a\*b

int multi(int& a, int& b, int& out){

return out = a\*b;

}



// 定义两种函数,FUNC1对应inc, FUNC2对应 multi

typedef int(FUNC1) (int in);

typedef int(FUNC2) (int&,int&,int&);



// 诡异的函数,这是模糊C大赛么...

void show( FUNC2\* func, int num, int& out ){

FUNC1\* pInc = inc;//原文这句写错了...orz... INC根本没定义

int numAdd1 = pInc( num );

// 等价于 numAdd1 = inc( num );

// 结果 num仍然是10, numAdd1 = 11



// 调用func函数,可以看到, main中传给func的是 multi.

// 所以调用的是 multi( 11, 10, out ), out = 110

func( numAdd1, num, out );

cout<}



int main(){

int a;

show(multi,10, a);

return 0;

}

==================================



13.请找出下面代码中的所以错误

说明：以下代码是把一个字符串倒序，如“abcd”倒序后变为“dcba”

==================================

main(){

char\*src="hello,world";

char\* dest=NULL;



int len=strlen(src);

dest=(char\*)malloc(len);//!1 少分配了一字节. strlen计数不包括\0



char\* d = dest;

char\* s = src[len];//!2 两个错误. 应该为 &src[ len-1 ]

while( len-- != 0 )

d++=s--;//!3 \*d++ = \*s--;

//!4 尾部要\0,应加上 \*d = 0;



printf("%s",dest);

return 0;

}

==================================



修正后的答案:

==================================

int main(){

char\* src = "hello,world";



int len = strlen(src);

char\* dest = (char\*)malloc(len+1);.



char\* d = dest;

char\* s = &src[len-1];

while( len-- != 0 )

\*d++=\*s--;

\*d = 0;



printf("%s",dest);

return 0;

}

==================================

1. 写一个“标准”宏MIN，这个宏输入两个参数并返回较小的一个。

#define MIN(A,B) ((A) <= (B) (A) : (B))

这个测试是为下面的目的而设的：

1). 标识#define在宏中应用的基本知识。这是很重要的，因为直到嵌入(inline)操作符变为标准C的一部分，宏是方便产生嵌入代码的唯一方法，对于嵌入式系统来说，为了能达到要求的性能，嵌入代码经常是必须的方法。

2). 三重条件操作符的知识。这个操作符存在C语言中的原因是它使得编译器能产生比if-then-else更优化的代码，了解这个用法是很重要的。

3). 懂得在宏中小心地把参数用括号括起来

4). 我也用这个问题开始讨论宏的副作用，例如：当你写下面的代码时会发生什么事？

least = MIN(\*p++, b);

3. 预处理器标识#error的目的是什么？

如果你不知道答案，请看参考文献1。这问题对区分一个正常的伙计和一个书呆子是很有用的。只有书呆子才会读C语言课本的附录去找出象这种

问题的答案。当然如果你不是在找一个书呆子，那么应试者最好希望自己不要知道答案。

死循环（Infinite loops）

4. 嵌入式系统中经常要用到无限循环，你怎么样用C编写死循环呢？

这个问题用几个解决方案。我首选的方案是：

while(1) { }

一些程序员更喜欢如下方案：

for(;;) { }

这个实现方式让我为难，因为这个语法没有确切表达到底怎么回事。如果一个应试者给出这个作为方案，我将用这个作为一个机会去探究他们这样做的

基本原理。如果他们的基本答案是：“我被教着这样做，但从没有想到过为什么。”这会给我留下一个坏印象。

第三个方案是用 goto

Loop:

...

goto Loop;

应试者如给出上面的方案，这说明或者他是一个汇编语言程序员（这也许是好事）或者他是一个想进入新领域的BASIC/FORTRAN程序员。

数据声明（Data declarations）

5. 用变量a给出下面的定义

a) 一个整型数（An integer）

b) 一个指向整型数的指针（A pointer to an integer）

c) 一个指向指针的的指针，它指向的指针是指向一个整型数（A pointer to a pointer to an integer）

d) 一个有10个整型数的数组（An array of 10 integers）

e) 一个有10个指针的数组，该指针是指向一个整型数的（An array of 10 pointers to integers）

f) 一个指向有10个整型数数组的指针（A pointer to an array of 10 integers）

g) 一个指向函数的指针，该函数有一个整型参数并返回一个整型数（A pointer to a function that takes an integer as an argument and returns an integer）

h) 一个有10个指针的数组，该指针指向一个函数，该函数有一个整型参数并返回一个整型数（ An array of ten pointers to functions that take an integer argument and return an integer ）

答案是：

a) int a; // An integer

b) int \*a; // A pointer to an integer

c) int \*\*a; // A pointer to a pointer to an integer

d) int a[10]; // An array of 10 integers

e) int \*a[10]; // An array of 10 pointers to integers

f) int (\*a)[10]; // A pointer to an array of 10 integers

g) int (\*a)(int); // A pointer to a function a that takes an integer argument and returns an integer

h) int (\*a[10])(int); // An array of 10 pointers to functions that take an integer argument and return an integer

人们经常声称这里有几个问题是那种要翻一下书才能回答的问题，我同意这种说法。当我写这篇文章时，为了确定语法的正确性，我的确查了一下书。

但是当我被面试的时候，我期望被问到这个问题（或者相近的问题）。因为在被面试的这段时间里，我确定我知道这个问题的答案。应试者如果不知道

所有的答案（或至少大部分答案），那么也就没有为这次面试做准备，如果该面试者没有为这次面试做准备，那么他又能为什么出准备呢？

Static

6. 关键字static的作用是什么？

这个简单的问题很少有人能回答完全。在C语言中，关键字static有三个明显的作用：

1). 在函数体，一个被声明为静态的变量在这一函数被调用过程中维持其值不变。

2). 在模块内（但在函数体外），一个被声明为静态的变量可以被模块内所用函数访问，但不能被模块外其它函数访问。它是一个本地的全局变量。

3). 在模块内，一个被声明为静态的函数只可被这一模块内的其它函数调用。那就是，这个函数被限制在声明它的模块的本地范围内使用。

大多数应试者能正确回答第一部分，一部分能正确回答第二部分，同是很少的人能懂得第三部分。这是一个应试者的严重的缺点，因为他显然不懂得本地化数据和代码范围的好处和重要性。

Const

7．关键字const是什么含意？

我只要一听到被面试者说：“const意味着常数”，我就知道我正在和一个业余者打交道。去年Dan Saks已经在他的文章里完全概括了const的所有用法，因此ESP(译者：Embedded Systems Programming)的每一位读者应该非常熟悉const能做什么和不能做什么.如果你从没有读到那篇文章，只要能说出const意味着“只读”就可以了。尽管这个答案不是完全的答案，但我接受它作为一个正确的答案。（如果你想知道更详细的答案，仔细读一下Saks的文章吧。）如果应试者能正确回答这个问题，我将问他一个附加的问题：下面的声明都是什么意思？

const int a;

int const a;

const int \*a;

int \* const a;

int const \* a const;

前两个的作用是一样，a是一个常整型数。第三个意味着a是一个指向常整型数的指针（也就是，整型数是不可修改的，但指针可以）。第四个意思a是一个指向整型数的常指针（也就是说，指针指向的整型数是可以修改的，但指针是不可修改的）。最后一个意味着a是一个指向常整型数的常指针（也就是说，指针指向的整型数是不可修改的，同时指针也是不可修改的）。如果应试者能正确回答这些问题，那么他就给我留下了一个好印象。顺带提一句，也许你可能会问，即使不用关键字const，也还是能很容易写出功能正确的程序，那么我为什么还要如此看重关键字const呢？我也如下的几下理由：

1). 关键字const的作用是为给读你代码的人传达非常有用的信息，实际上，声明一个参数为常量是为了告诉了用户这个参数的应用目的。如果你曾花很多时间清理其它人留下的垃圾，你就会很快学会感谢这点多余的信息。（当然，懂得用const的程序员很少会留下的垃圾让别人来清理的。）

2). 通过给优化器一些附加的信息，使用关键字const也许能产生更紧凑的代码。

3). 合理地使用关键字const可以使编译器很自然地保护那些不希望被改变的参数，防止其被无意的代码修改。简而言之，这样可以减少bug的出现。

Volatile

8. 关键字volatile有什么含意 并给出三个不同的例子。

一个定义为volatile的变量是说这变量可能会被意想不到地改变，这样，编译器就不会去假设这个变量的值了。精确地说就是，优化器在用到这个变量时必须每次都小心地重新读取这个变量的值，而不是使用保存在寄存器里的备份。下面是volatile变量的几个例子：

1). 并行设备的硬件寄存器（如：状态寄存器）

2). 一个中断服务子程序中会访问到的非自动变量(Non-automatic variables)

3). 多线程应用中被几个任务共享的变量

回答不出这个问题的人是不会被雇佣的。我认为这是区分C程序员和嵌入式系统程序员的最基本的问题。嵌入式系统程序员经常同硬件、中断、RTOS等等打交道，所用这些都要求volatile变量。不懂得volatile内容将会带来灾难。

假设被面试者正确地回答了这是问题（嗯，怀疑这否会是这样），我将稍微深究一下，看一下这家伙是不是直正懂得volatile完全的重要性。

1). 一个参数既可以是const还可以是volatile吗？解释为什么。

2). 一个指针可以是volatile 吗？解释为什么。

3). 下面的函数有什么错误：

int square(volatile int \*ptr)

{

return \*ptr \* \*ptr;

}

下面是答案：

1). 是的。一个例子是只读的状态寄存器。它是volatile因为它可能被意想不到地改变。它是const因为程序不应该试图去修改它。

2). 是的。尽管这并不很常见。一个例子是当一个中服务子程序修该一个指向一个buffer的指针时。

3). 这段代码的有个恶作剧。这段代码的目的是用来返指针\*ptr指向值的平方，但是，由于\*ptr指向一个volatile型参数，编译器将产生类似下面的代码：

int square(volatile int \*ptr)

{

int a,b;

a = \*ptr;

b = \*ptr;

return a \* b;

}

由于\*ptr的值可能被意想不到地该变，因此a和b可能是不同的。结果，这段代码可能返不是你所期望的平方值！正确的代码如下：

long square(volatile int \*ptr)

{

int a;

a = \*ptr;

return a \* a;

}

位操作（Bit manipulation）

9. 嵌入式系统总是要用户对变量或寄存器进行位操作。给定一个整型变量a，写两段代码，第一个设置a的bit 3，第二个清除a 的bit 3。在以上两个操作中，要保持其它位不变。

对这个问题有三种基本的反应

1). 不知道如何下手。该被面者从没做过任何嵌入式系统的工作。

2). 用bit fields。Bit fields是被扔到C语言死角的东西，它保证你的代码在不同编译器之间是不可移植的，同时也保证了的你的代码是不可重用的。我最近不幸看到Infineon为其较复杂的通信芯片写的驱动程序，它用到了bit fields因此完全对我无用，因为我的编译器用其它的方式来实现bit fields的。从道德讲：永远不要让一个非嵌入式的家伙粘实际硬件的边。

3). 用 #defines 和 bit masks 操作。这是一个有极高可移植性的方法，是应该被用到的方法。最佳的解决方案如下：

#define BIT3 (0x1<<3)

static int a;

void set\_bit3(void)

{

a |= BIT3;

}

void clear\_bit3(void)

{

a &= ~BIT3;

}

一些人喜欢为设置和清除值而定义一个掩码同时定义一些说明常数，这也是可以接受的。我希望看到几个要点：说明常数、|=和&=~操作。

访问固定的内存位置（Accessing fixed memory locations）

10. 嵌入式系统经常具有要求程序员去访问某特定的内存位置的特点。在某工程中，要求设置一绝对地址为0x67a9的整型变量的值为0xaa66。编译器是一个纯粹的ANSI编译器。写代码去完成这一任务。

这一问题测试你是否知道为了访问一绝对地址把一个整型数强制转换（typecast）为一指针是合法的。这一问题的实现方式随着个人风格不同而不同。典型的类似代码如下：

int \*ptr;

ptr = (int \*)0x67a9;

\*ptr = 0xaa55;

一个较晦涩的方法是：

\*(int \* const)(0x67a9) = 0xaa55;

即使你的品味更接近第二种方案，但我建议你在面试时使用第一种方案。

中断（Interrupts）

11. 中断是嵌入式系统中重要的组成部分，这导致了很多编译开发商提供一种扩展—让标准C支持中断。具代表事实是，产生了一个新的关键字\_\_interrupt。下面的代码就使用了\_\_interrupt关键字去定义了一个中断服务子程序(ISR)，请评论一下这段代码的。

\_\_interrupt double compute\_area (double radius)

{

double area = PI \* radius \* radius;

printf(" Area = %f", area);

return area;

}

这个函数有太多的错误了，以至让人不知从何说起了：

1). ISR 不能返回一个值。如果你不懂这个，那么你不会被雇用的。

2). ISR 不能传递参数。如果你没有看到这一点，你被雇用的机会等同第一项。

3). 在许多的处理器/编译器中，浮点一般都是不可重入的。有些处理器/编译器需要让额处的寄存器入栈，有些处理器/编译器就是不允许在ISR中做浮点运算。此外，ISR应该是短而有效率的，在ISR中做浮点运算是不明智的。

4). 与第三点一脉相承，printf()经常有重入和性能上的问题。如果你丢掉了第三和第四点，我不会太为难你的。不用说，如果你能得到后两点，那么你的被雇用前景越来越光明了。

代码例子（Code examples）

12 . 下面的代码输出是什么，为什么？

void foo(void)

{

unsigned int a = 6;

int b = -20;

(a+b > 6) puts("> 6") : puts("<= 6");

}

这个问题测试你是否懂得C语言中的整数自动转换原则，我发现有些开发者懂得极少这些东西。不管如何，这无符号整型问题的答案是输出是“>6”。原因是当表达式中存在有符号类型和无符号类型时所有的操作数都自动转换为无符号类型。 因此-20变成了一个非常大的正整数，所以该表达式计算出的结果大于6。这一点对于应当频繁用到无符号数据类型的嵌入式系统来说是丰常重要的。如果你答错了这个问题，你也就到了得不到这份工作的边缘。

13. 评价下面的代码片断：

unsigned int zero = 0;

unsigned int compzero = 0xFFFF;

/\*1's complement of zero \*/

对于一个int型不是16位的处理器为说，上面的代码是不正确的。应编写如下：

unsigned int compzero = ~0;

这一问题真正能揭露出应试者是否懂得处理器字长的重要性。在我的经验里，好的嵌入式程序员非常准确地明白硬件的细节和它的局限，然而PC机程序往往把硬件作为一个无法避免的烦恼。

到了这个阶段，应试者或者完全垂头丧气了或者信心满满志在必得。如果显然应试者不是很好，那么这个测试就在这里结束了。但如果显然应试者做得不错，那么我就扔出下面的追加问题，这些问题是比较难的，我想仅仅非常优秀的应试者能做得不错。提出这些问题，我希望更多看到应试者应付问题的方法，而不是答案。不管如何，你就当是这个娱乐吧…

动态内存分配（Dynamic memory allocation）

14. 尽管不像非嵌入式计算机那么常见，嵌入式系统还是有从堆（heap）中动态分配内存的过程的。那么嵌入式系统中，动态分配内存可能发生的问题是什么？

这里，我期望应试者能提到内存碎片，碎片收集的问题，变量的持行时间等等。这个主题已经在ESP杂志中被广泛地讨论过了（主要是 P.J. Plauger, 他的解释远远超过我这里能提到的任何解释），所有回过头看一下这些杂志吧！让应试者进入一种虚假的安全感觉后，我拿出这么一个小节目：下面的代码片段的输出是什么，为什么？

char \*ptr;

if ((ptr = (char \*)malloc(0)) == NULL)

puts("Got a null pointer");

else

puts("Got a valid pointer");

这是一个有趣的问题。最近在我的一个同事不经意把0值传给了函数malloc，得到了一个合法的指针之后，我才想到这个问题。这就是上面的代码，该代码的输出是“Got a valid pointer”。我用这个来开始讨论这样的一问题，看看被面试者是否想到库例程这样做是正确。得到正确的答案固然重要，但解决问题的方法和你做决定的基本原理更重要些。

Typedef

15. Typedef 在C语言中频繁用以声明一个已经存在的数据类型的同义字。也可以用预处理器做类似的事。例如，思考一下下面的例子：

#define dPS struct s \*

typedef struct s \* tPS;

以上两种情况的意图都是要定义dPS 和 tPS 作为一个指向结构s指针。哪种方法更好呢？（如果有的话）为什么？

这是一个非常微妙的问题，任何人答对这个问题（正当的原因）是应当被恭喜的。答案是：typedef更好。思考下面的例子：

dPS p1,p2;

tPS p3,p4;

第一个扩展为

struct s \* p1, p2;

上面的代码定义p1为一个指向结构的指，p2为一个实际的结构，这也许不是你想要的。第二个例子正确地定义了p3 和p4 两个指针。

晦涩的语法

16. C语言同意一些令人震惊的结构,下面的结构是合法的吗，如果是它做些什么？

int a = 5, b = 7, c;

c = a+++b;

这个问题将做为这个测验的一个愉快的结尾。不管你相不相信，上面的例子是完全合乎语法的。问题是编译器如何处理它？水平不高的编译作者实际上会争论这个问题，根据最处理原则，编译器应当能处理尽可能所有合法的用法。因此，上面的代码被处理成：

c = a++ + b;

因此, 这段代码持行后a = 6, b = 7, c = 12。

如果你知道答案，或猜出正确答案，做得好。如果你不知道答案，我也不把这个当作问题。我发现这个问题的最大好处是:这是一个关于代码编写风格，代码的可读性，代码的可修改性的好的话题



# posted by 孜孜不倦心 @ 2006-02-28 22:24 评论(1)

aaa

1. Web服务器控件和Windows控件的执行有何不同？

【解答】一般来说，大部分Web服务器控件和Windows控件的功能和使用都很相似，但是二者在内部实现上却有着本质的区别。Windows控件的属性、方法、事件等都是在本机上执行的，而Web服务器控件则全部是在服务器端执行的。

2. 利用HiddenField控件是否可用来存储保密数据？为什么？

【解答】HiddenField控件，即隐藏输入框的服务器控件，可以用于存储非显示的值。在Web应用程序中，可以使用HiddenField控件来存储Web窗体页的状态值。但是，在Web应用程序运行期间，查看源代码会发现该控件是以<input type="hidden" name="HiddenField1" id="HiddenField1" value="3" />的形式呈现给客户端浏览器的，其隐藏值value是用户可见的，所以它不适用于存储需要保密数据。

3. SqlDataSource组件的作用是什么？

【解答】SqlDataSource组件是一种数据源控件，它可以连接到任何ADO.NET支持的数据源（如SQL Server、Oracle等），从中检索数据，并使得其他控件（如GridView等控件）可以绑定到数据源。使用SqlDataSource组件访问数据，只需要提供用于连接到数据库的连接字符串，并定义访问数据的SQL语句或存储过程。在运行时，SqlDataSource组件会自动打开与数据库的连接，执行SQL语句或存储过程，完成数据访问后自动关闭连接。

4. 采用GridView控件、DataList控件、DetailsView控件来显示数据各有何特点？

【解答】GridView控件、DataList控件、DetailsView控件都可以利用SqlDataSource控件绑定SQL数据源自动显示数据。就显示数据的能力来说，GridView控件、DataList控件一次可以显示数据表中的多条记录，而DetailsView控件每次只能显示数据表中的一条记录；就操作数据的能力来说，GridView控件、DetailsView控件都可以自动利用SqlDataSource控件的数据操作命令来编辑所显示的数据，而DataList控件则需要编写自身的相应事件才可以完成编辑数据的操作。

5. 使用站点地图进行导航时，SiteMapPath控件与Menu控件有何不同？

【解答】利用站点地图Web.sitemap进行网站导航时，SiteMapPath控件仅能显示用于在站点地图中列出的页面中显示导航路径，而Menu控件即使使用在站点地图中没有列出的页面中，也可以生成菜单，显示所有的导航数据。

**ASP笔试题**

1、请写出下面的ASP中常用函数或对象的作用。（20分）

ASC() 作用：

CInt() 作用：

Fix() 作用：

LTrim() 作用：

UCase() 作用：

Response.Write 作用：

Response.Redirect 作用：

Request.QueryString 作用：

Server.CreateObject 作用：

Request.ServerVariables("REMOTE\_ADDR") 作用：

2、请指出以下三个对象在ASP页面间传递参数时的区别。（10分）

Request、Request.Form、Request.QueryString

3、请写出application与session的区别。（5分）

4、请用ASP实现冒泡排序。（10分）

5. 有两个表Table\_A和Table\_B，字段和内容分别是

表 Table\_A 表 Table\_B

userid username Address userid username Address

---------------------------------------- ----------------------------------------

1 A 1 D

2 B 2 E

3 C 3 F

问题一：请写出一条SQL语句，目的是将Table\_B和Table\_A比较，如果userid相同，就把Table\_B中Address的值写到Table\_A的字段Address里。（10分）

问题二：请写出一条SQL语句，目的是将Table\_B和Table\_A比较，如果userid相同，就把Table\_A中的username的值、Table\_B中的Address的值读出来并写入一个新表。（10分）

6、什么叫做SQL注入，如何防止？请举例说明。（10分）

7、如何实现ASP页面在网页上显示？请详细说明。（5分）

8、表table\_list里有100条订单数据，该表有：ID，LIST＿NO，ORDER＿ID，USER＿NAME，EDIT＿DATE等字段，其中字段ID为自增ID，请写一段程序，把该表ID为奇数的记录显示在一个页面上，用表格格式化。（20）

附加题：有分别显示数字0-9的10张图片0.gif、1.gif、……9.gif，请用这些图片

简单的实现常见的验证码效果。要求写成两个ASP文件a.asp和b.asp，其中a.

asp是表单页面； b.asp是表单处理页面、包含对验证码的判断。

1、估计一下杭州有多少软件工程师，如果允许，你还需要那些调研工作？并给出你的推导过程。你认为你的 估算结果可信吗？

2、看过那些软件方面的书籍

3、什么是软件，软件开发包括那些步骤，并说出你对这些步骤的理解

4、OSI网络结构的七层模型分别是什么，并说出你认为划分为7层的理由。

5、软件过程中了解、使用过版本控制或建模工具吗？对你工作有什么帮助？

6、软件过程中你都写过什么文档？分别是什么目的及其对你的帮助？

7、我们知道，C++将内存划分为三个逻辑区域：堆、栈和静态存储，请说出它们的区别及你的理解。

8、字符串A是由n个小写英文字母（a ~ z）构成的，定义为char A[n]。你能用更少的空间表示这个字符串吗？请写出从char A[n]到你的新的储存格式的转换函数。（请用C/C++编程，不允许上机操作）

12、对现在的Stack（栈）数据结构进行改进，加一个min()功能，使之能在常数，即O(1)，时间内给出栈中的最小值。可对push()和pop()函数进行修改，但要求其时间复杂度都只能是O(1)。

9、C++构造函数为什么不能是虚函数？

10、C++中virtual与inline的含义分别是什么？虚函数的特点；内联函数的特点；一个函数能否即是虚函数又是内联函数？

12、请列出实现C++代码你认为需要注意的一些问题。

11、你在原来的项目中有没有遇到什么困难，又是如何解决的？

1.new有几种用法

第一种:new Class();

第二种:覆盖方法

public new XXXX(){}

第三种:new 约束指定泛型类声明中的任何类型参数都必须有公共的无参数构造函数。

2.如何把一个array复制到arrayList里

foreach( object o in array )arrayList.Add(o);

3.datagrid.datasouse可以连接什么数据源 [dataset,datatable,dataview]

dataset,datatable,dataview , IList

4.概述反射和序列化(框架设计)

反射:程序集包含模块，而模块包含类型，类型又包含成员。反射则提供了封装程序集、模块和类型的对象。您可以使用反射动态地创建类型的实例，将类型绑定到现有对象，或从现有对象中获取类型。然后，可以调用类型的方法或访问其字段和属性

序列化:序列化是将对象转换为容易传输的格式的过程。例如，可以序列化一个对象，然后使用 HTTP 通过 Internet 在客户端和服务器之间传输该对象。在另一端，反序列化将从该流重新构造对象。

5.概述o/r mapping 的原理

利用反射，配置 将类于数据库表映射

6.类成员有( )种可访问形式

可访问形式？不懂。

可访问性：public ,protected ,private,internal

7.用sealed修饰的类有什么特点

sealed 修饰符用于防止从所修饰的类派生出其它类。如果一个密封类被指定为其他类的基类，则会发生编译时错误。

密封类不能同时为抽象类。

sealed 修饰符主要用于防止非有意的派生，但是它还能促使某些运行时优化。具体说来，由于密封类永远不会有任何派生类，所以对密封类的实例的虚拟函数成员的调用可以转换为非虚拟调用来处理。

8.列举ADO.NET中的五个主要对象，并简单描述

connection,command,dataReader,trans,dataset ...

9.执行下面代码后：

String strTemp ="abcdefg 某某某"；

Int i System.Text.Encoding.Default.GetBytes(strTemp).Length;

Int j = strTemp.Length;

问：i=(14 ) ；j=(11 )

i=(14 ) ；j=(11 ) 中文两个字节

10.C#中，string str = null 与 string str =""，请尽量用文字说明区别。(要点：说明详细的内存空间分配)

string str ="" 分配空间 (在堆栈中).

11.详述.NET里class和struct的异同！

class:放在 ? struct放在？

struct值传递

类与结构有很多相似之处：结构可以实现接口，并且可以具有与类相同的成员类型。然而，结构在几个重要方面不同于类：结构为值类型而不是引用类型，并且结构不支持继承。结构的值存储在“在堆栈上”或“内联”。细心的程序员有时可以通过聪明地使用结构来增强性能。

12.概述.NET里对 remoting 和 webservice 两项技术的理解和实际中的应用。

远程逻辑调用，remoing接口只能用在.net中

13.什么是code-behind技术

aspx and cs

14.概述三层结构体系

web/business/dataaccess

15.asp.net如何实现MVC模式，举例说明！

web/business/dataaccess (m:模板，v：视图，c控制器)

.NET面试题

1、当类T只声明了私有实例构造函数时，则在T的程序文本外部，\_\_\_\_\_（可以/不可以）从T派生出新的类，\_\_\_\_\_（可以/不可以）直接创建T的任何实例。

2、重载的关键字\_\_\_\_\_,重写的关键字\_\_\_\_\_,请描述一下他们的区别

3、请描述一下.Net架构

4、请描述DataReader与Dataset有什么区别

5、请描述ViewState 与Session有什么区别

6、请编程遍历窗体上所有TextBox控件并给它赋值为string.Empty

7、请编程实现单向链表的倒置，节点的结构自己定义

8、请找出下面这段程序的错误并修正

Public Function getDataFromDB(ByVal strID As String, ByRef tblDT As DataTable) As Integer

Dim connStr As String = \_

System.Configuration.ConfigurationSettings.AppSettings("DBConnectionString")

Dim conn As SqlConnection

Dim ds As New DataSet

Dim iRet As Integer = 0

Dim sqlDA As SqlDataAdapter

Dim sqlComm As SqlCommand

Dim sql As String

If IsNothing(strID) Then

strID = String.Empty

End If

strID = strID.Trim

conn = New SqlConnection(connStr)

Try

conn.Open()

sql = "SELECT \* FROM WWJB01 WHERE ID LIKE '%" & strID & "%'"

sqlComm = New SqlCommand(sql, conn)

sqlDA = New SqlDataAdapter

sqlDA.SelectCommand = sqlComm

sqlDA.Fill(ds)

If Not IsNothing(ds) Then

If ds.Tables.Count > 1 Then tblDT = ds.Tables(0).Copy

iRet = ds.Tables(0).Rows.Count

End If

End If

Catch ex As Exception

Throw ex

Finally

If Not conn.State = ConnectionState.Closed Then

conn.Close()

conn.Dispose()

End If

End Try

Return iRet

End Function

1.写一个函数 int strcmp(const str \*str1,const str \*str2)

要求：如果str1=str2 return 0;如果str1>str2 返回一个正数;如果str1<str2 返回一个负数。

不能调用其他函数。。。。。

2.class A{

void P1(){cout<<"A111"};

void P2(){cout<<"A222"};

};

class B:public A{

void P1(){cout<<"B111"};

virtual void P2(){cout<<"B222"};

}

..........

A\*cl=NULL;

cl=new A;

A->P1();

A->P2();

delete cl;

cl=NULL;

cl=new B;

A->P1();

A->P2();

delete cl;

cl=NULL;

..........

写出运行结果;

3.说一说const的用处以及好处?

4.编写一个函数；并封装成一个dll

int ULCompare(lPSTR str1,LPSTR str2,BOOL m)；URL地址字符串比较 .

5.就是问问你对ASCII,UNICOD,还有一个种什么编码哟，的特点以及优点；

6.正则表达式做全文搜索的好处。

7.说说B树的优点以及用法。

1、How many bytes does an IP adress consist of?

2、用SQL写语句：在下列三张表中选出CLASSID 为“1”的人的NAME和AGE。

表1 表2 表3

No Name Age No ID No

102 Ja 23 102 1 102

103 Ba 23 104 1 103

104 Na 24 103 2 104

3、用你熟悉的语言写出几行用于访问数据库的程序代码。

4、写出下列标记的作用

<META>

<LI>

<BR>

<TD>

<FRAME>

5、在网页中如何表示下列符号：

<

>

&

》

《

ASP测试题：

1. 根据时间，转化提示语，例如：23点就提示：现在时间是十一点,凌晨0点的话就是：现在时间是十二点（提示用select case语句或者 使用split语句）,12小时制，从一点到十二点

2. 数组转化成字符串arr(0)="a",arr(1)="b",arr(26)="z" A,C,D...X,Y,Z

3. 截取字符串长度考虑汉字（比如允许20个字母，但是却只能允许10个汉字，因为1个汉字占2个字母的字节长度）

4. 计算一个字符串在另外一个字符串中出现的次数

5. 解决cint和clng溢出问题,提示'cint返回的是 Integer 子类型的 Variant:-32,768 到 32,767 之间,'clng返回的是 Long 子类型的 Variant:-2,147,483,648 到 2,147,483,647 之间

6. 把一个数的小点后面的部分去掉,数字可能超大比如21424123411234127349801273491234.12

7. 将一个数字135487986546转化为135,487,986,546的形式显示，不用FormatNumber函数

8. 输入一个汉字，显示拼音声母，例“水”显示”S”

9. 如何把阿拉伯小写数字(包括小数)1234123.23转化成英文:one million, two hundred and thirty-four thousand, one hundred and twenty-three point two three

10. 把1234123412.21元转化成大写的人民币:壹拾贰亿叁仟肆佰壹拾贰万叁仟肆佰壹拾贰元贰角壹分

11. 数字日期转化为汉字日期格式,例: 1993-3-8转为：一九九三年三月八日

12. 生成一个8位的随机密码：ASkasddK

13. 计算当前日期属于定义时间段内的第几个星期

14. 给北京2008搞个倒计时(不是即时走，是刷新一下，时间变一下

15. 用于用户发布的各种信息过滤，带脏话过滤和Html过滤

16. 写一个输出不同风格时间的排列方式函数:2005-05-01或2005.05.01或2005年5月1日

17. 去掉字符串开头的连续的回车和空格

18. 给一个email地址,判断是否有效,返回1表示正确

19. 检测一个URL是否合法，如果没有http://则自动加上，例：www.cc..com 或http://ddd/

20. 如何获得上个月份是几月？

1\ 编译中的问题：全局变量如int i=5; int\*(pf)()=foo; 分别在何时被初始化？设计时候如何具体的实现。

2\ 解释例如mov ax,100H 这样一条指令的cpu, os, memory等都完成了什么样的工作。

3\ Strlen（）的C语言实现，不能使用任何变量。

4\ C语言参数的入栈顺序？为什么这么实现？

5\ C语言中字符串的翻转，最高效率（时间和空间）的实现？

6\ 4. 某个程序在一个嵌入式系统(200M的CPU,50M的SDRAM)中已经最化了，换到另一个系统

(300M的CPU,50M的SDRAM)中运行，还需要优化吗？

7 \有两个线程

void producer()

{

while(1)

{

GeneratePacket();

PutPacketIntoBuffer();

Signal(customer);

}

}

void customer()

{

while(1)

{

WaitForSignal();

if(PacketInBuffer>10)

{

ReadAllPackets();

ProcessPackets();

}

}

}

（1）有没有其他方法可以提高程序的性能

（2）可不可以不使用信号之类的机制来实现上述的功能

.net笔试题

姓名： 日期：

1. 填空: (1)面向对象的语言具有\_\_\_\_\_\_\_\_性、\_\_\_\_\_\_\_\_\_性、\_\_\_\_\_\_\_\_性。

(2)能用foreach遍历访问的对象需要实现 \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_接口或声明\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_方法的类型。

(3)列举ADO.net中的五个主要对象\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_、\_\_\_\_\_\_\_\_\_\_\_\_\_、\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_、\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_、\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_。

2. 不定项选择：

(1) 以下叙述正确的是：

A. 接口中可以有虚方法。 B. 一个类可以实现多个接口。

C. 接口不能被实例化。 D. 接口中可以包含已实现的方法。

(2) 从数据库读取记录，你可能用到的方法有：

A. ExecuteNonQuery B. ExecuteScalar

C. Fill D. ExecuteReader

3. 简述 private、 protected、 public、 internal 修饰符的访问权限。

4. 写出一条Sql语句： 取出表A中第31到第40记录（SQLServer, 以自动增长的ID作为主键, 注意：ID可能不是连续的。）

5 .列举ASP.NET 页面之间传递值的几种方式。

6. 写出程序的输出结果

class Class1 {

private string str = "Class1.str";

private int i = 0;

static void StringConvert(string str) {

str = "string being converted.";

}

static void StringConvert(Class1 c) {

c.str = "string being converted.";

}

static void Add(int i) {

i++;

}

static void AddWithRef(ref int i) {

i++;

}

static void Main() {

int i1 = 10;

int i2 = 20;

1. 装箱、拆箱操作发生在什么时候？
2. 什么时候可以用foreach语句代替while循环？
3. .net framwork通过什么与非托览代码进行交互操作
4. 以下代码能否通过？请给出理由

Try{}

Catch (file not found exception (1){}

Catch (file not found exception (2){}

Catch (file not found exception (3){}

Catch{}

1. 请问string类与scringbuilder类有什么区别？分别适用在什么情况下？
2. 强命名程序集和弱命名程序集在使用上有什么区别？
3. int和int16有什么区别？
4. C#中方法的参数引用有哪几种方式？
5. 说一下自己的职业道路。
6. 怎么样在网上写一个页面发email？
7. SQL里的数据转换服务是什么？
8. 从SQL里取出一条记录，邦定DATAGIRD。
9. 写个登录页面。
10. 你想在一个怎么样的工作环境下工作？

1.如何用Asp判断你的网站的虚拟物理路径   
答：使用Mappath方法   
< p align="center" >< font size="4" face="Arial" >< b >   
The Physical path to this virtual website is:   
< /b >< /font >   
< font color="#FF0000" size="6" face="Arial" >   
< %= Server.MapPath("\")% >   
< /font >< /p >   
2.我如何知道使用者所用的浏览器？   
答：使用the Request object方法   
strBrowser=Request.ServerVariables("HTTP\_USER\_AGENT")   
If Instr(strBrowser,"MSIE") < > 0 Then   
Response.redirect("ForMSIEOnly.htm")   
Else   
Response.redirect("ForAll.htm")   
End If   
  
3.如何计算每天的平均反复访问人数   
答：解决方法   
< % startdate=DateDiff("d",Now,"01/01/1990")   
if strdate< 0 then startdate=startdate\*-1   
avgvpd=Int((usercnt)/startdate) % >   
显示结果   
< % response.write(avgvpd) % >   
that is it.this page have been viewed since November 10,1998   
  
4.如何显示随机图象   
< % dim p,ppic,dpic   
ppic=12   
randomize   
p=Int((ppic\*rnd)+1)   
dpic="graphix/randompics/"&p&".gif"   
% >   
显示   
< img src="< %=dpic% >" >   
  
5.如何回到先前的页面   
答：< a href="< %=request.serverVariables("Http\_REFERER")% >" >preivous page< /a >   
或用图片如：< img src="arrowback.gif" alt="< %=request.serverVariables("HTTP\_REFERER")% >" >   
  
6.如何确定对方的IP地址   
答：< %=Request.serverVariables("REMOTE\_ADDR)% >   
  
7.如何链结到一副图片上   
答：< % @Languages=vbs cript % >   
< % response.expires=0   
strimagename="graphix/errors/erroriamge.gif"   
response.redirect(strimagename)   
% >   
  
8.强迫输入密码对话框   
答：把这句话放载页面的开头   
< % response.status="401 not Authorized"   
response.end   
% >   
  
9.如何传递变量从一页到另一页   
答：用 HIDDEN 类型来传递变量   
< % form method="post" action="mynextpage.asp" >   
< % for each item in request.form % >   
< input namee="< %=item% >" type="HIDDEN"   
value="< %=server.HTMLEncode(Request.form(item)) % >" >   
< % next % >   
< /form >   
  
10.为何我在 asp 程序内使用 msgbox，程序出错说没有权限   
答：由于 asp 是服务器运行的，如果可以在服务器显示一个对话框，那么你只好等有人按了确定之后，你的程序才能继续执行，而一般服务器不会有人守着，所以微软不得不禁止这个函数，并胡乱告诉你 (:) 呵呵) 没有权限。但是ASP和客户端脚本结合倒可以显示一个对话框，as follows:   
< % yourVar="测试对话框"% >   
< % s cript language='javas cript' >   
alert("< %=yourvar% >")   
< /s cript >   
  
11.有没有办法保护自己的源代码，不给人看到   
答：可以去下载一个微软的Windows s cript Encoder，它可以对asp的脚本和客户端javas cript/vbs cript脚本进行加密。。。不过客户端加密后，只有ie5才能执行，服务器端脚本加密后，只有服务器上安装有s cript engine 5（装一个ie5就有了）才能执行。   
  
12.怎样才能将 query string 从一个 asp 文件传送到另一个？   
答：前者文件加入下句： Response.Redirect("second.asp?" & Request.ServerVariables("QUERY\_STRING"))   
  
13.global.asa文件总是不起作用?   
答：只有web目录设置为web application, global.asa才有效，并且一个web application的根目录下 global.asa才有效。IIS4可以使用Internet Service Manager设置application setting 怎样才能使得htm文件如同asp文件一样可以执行脚本代码？   
  
14.怎样才能使得htm文件如同asp文件一样可以执行脚本代码？   
答：Internet Sevices Manager - > 选择default web site - >右鼠键- >菜单属性-〉主目录- > 应用程序设置（Application Setting）- > 点击按钮 "配置"- > app mapping - >点击按钮"Add" - > executable browse选择 \WINNT\SYSTEM32\INETSRV\ASP.DLL EXTENSION 输入 htm method exclusions 输入PUT.DELETE 全部确定即可。但是值得注意的是这样对htm也要由asp.dll处理，效率将降低。   
  
15.如何注册组件   
答：有两种方法。   
第一种方法：手工注册 DLL 这种方法从IIs 3.0一直使用到IIs 4.0和其它的Web Server。它需要你在命令行方式下来执行，进入到包含有DLL的目录，并输入：regsvr32 component\_name.dll 例如 c:\temp\regsvr32 AspEmail.dll 它会把dll的特定信息注册入服务器中的注册表中。然后这个组件就可以在服务器上使用了，但是这个方法有一个缺陷。当使用这种方法注册完毕组件后，该组件必须要相应的设置NT的匿名帐号有权限执行这个dll。特别是一些组件需要读取注册表，所以，这个注册组件的方法仅仅是使用在服务器上没有MTS的情况下，要取消注册这个dll，使用：regsvr32 /u aspobject.dll example c:\temp\regsvr32 /u aneiodbc.dll   
  
第二种方法：使用MTS(Microsoft Transaction Server) MTS是IIS 4新增特色，但是它提供了巨大的改进。MTS允许你指定只有有特权的用户才能够访问组件，大大提高了网站服务器上的安全性设置。在MTS上注册组件的步骤如下：   
1) 打开IIS管理控制台。   
2) 展开transaction server，右键单击"pkgs installed"然后选择"new package"。   
3) 单击"create an empty package"。   
4) 给该包命名。   
5) 指定administrator帐号或则使用"interactive"（如果服务器经常是使用administrator 登陆的话）。   
6) 现在使用右键单击你刚建立的那个包下面展开后的"components"。选择 "new then component"。   
7) 选择 "install new component" 。   
8) 找到你的.dll文件然后选择next到完成。   
要删除这个对象，只要选择它的图标，然后选择delete。   
附注：特别要注意第二种方法，它是用来调试自己编写组件的最好方法，而不必每次都需要重新启动机器了。   
  
16. ASP与Access数据库连接：   
  
<%@ language=VBs cript%>   
<%   
dim conn,mdbfile   
mdbfile=server.mappath("数据库名称.mdb")   
set conn=server.createobject("adodb.connection")   
conn.open "driver={microsoft access driver (\*.mdb)};uid=admin;pwd=数据库密码;dbq="&mdbfile   
  
%>   
  
17. ASP与SQL数据库连接：   
  
<%@ language=VBs cript%>   
<%   
dim conn   
set conn=server.createobject("ADODB.connection")   
con.open "PROVIDER=SQLOLEDB;DATA SOURCE=SQL服务器名称或IP地址;UID=sa;PWD=数据库密码;DATABASE=数据库名称   
%>   
  
建立记录集对象：   
  
set rs=server.createobject("adodb.recordset")   
rs.open SQL语句,conn,3,2   
  
18. SQL常用命令使用方法：   
  
(1) 数据记录筛选：   
  
sql="select \* from 数据表 where 字段名=字段值 order by 字段名 [desc]"   
  
sql="select \* from 数据表 where 字段名 like '%字段值%' order by 字段名 [desc]"   
  
sql="select top 10 \* from 数据表 where 字段名 order by 字段名 [desc]"   
  
sql="select \* from 数据表 where 字段名 in ('值1','值2','值3')"   
  
sql="select \* from 数据表 where 字段名 between 值1 and 值2"   
  
(2) 更新数据记录：   
  
sql="update 数据表 set 字段名=字段值 where 条件表达式"   
  
sql="update 数据表 set 字段1=值1,字段2=值2 …… 字段n=值n where 条件表达式"   
  
(3) 删除数据记录：   
  
sql="delete from 数据表 where 条件表达式"   
  
sql="delete from 数据表" (将数据表所有记录删除)   
  
(4) 添加数据记录：   
  
sql="insert into 数据表 (字段1,字段2,字段3 …) valuess (值1,值2,值3 …)"   
  
sql="insert into 目标数据表 select \* from 源数据表" (把源数据表的记录添加到目标数据表)   
  
(5) 数据记录统计函数：   
  
AVG(字段名) 得出一个表格栏平均值   
COUNT(\*|字段名) 对数据行数的统计或对某一栏有值的数据行数统计   
MAX(字段名) 取得一个表格栏最大的值   
MIN(字段名) 取得一个表格栏最小的值   
SUM(字段名) 把数据栏的值相加   
  
引用以上函数的方法：   
  
sql="select sum(字段名) as 别名 from 数据表 where 条件表达式"   
set rs=conn.excute(sql)   
  
用 rs("别名") 获取统的计值，其它函数运用同上。   
  
(5) 数据表的建立和删除：   
  
CREATE TABLE 数据表名称(字段1 类型1(长度),字段2 类型2(长度) …… )   
  
例：CREATE TABLE tab01(name varchar(50),datetime default now())   
  
DROP TABLE 数据表名称 (永久性删除一个数据表)   
  
19. 记录集对象的方法：   
  
rs.movenext 将记录指针从当前的位置向下移一行   
rs.moveprevious 将记录指针从当前的位置向上移一行   
rs.movefirst 将记录指针移到数据表第一行   
rs.movelast 将记录指针移到数据表最后一行   
rs.absoluteposition=N 将记录指针移到数据表第N行   
rs.absolutepage=N 将记录指针移到第N页的第一行   
rs.pagesize=N 设置每页为N条记录   
rs.pagecount 根据 pagesize 的设置返回总页数   
rs.recordcount 返回记录总数   
rs.bof 返回记录指针是否超出数据表首端，true表示是，false为否   
rs.eof 返回记录指针是否超出数据表末端，true表示是，false为否   
rs.delete 删除当前记录，但记录指针不会向下移动   
rs.addnew 添加记录到数据表末端   
rs.update 更新数据表记录   
  
－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－－   
  
20 Recordset对象方法   
  
Open方法   
  
recordset.Open Source,ActiveConnection,CursorType,LockType,Options   
  
Source   
Recordset对象可以通过Source属性来连接Command对象。Source参数可以是一个Command对象名称、一段SQL命令、一个指定的数据表名称或是一个Stored Procedure。假如省略这个参数，系统则采用Recordset对象的Source属性。   
  
ActiveConnection   
Recordset对象可以通过ActiveConnection属性来连接Connection对象。这里的ActiveConnection可以是一个Connection对象或是一串包含数据库连接信息（ConnectionString）的字符串参数。   
  
CursorType   
Recordset对象Open方法的CursorType参数表示将以什么样的游标类型启动数据，包括adOpenForwardOnly、adOpenKeyset、adOpenDynamic及adOpenStatic，分述如下：   
--------------------------------------------------------------   
常数 常数值 说明   
-------------------------------------------------------------   
adOpenForwardOnly 0 缺省值，启动一个只能向前移动的游标（Forward Only）。   
adOpenKeyset 1 启动一个Keyset类型的游标。   
adOpenDynamic 2 启动一个Dynamic类型的游标。   
adOpenStatic 3 启动一个Static类型的游标。   
-------------------------------------------------------------   
以上几个游标类型将直接影响到Recordset对象所有的属性和方法，以下列表说明他们之间的区别。   
  
-------------------------------------------------------------   
Recordset属性 adOpenForwardOnly adOpenKeyset adOpenDynamic adOpenStatic   
-------------------------------------------------------------   
AbsolutePage 不支持 不支持 可读写 可读写   
AbsolutePosition 不支持 不支持 可读写 可读写   
ActiveConnection 可读写 可读写 可读写 可读写   
BOF 只读 只读 只读 只读   
Bookmark 不支持 不支持 可读写 可读写   
CacheSize 可读写 可读写 可读写 可读写   
CursorLocation 可读写 可读写 可读写 可读写   
CursorType 可读写 可读写 可读写 可读写   
EditMode 只读 只读 只读 只读   
EOF 只读 只读 只读 只读   
Filter 可读写 可读写 可读写 可读写   
LockType 可读写 可读写 可读写 可读写   
MarshalOptions 可读写 可读写 可读写 可读写   
MaxRecords 可读写 可读写 可读写 可读写   
PageCount 不支持 不支持 只读 只读   
PageSize 可读写 可读写 可读写 可读写   
RecordCount 不支持 不支持 只读 只读   
Source 可读写 可读写 可读写 可读写   
State 只读 只读 只读 只读   
Status 只读 只读 只读 只读   
AddNew 支持 支持 支持 支持   
CancelBatch 支持 支持 支持 支持   
CancelUpdate 支持 支持 支持 支持   
Clone 不支持 不支持   
Close 支持 支持 支持 支持   
Delete 支持 支持 支持 支持   
GetRows 支持 支持 支持 支持   
Move 不支持 支持 支持 支持   
MoveFirst 支持 支持 支持 支持   
MoveLast 不支持 支持 支持 支持   
MoveNext 支持 支持 支持 支持   
MovePrevious 不支持 支持 支持 支持   
NextRecordset 支持 支持 支持 支持   
Open 支持 支持 支持 支持   
Requery 支持 支持 支持 支持   
Resync 不支持 不支持 支持 支持   
Supports 支持 支持 支持 支持   
Update 支持 支持 支持 支持   
UpdateBatch 支持 支持 支持 支持   
--------------------------------------------------------------   
其中NextRecordset方法并不适用于Microsoft Access数据库。   
  
LockType   
Recordset对象Open方法的LockType参数表示要采用的Lock类型，如果忽略这个参数，那么系统会以Recordset对象的LockType属性为预设值。LockType参数包含adLockReadOnly、adLockPrssimistic、adLockOptimistic及adLockBatchOptimistic等，分述如下：   
  
-------------------------------------------------------------   
常数 常数值 说明   
--------------------------------------------------------------   
adLockReadOnly 1 缺省值，Recordset对象以只读方式启动，无法运行AddNew、Update及Delete等方法   
adLockPrssimistic 2 当数据源正在更新时，系统会暂时锁住其他用户的动作，以保持数据一致性。   
adLockOptimistic 3 当数据源正在更新时，系统并不会锁住其他用户的动作，其他用户可以对数据进行增、删、改的操作。   
adLockBatchOptimistic 4 当数据源正在更新时，其他用户必须将CursorLocation属性改为adUdeClientBatch才能对数据进行增、   
删、改的操作。

<%   
'Asp连SQLServer数据库(本地数据库)   
dim conn,connstr   
  
connstr="Provider = Sqloledb; User ID =sa; Password =123123; Initial Catalog =vote; Data Source =(local);"   
Set conn = Server.CreateObject("ADODB.Connection")   
conn.Open connstr   
If Err then   
Err.Clear   
Set conn = Nothing   
Response.Write "系统调整中"   
Response.End   
End If   
%>   
  
<%   
'Asp连SQLServer数据库(远程数据库)   
dim conn,connstr   
  
connstr="Provider = Sqloledb; User ID =hm; Password =newfeel; Initial Catalog =vote; Data Source =192.168.168.38;"   
Set conn = Server.CreateObject("ADODB.Connection")   
conn.Open connstr   
If Err then   
Err.Clear   
Set conn = Nothing   
Response.Write "系统调整中"   
Response.End   
End If   
%>   
  
  
<%   
dim MM\_gbook\_STRING,db,conn   
db="database/Shop.mdb" '数据库路径及名称   
Set conn = Server.CreateObject("ADODB.Connection")   
MM\_gbook\_STRING="Provider=Microsoft.Jet.OLEDB.4.0;Data Source=" & Server.MapPath(""&db&"")   
conn.Open MM\_gbook\_STRING   
%>   
  
'带密码   
dim conn   
set conn=server.CreateObject("adodb.connection")   
conn.open "provider=microsoft.jet.oledb.4.0;data source="&Server.MapPath("mydata.mdb")&";uid=id;pwd=pwd"

1. C#是否支持多继承，并阐述理由。
2. 详细阐述cookie 、viewstate、session之间的区别，以及其各自工作原理。
3. 在C#中利用SOCKET进行网络通信编程的一般步骤是什么？
4. int[][] myarray3=new int[3][]{new int[3]{5,6,2},new int[5]{6,9,7,8,3},new int[2]{3,2}};myarray3[2][2]的值是（ ）。
5. 您要创建asp.net应用程序用于运行某公司内部的web站点，这个应用程序包含了50个页面。您想要配置这个应用程序以便当发生一个http代码错误时它可以显示一个自定义的错误页面给用户。您想要花最少的代价完成这些目标，您应该怎么做？
6. 是否了解URL Rewrite？请简要说明其原理和在asp.net中的实现方式。
7. .net2.0和.net1.1比较有哪些区别？
8. 用C#写一段选择排序算法，要求用自己的编程风格。

1 session和cookies的本质区别是什么？  
2 request对象可以访问任何基于http请求传送的消息，他的查找顺寻是什么？（是querystring,form,servervariables,clientcertificate,cookies）  
3 delete和turncate table有何区别？  
4．有100个和尚吃100个馒头，大和尚1人吃4个，小和尚4人吃1个，问有多少个大和尚和多少个小和尚？

5．请使用ASP提供的计数器组件，在页面上显示“您是第”X “位访客”。（只要写出主要代码即可）

6．假设有数据库address,内有表link,该表中包含字段：姓名（name varchar(50)）,电话(tel varchar(20)),email(email varchar(50))，请分页将link表中的数据显示出来，且每页显示记录8条。

1. 用uinl和C#描述责任链。
2. 填空题：二分算法a【n】找出x值如果没找到返回-1用工具建模与每天如何编码测试（ ）？
3. 逻辑题：有一根材质均匀的绳子，要烧一个小时才能烧完。那有多根这样的绳子，如何判断半小时？一小时十五分？
4. 请叙述类与结构区别。
5. ASP.NET题

在一个asp项目中，引用一个webservice需要怎么做？

6. 采用C#语言及面向对象的方法，设计一个几何图形面积计算的程序，要求能够实现对正三角形、圆形、正方形面积的计算。用户输入形状、边长（半径），程序出处形状的面积。

一、问答

1、实模式与保护模式。为什么要设计这两种模式？好处在什么地方？分别写出各自寻址的

过程。

2、请阅读以下一段程序，并给出答案。

class A

{ public：

A(){ doSth() }

virtual void doSth(){ printf("I am A");}

}

class B：public A

{ public：

virtual void doSth(){ printf("I am B");}

}

B b;

执行结果是什么？为什么？

3、在STL的应用中 map这种key-value的应用很多，如果key的类型是GUID，该如

何处理？

4、一个内存变量a=5，有5个线程需要对其进行操作，其中3个对a进行加1操作，2个对a进

行减1操作，为了保证能够得到正常结果6，需要使用什么方法？（列出越多越好）

5、描述并比较以下对象：事件，信标，临界区，互斥对象。

6、cdecl、stdcall、fastcall是什么？哪种可以实现个数不定的入口参数，为什么？

二、程序设计（以下题目请写出实现代码）

1、有一段文本，统计其中的单词数。例如：

As a technology , "HailStorm" is so new that it is still only known by its

code name.

注意：单词间的间隔不一定是一个空格

2、国际象棋有8×8格，每个格子可放一个棋子。皇后的规则是可以横、竖、斜移动。在一

个棋盘放置8个皇后，并使它们互相无法威胁到彼此。

3、输入二个64位的十进制数，计算相乘之后的乘积。

1.分别写出BOOL,int,float,指针类型的变量a 与“零”的比较语句。

答案：

BOOL : if ( !a ) or if(a)

int : if ( a == 0)

float : const EXPRESSION EXP = 0.000001

if ( a < EXP && a >-EXP)

pointer : if ( a != NULL) or if(a == NULL)

2.请说出const与#define 相比，有何优点？

答案：1） const 常量有数据类型，而宏常量没有数据类型。编译器可以对前者进行类型安全检查。而对后者只进行字符替换，没有类型安全检查，并且在字符替换可能会产生意料不到的错误。

2） 有些集成化的调试工具可以对const 常量进行调试，但是不能对宏常量进行调试。

3.简述数组与指针的区别？

数组要么在静态存储区被创建（如全局数组），要么在栈上被创建。指针可以随时指向任意类型的内存块。

(1)修改内容上的差别

char a[] = “hello”;

a[0] = ‘X’;

char \*p = “world”; // 注意p 指向常量字符串

p[0] = ‘X’; // 编译器不能发现该错误，运行时错误

(2) 用运算符sizeof 可以计算出数组的容量（字节数）。sizeof(p),p 为指针得到的是一个指针变量的字节数，而不是p 所指的内存容量。C++/C 语言没有办法知道指针所指的内存容量，除非在申请内存时记住它。注意当数组作为函数的参数进行传递时，该数组自动退化为同类型的指针。

char a[] = "hello world";

char \*p = a;

cout<< sizeof(a) << endl; // 12 字节

cout<< sizeof(p) << endl; // 4 字节

计算数组和指针的内存容量

void Func(char a[100])

{

cout<< sizeof(a) << endl; // 4 字节而不是100 字节

}

4.类成员函数的重载、覆盖和隐藏区别？

答案：

a.成员函数被重载的特征：

（1）相同的范围（在同一个类中）；

（2）函数名字相同；

（3）参数不同；

（4）virtual 关键字可有可无。

b.覆盖是指派生类函数覆盖基类函数，特征是：

（1）不同的范围（分别位于派生类与基类）；

（2）函数名字相同；

（3）参数相同；

（4）基类函数必须有virtual 关键字。

c.“隐藏”是指派生类的函数屏蔽了与其同名的基类函数，规则如下：

（1）如果派生类的函数与基类的函数同名，但是参数不同。此时，不论有无virtual关键字，基类的函数将被隐藏（注意别与重载混淆）。

（2）如果派生类的函数与基类的函数同名，并且参数也相同，但是基类函数没有virtual 关键字。此时，基类的函数被隐藏（注意别与覆盖混淆）

5. There are two int variables: a and b, don’t use “if”, “? :”, “switch”or other judgement statements, find out the biggest one of the two numbers.

答案：( ( a + b ) + abs( a - b ) ) / 2

6. 如何打印出当前源文件的文件名以及源文件的当前行号？

答案：

cout << \_\_FILE\_\_ ;

cout<<\_\_LINE\_\_ ;

\_\_FILE\_\_和\_\_LINE\_\_是系统预定义宏，这种宏并不是在某个文件中定义的，而是由编译器定义的。

7. main 主函数执行完毕后，是否可能会再执行一段代码，给出说明？

答案：可以，可以用\_onexit 注册一个函数，它会在main 之后执行int fn1(void), fn2(void), fn3(void), fn4 (void);

void main( void )

{

String str("zhanglin");

\_onexit( fn1 );

\_onexit( fn2 );

\_onexit( fn3 );

\_onexit( fn4 );

printf( "This is executed first.\n" );

}

int fn1()

{

printf( "next.\n" );

return 0;

}

int fn2()

{

printf( "executed " );

return 0;

}

int fn3()

{

printf( "is " );

return 0;

}

int fn4()

{

printf( "This " );

return 0;

}

The \_onexit function is passed the address of a function (func) to be called when the program terminates normally. Successive calls to \_onexit create a register of functions that are executed in LIFO (last-in-first-out) order. The functions passed to \_onexit cannot take parameters.

8. 如何判断一段程序是由C 编译程序还是由C++编译程序编译的？

答案：

#ifdef \_\_cplusplus

cout<<"c++";

#else

cout<<"c";

#endif

9.文件中有一组整数，要求排序后输出到另一个文件中

答案：

＃i nclude

＃i nclude

using namespace std;

void Order(vector& data) //bubble sort

{

int count = data.size() ;

int tag = false ; // 设置是否需要继续冒泡的标志位

for ( int i = 0 ; i < count ; i++)

{

for ( int j = 0 ; j < count - i - 1 ; j++)

{

if ( data[j] > data[j+1])

{

tag = true ;

int temp = data[j] ;

data[j] = data[j+1] ;

data[j+1] = temp ;

}

}

if ( !tag )

break ;

}

}

void main( void )

{

vectordata;

ifstream in("c:\\data.txt");

if ( !in)

{

cout<<"file error!";

exit(1);

}

int temp;

while (!in.eof())

{

in>>temp;

data.push\_back(temp);

}

in.close(); //关闭输入文件流

Order(data);

ofstream out("c:\\result.txt");

if ( !out)

{

cout<<"file error!";

exit(1);

}

for ( i = 0 ; i < data.size() ; i++)

out<<DATA[I]<<" ?;

10. 链表题：一个链表的结点结构

struct Node

{

int data ;

Node \*next ;

};

typedef struct Node Node ;

(1)已知链表的头结点head,写一个函数把这个链表逆序 ( Intel)

Node \* ReverseList(Node \*head) //链表逆序

{

if ( head == NULL || head->next == NULL )

return head;

Node \*p1 = head ;

Node \*p2 = p1->next ;

Node \*p3 = p2->next ;

p1->next = NULL ;

while ( p3 != NULL )

{

p2->next = p1 ;

p1 = p2 ;

p2 = p3 ;

p3 = p3->next ;

}

p2->next = p1 ;

head = p2 ;

return head ;

}

(2)已知两个链表head1 和head2 各自有序，请把它们合并成一个链表依然有序。(保留所有结点，即便大小相同）

Node \* Merge(Node \*head1 , Node \*head2)

{

if ( head1 == NULL)

return head2 ;

if ( head2 == NULL)

return head1 ;

Node \*head = NULL ;

Node \*p1 = NULL;

Node \*p2 = NULL;

if ( head1->data < head2->data )

{

head = head1 ;

p1 = head1->next;

p2 = head2 ;

}

else

{

head = head2 ;

p2 = head2->next ;

p1 = head1 ;

}

Node \*pcurrent = head ;

while ( p1 != NULL && p2 != NULL)

{

if ( p1->data <= p2->data )

{

pcurrent->next = p1 ;

pcurrent = p1 ;

p1 = p1->next ;

}

else

{

pcurrent->next = p2 ;

pcurrent = p2 ;

p2 = p2->next ;

}

}

if ( p1 != NULL )

pcurrent->next = p1 ;

if ( p2 != NULL )

pcurrent->next = p2 ;

return head ;

}

(3)已知两个链表head1 和head2 各自有序，请把它们合并成一个链表依然有序，这次要求用递归方法进行。 (Autodesk)

答案：

Node \* MergeRecursive(Node \*head1 , Node \*head2)

{

if ( head1 == NULL )

return head2 ;

if ( head2 == NULL)

return head1 ;

Node \*head = NULL ;

if ( head1->data < head2->data )

{

head = head1 ;

head->next = MergeRecursive(head1->next,head2);

}

else

{

head = head2 ;

head->next = MergeRecursive(head1,head2->next);

}

return head ;

}

1.面向对象的思想主要包括什么？

答:这个题范围太广,不知道说什么.

任何事物都可以理解为对象，再具体谈到： 继承。封装。多态。

2.什么是ASP.net中的用户控件

答:用户控件就是.ascx扩展名的东西喽,可以拖到不同的页面中调用,以节省代码.比如登陆可能在多个页面上有,就可以做成用户控件,但是有一个问题就是用户控件拖到不同级别的目录下后里面的图片等的相对路径会变得不准确,需要自已写方法调整.

3.什么叫应用程序域？什么是受管制的代码？什么是强类型系统？什么是装箱和拆箱？什么是重载？CTS、CLS和CLR分别作何解释？

答:装箱就是把值类型转成引用类型,从MS IL角度看好像是boxing,没记错的话是把值从堆栈转到堆中.拆箱相反,重载就是指一个方法名同,参数个数不同,返回值可以相同的方法.CLR是通用语言运行时,其它的不清楚.

4.列举一下你所了解的XML技术及其应用

答:XML可是好东西,保存配置,站与站之间的交流,WEB SERVICE。以及与数据库的数据交互等地方都要用它.

5.值类型和引用类型的区别？写出C#的样例代码。

答:结构是值类型,类是引用类型,所以传结构就是值类型的应用啦,传对象或类就是引用类型的,这个不用多写了吧.

6.ADO.net中常用的对象有哪些？分别描述一下。

答:connection command sqladapter dataset datatable dataview等等.写不完了.

7.如何理解委托？

答:据说相当于函数指针,定义了委托就可以在不调用原方法名称的情况下调用那个方法.

msdn2005中是这样解释的:

委托具有以下特点：

委托类似于 C++ 函数指针，但它是类型安全的。

委托允许将方法作为参数进行传递。

委托可用于定义回调方法。

委托可以链接在一起；例如，可以对一个事件调用多个方法。

方法不需要与委托签名精确匹配。有关更多信息，请参见协变和逆变。

C# 2.0 版引入了匿名方法的概念，此类方法允许将代码块作为参数传递，以代替单独定义的方法。

8.C#中的接口和类有什么异同。

答:这个异同可多了,要说清楚还真不容易.（可从两者间的可继承性方面来谈）

9.。net中读写数据库需要用到哪些类？他们的作用

答:这个类自已可以写的啊,你是指基类吗?那configuration,sqlconnection,sqlcommand等都要用到.

10.UDP连接和TCP连接的异同。

答:前者只管传,不管数据到不到,无须建立连接.后者保证传输的数据准确,须要连结.

11.ASP.net的身份验证方式有哪些？分别是什么原理？

答:form认证,windows集成认证等,原理不清楚.

12.进程和线程分别怎么理解？

答:进程是老子,线程是儿子,没有老子就没有儿子,一个老子可以有多个儿子.一个儿子可以成为别人的儿子,一个老子也可以为别的老子生儿子.

13.什么是code-Behind技术。

答:代码分离,这是个明智的东西,像ASP这样混成一堆很不爽.或者可以理解成HTML代码写在前台,C#代码写在后台.当然前台也有脚本,类的调用等,其实写在一起也是可以的.

14.活动目录的作用。

答:这个不明白.请明白的补充一下.

15.net中读写XML的类都归属于哪些命名空间？

答:System.Xml

我自已写的就不一定了,嘿嘿.

16.解释一下UDDI、WSDL的意义及其作用。

答:什么东西?

17.什么是SOAP,有哪些应用。

答:SOAP（Simple Object Access Protocol ）简单对象访问协议是在分散或分布式的环境中交换信息并执行远程过程调用的协议，是一个基于XML的协议。使用SOAP，不用考虑任何特定的传输协议（最常用的还是HTTP协议），可以允许任何类型的对象或代码，在任何平台上，以任何一直语言相互通信。这种相互通信采用的是XML格式的消息,具体请看:http://playist.blogchina.com/2521621.html

18.如何部署一个ASP.net页面。

答:随便啦,想直接上传就直接上传,想打包成EXE就打包,看个人喜好.

19.如何理解.net中的垃圾回收机制。

答:GC?对象创建了总要清除啊,不然内存哪够用?

20.常用的调用webservice方法有哪些？

答:调用就调用,还有很多方法吗?

第一，谈谈final, finally, finalize的区别。

第二，Anonymous Inner Class (匿名内部类) 是否可以extends(继承)其它类，是否可以implements(实现)interface(接口)?

第三，Static Nested Class 和 Inner Class的不同，说得越多越好(面试题有的很笼统)。

第四，&和&&的区别。

第五，HashMap和Hashtable的区别。

第六，Collection 和 Collections的区别。

第七，什么时候用assert。

第八，GC是什么? 为什么要有GC?

第九，String s = new String("xyz");创建了几个String Object?

第十，Math.round(11.5)等於多少? Math.round(-11.5)等於多少?

第十一，short s1 = 1; s1 = s1 + 1;有什么错? short s1 = 1; s1 += 1;有什么错?

第十二，sleep() 和 wait() 有什么区别?

第十三，Java有没有goto?

第十四，数组有没有length()这个方法? String有没有length()这个方法?

第十五，Overload和Override的区别。Overloaded的方法是否可以改变返回值的类型?

第十六，Set里的元素是不能重复的，那么用什么方法来区分重复与否呢? 是用==还是equals()? 它们有何区别?

第十七，给我一个你最常见到的runtime exception。

第十八，error和exception有什么区别?

第十九，List, Set, Map是否继承自Collection接口?

第二十，abstract class和interface有什么区别?

第二十一，abstract的method是否可同时是static,是否可同时是native，是否可同时是synchronized?

第二十二，接口是否可继承接口? 抽象类是否可实现(implements)接口? 抽象类是否可继承实体类(concrete class)?

第二十三，启动一个线程是用run()还是start()?

第二十四，构造器Constructor是否可被override?

第二十五，是否可以继承String类?

第二十六，当一个线程进入一个对象的一个synchronized方法后，其它线程是否可进入此对象的其它方法?

第二十七，try {}里有一个return语句，那么紧跟在这个try后的finally {}里的code会不会被执行，什么时候被执行，在return前还是后?

第二十八，编程题: 用最有效率的方法算出2乘以8等於几?

第二十九，两个对象值相同(x.equals(y) == true)，但却可有不同的hash code，这句话对不对?

第三十，当一个对象被当作参数传递到一个方法后，此方法可改变这个对象的属性，并可返回变化后的结果，那么这里到底是值传递还是引用传递?

第三十一，swtich是否能作用在byte上，是否能作用在long上，是否能作用在String上?

第三十二，编程题: 写一个Singleton出来。

1.写一个实现对一段字符串翻转的方法，附加一些条件，如其中包括“，”、“.”，对其设计测试用 例 。

2.对一支纸杯设计测试用例（可以是广义的杯，不一定是某一支特定功能的杯）

开发语言概念题

3.什么是反射？

4.用Singleton如何写设计模式

5.C#中的垃圾回收机制是怎样的？

6.什么是Application Pool？

7.链表和数组的区别，各有什么优缺点.

8.Remoting在客户端服务器怎么实现 ?

9.什么是友元函数？

10.用标准C如何实现多态？

11.什么是虚函数？

12.什么是抽象函数？

13.什么是内存泄漏，怎样最简单的方法判断内存泄露？

第一题：ASP中，VBScript的唯一的数据类型是什么？

第二题：在ASP中，VBScript有多种控制程序流程语句，如If…Then, Select… Case,

For … Next, Do … Loop, Exit等语句。请为这五个语句分别写一段使用的代码。

第三题：请看如下代码

<%

TestString="Test"

TestA

TestB

Response.write TestString

Sub TestA()

TestString="TestA"

End Sub

Sub TestB()

Dim TestString

TestString="TestB"

End Sub

%>

这段代码执行后，运行结果是什么？并解释一下为什么？

第四题：在ASP中，Server中有一个方法是URLEncode(string)

如: response.write Server.URLEncode("Test.ASP?TestNum=100&TestStr=你好")

结果输出: Test%2EASP%3FTestNum%3D100%26TestStr%3D%C4%E3%BA%C3

在ASP中,有ASC(String),Hex(Number),Mid(String,start,[,length])这三个可能用

到的函数，如果是三个函数的用法

如:

ASC("A")=65,ASC("你")= -15133

Hex(65)="41",Hex(-15133)="C4E3"

Mid("hello",2,1)="e", mid("this is test!",9,2)="te"

现在要求编写编码函数Function TestEncode(SourceString),及一个解码函数

Function TestDecode(CodeString)。TestEncode(SourceString)是将SourceString

串中非字母且非汉字且非数字的字符转换为对应Ansi编码的十六进制编码！

如:

TestEncode("Test.ASP?TestNum=100&TestStr=你好")=

"Test%2EASP%3FTestNum%3D100%26TestStr%3D你好"

而TestDecode(CodeString)是将编码的串还原，是TestEncode的逆函数。

第五题：

编写一个星期的函数GetWeek(aDate)

返回"星期一、星期二、星期三..."

第六题：

用ASP输出九九乘法口决表的网页

输出如下：

1\*1=1

1\*2=2 2\*2=4

1\*3=3 2\*3=6 3\*3=9

...

要求编写一个完整的ASP文件

第七题到第九题

已知SQL Server数据库的有一个数据库TestDB，学生表结构如下：

表名:Student

字段名 类型 说明

id int 自增1

name varchar(16)

sex char(1) 'F'表示女性,'M'表示男性

... ...

已知已经定义了ADODB.Connection对象ConnTestDB已连接了上述的TestDB数据库

可以在以后的测试题中直接引用该对象.

第七题：

编写ASP代码,将Student中的人的姓名及性别列出来,并给统计学生人数如下:

姓名 性别

张三 男

李四 男

王五 女

... ...

总共有100个学生

第八题:

在上述数据库中,有一个表存放学生的得分的，结构如下：

表名：Score

字段名 类型 说明

StuID int 学生的ID值，关系是：Score.StuID=Student.ID

Chinese int

math int

要求输出内容：

姓名 语文 数学 总成绩

张三 60 100 160

...

请编写实现上述功的ASP代码

第九题：

已知：

某一学生：陈六，男，语文80分，数学60分，现要求编写ASP代码

将该学的数据插入数据库中，分别插入到上述的两个表Student,Score表中。

1. 关联、聚合(Aggregation)以及组合(Composition)的区别？

涉及到UML中的一些概念：关联是表示两个类的一般性联系，比如“学生”和“老师”就是一种关联关系；聚合表示has-a的关系，是一种相对松散的关系，聚合类不需要对被聚合类负责，如下图所示，用空的菱形表示聚合关系：

从实现的角度讲，聚合可以表示为:

class A {...} class B { A\* a; .....}

而组合表示contains-a的关系，关联性强于聚合：组合类与被组合类有相同的生命周期，组合类要对被组合类负责，采用实心的菱形表示组合关系：

实现的形式是:

class A{...} class B{ A a; ...}

2.面向对象的三个基本特征，并简单叙述之？

1）. 封装：将客观事物抽象成类，每个类对自身的数据和方法实行protection(private, protected,public)

2）. 继承：广义的继承有三种实现形式：实现继承（指使用基类的属性和方法而无需额外编码的能力）、可视继承（子窗体使用父窗体的外观和实现代码）、接口继承（仅使用属性和方法，实现滞后到子类实现）。前两种（类继承）和后一种（对象组合=>接口继承以及纯虚函数）构成了功能复用的两种方式。

3）. 多态：是将父对象设置成为和一个或更多的他的子对象相等的技术，赋值之后，父对象就可以根据当前赋值给它的子对象的特性以不同的方式运作。简单的说，就是一句话：允许将子类类型的指针赋值给父类类型的指针。

3. 重载（overload)和重写(overried，有的书也叫做“覆盖”）的区别？

常考的题目。从定义上来说：

重载：是指允许存在多个同名函数，而这些函数的参数表不同（或许参数个数不同，或许参数类型不同，或许两者都不同）。

重写：是指子类重新定义复类虚函数的方法。

从实现原理上来说：

重载：编译器根据函数不同的参数表，对同名函数的名称做修饰，然后这些同名函数就成了不同的函数（至少对于编译器来说是这样的）。如，有两个同名函数：function func(p:integer):integer;和function func(p:string):integer;。那么编译器做过修饰后的函数名称可能是这样的：int\_func、str\_func。对于这两个函数的调用，在编译器间就已经确定了，是静态的。也就是说，它们的地址在编译期就绑定了（早绑定），因此，重载和多态无关！

重写：和多态真正相关。当子类重新定义了父类的虚函数后，父类指针根据赋给它的不同的子类指针，动态的调用属于子类的该函数，这样的函数调用在编译期间是无法确定的（调用的子类的虚函数的地址无法给出）。因此，这样的函数地址是在运行期绑定的（晚绑定）。

4. 多态的作用？

主要是两个：1. 隐藏实现细节，使得代码能够模块化；扩展代码模块，实现代码重用；2. 接口重用：为了类在继承和派生的时候，保证使用家族中任一类的实例的某一属性时的正确调用。

5. Ado与Ado.net的相同与不同？

除了“能够让应用程序处理存储于DBMS 中的数据“这一基本相似点外，两者没有太多共同之处。但是Ado使用OLE DB 接口并基于微软的COM 技术，而ADO.NET 拥有自己的ADO.NET 接口并且基于微软的.NET 体系架构。众所周知.NET 体系不同于COM 体系，ADO.NET 接口也就完全不同于ADO和OLE DB 接口，这也就是说ADO.NET 和ADO是两种数据访问方式。ADO.net 提供对XML 的支持。

6. New delete 与malloc free 的联系与区别?

答案：都是在堆(heap)上进行动态的内存操作。用malloc函数需要指定内存分配的字节数并且不能初始化对象，new 会自动调用对象的构造函数。delete 会调用对象的destructor，而free 不会调用对象的destructor.

7. #define DOUBLE(x) x+x ，i = 5\*DOUBLE(5)； i 是多少？

答案：i 为30。

8. 有哪几种情况只能用intialization list 而不能用assignment?

答案：当类中含有const、reference 成员变量；基类的构造函数都需要初始化表。

9. C++是不是类型安全的？

答案：不是。两个不同类型的指针之间可以强制转换（用reinterpret cast)。C#是类型安全的。

10. main 函数执行以前，还会执行什么代码？

答案：全局对象的构造函数会在main 函数之前执行。

1. 分析一下这段程序的输出 (Autodesk)

class B

{

public:

B()

{

cout<<"default constructor"<}

~B()

{

cout<<"destructed"<}

B(int i):data(i) //B(int) works as a converter ( int -> instance of B)

{

cout<<"constructed by parameter " << data <}

private:

int data;

};

B Play( B b)

{

return b ;

}

(1) results:

int main(int argc, char\* argv[]) constructed by parameter 5

{ destructed B(5)形参析构

B t1 = Play(5); B t2 = Play(t1); 　 destructed t1形参析构

return 0;　　　　　　　　　　　　　　 destructed t2　注意顺序！

} destructed t1

(2) results:

int main(int argc, char\* argv[]) constructed by parameter 5

{ destructed B(5)形参析构

B t1 = Play(5); B t2 = Play(10); 　 constructed by parameter 10

return 0;　　　　　　　　　　　　　　 destructed B(10)形参析构

} destructed t2　注意顺序！

destructed t1

2. 写一个函数找出一个整数数组中，第二大的数 （microsoft）

答案：

const int MINNUMBER = -32767 ;

int find\_sec\_max( int data[] , int count)

{

int maxnumber = data[0] ;

int sec\_max = MINNUMBER ;

for ( int i = 1 ; i < count ; i++)

{

if ( data[i] > maxnumber )

{

sec\_max = maxnumber ;

maxnumber = data[i] ;

}

else

{

if ( data[i] > sec\_max )

sec\_max = data[i] ;

}

}

return sec\_max ;

}

3. 写一个在一个字符串(n)中寻找一个子串(m)第一个位置的函数。

KMP算法效率最好，时间复杂度是Ｏ(n+m),详见：http://www.zhanglihai.com/blog/c\_335\_kmp.html

4. 多重继承的内存分配问题：

比如有class A : public class B, public class C {}

那么A的内存结构大致是怎么样的？

5. 如何判断一个单链表是有环的？（注意不能用标志位，最多只能用两个额外指针）

struct node { char val; node\* next;}

bool check(const node\* head) {} //return false : 无环；true: 有环

一种O（n）的办法就是（搞两个指针，一个每次递增一步，一个每次递增两步，如果有环的话两者必然重合，反之亦然）：

bool check(const node\* head)

{

if(head==NULL) return false;

node \*low=head, \*fast=head->next;

while(fast!=NULL && fast->next!=NULL)

{

low=low->next;

fast=fast->next->next;

if(low==fast) return true;

}

return false;

}

1.简述堆和栈的区别。

2.简述虚函数的本质和实现机制。

3.列举C++中向函数传递参数的各种方式，并比较它们之间的区别和优缺点。

4.static关键字有几种用途？请简述每种用途。

5.简述引用和指针的异同。

6.简述函数重载（overloading）和覆盖（overriding）的区别。

7.什么是深拷贝？什么是浅拷贝？

8.请画出二维数组char a[4][3]的内存结构图，并指出a[3][2]和a[2][3]在内存中的位置。

9.简述包含和私有继承的异同。

10.用模板实现一个通用的栈，要求实现empty, size, push, pop, top五个成员函数，并用异常机制来报告错误（比如在一个大小为0的栈上执行pop操作）。

标准库

11.用string类编写一个程序，它从标准输入接收一行输入，然后统计出其中单词的个数，单词的分隔符只有逗号、点号和空格三种。

12.简述STL的容器container、迭代器iterator、算法algorithm和函数器functor的作用以及它们之间的关系，举例说明。

13.下面的代码存在什么问题，如何修正？

list list1;

for (int i = 0; i < 8; i ++) {

list1.push\_back(i);

}

for (list::iterator it = list1.begin(); it != list1.end(); ++it) {

if (\*it % 2 == 0) {

list1.erase(it);

}

}

Win32

14.比较ANSI和Unicode的区别，并说明C Runtime Library和Win32 API如何支持二者。

15.简述线程同步的四种机制。

16.简述虚拟内存机制，并描述在win95和winnt下内存空间的结构。

17.简述DLL的实现机制以及输出DLL中函数的两种方法。

网络

18.简述TCP和UDP之间的区别。

19.简述Winsock的六种主要IO处理模式。

20.描述一个你以前实现过的服务器的结构图，并分析其中可能存在的瓶颈。

1 dll,lib处理实现 .lib文件中包含什么? dll在内存中被多个函数共享,有几个拷贝?

2 template,template等编译的时候作为几个类

3 struct ,class的区别,联系

4 MAcro,inline区别,实现.编译运行时

5 进程与线程区别

6 hdc-cdc区别联系

7 model dialog,modeless dialog,从创建到销毁全过程具体描述

8 各个线程new的内存,别的线程是否可以使用？

9 RUNTIME\_CLASS, 是什么？

一.填空题

1.c#中的三元运算符是\_\_?:\_\_\_?

2.当整数a赋值给一个object对象时，整数a将会被\_\_装箱(封装)\_\_\_?

3.类成员有\_\_\_\_\_种可访问形式？ this.;new Class().Method;

4.public static const int A=1;这段代码有错误么？是什么？ const不能用static修饰

5.float f=-123.567F;

int i=(int)f;

i的值现在是\_\_\_\_\_? 123

6.利用operator声明且仅声明了“==”，有什么错误么?

7.委托声明的关键字是\_\_\_\_\_\_? delagete

8.用sealed修饰的类有什么特点？密封，不能继承

9.在Asp.net中所有的自定义用户控件都必须继承自\_\_\_\_\_\_\_\_?Control

10.在.Net中所有可序列化的类都被标记为\_\_\_\_\_?

11.在.Net托管代码中我们不用担心内存漏洞，这是因为有了\_\_\_\_\_\_?gc

12.下面的代码中有什么错误吗？\_\_\_\_\_\_\_

using System;

class A

{

public virtual void F(){

Console.WriteLine("A.F");

}

}

abstract class B:A

{

public abstract override void F(); // new public abstract void F();

}

13.当类T只声明了私有实例构造函数时，则在T的程序文本外部，\_\_\_可以\_\_\_（可以 or 不可以）从T派生出新的类，不可以\_\_\_\_（可以 or 不可以）直接创建T的任何实例。

14.下面这段代码有错误么？

switch (i){

case():

CaseZero();

break;

case 1:

CaseOne();

break;

case 2:

dufault; //wrong

CaseTwo();

break;

}

15.在.Net中，类System.Web.UI.Page 可以被继承么？可以

二.简答题

1.在c#中using和new这两个关键字有什么意义，请写出你所知道的意义？using 指令 和语句 new 创建实例 new 隐藏基类中方法

4.谈谈类和结构的区别？类是引用类型、结构是值类型

5.一个长度为10000的字符串，通过随机从a-z中抽取10000个字符组成。请用c＃语言编写主要程序来实现。

6.对于这样的一个枚举类型：

      enum Color:byte{

          Red,

          Green,

          Blue,

          Orange

       }

     string[] ss=Enum.GetNames(typeof(Color));

     byte[]   bb=Enum.GetValues(typeof(Color));

试写一段程序显示出枚举类型中定义的所有符号名称以及它们对应的数值。

7.您了解设计模式么？请列出您所知道的设计模式的名称。 //

8.请在SQL Server中设计表来保存一个树状结构的组织结构图（假设结构图中只有名称这一项内容需要保存），如果我想查询某一职位下的所有职位，用一个存储过程来实现，你有什么思路？

9.什么叫做SQL注入，如何防止？请举例说明。

10.下面这段代码输出什么？为什么？

     int i=5;

     int j=5;

     if (Object.ReferenceEquals(i,j))

          Console.WriteLine("Equal");

     else

          Console.WriteLine("Not Equal");

//不相等，因为比较的是对象

一、

1、写出以下函数的功能简述

Len Trim

Now Replace

2、写出ASP中常用的5大对象并作简单的解释

3、Response对象的\_\_\_\_\_\_\_\_\_\_\_\_\_方法可以将虚拟路径转化为物理路径。

4、<%Session.abandon%>意思为:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

5、 在ASP中，VBScript有多种控制程序流程语句，如If…Then, Select… Case,

For … Next, Do … Loop, Exit等语句。请为这五个语句分别写一段使用的代码。

二、程序分析

6、用ASP输出九九乘法口决表的网页

输出如下：

1\*1=1

1\*2=2 2\*2=4

1\*3=3 2\*3=6 3\*3=9

请写出完整的ASP代码程序。

7、编写一个通用的错误信息提示模块或者数据操作模块（连接数据库和关闭数据库），并注明其调用方法及相关参数说明。

[数据库操作模块相关提示：server.createobject("ADODB.Connection")

server.createobject("ADODB.Recordset")

Driver={SQL Server}; server=(local) database=fjw uid=sa pwd=sa

以上仅以MSSQL为例提示]

三、数据库相关

News表 (新闻内容表)

Id ClassID AddName title datatime

1 3 admin 湖南有了新的变化 2006-9-15

2 4 李先 长沙也有了新的变化 2005-6-9

…………

.

Newsclass表 (新闻栏目表)

ClassID ClassName

2 国外要闻

3 国内要闻

4 本地要闻

8、写一个SQL语句查询显示出新闻的栏目与标题及添加者按时间排序

9、设计一个简单的存储过程从上两表中查询，返回ID、添加者、栏目名、新闻标题、添加时间。该存储过程不使用任何参数。

SQL

1．现有以下数据表，请写出查询、插入、更新、删除数据表Hotel或数据表Room中记录的SQL语句。

2．现有需求：一个用户需要订购一些商品。

题目：上列需求中需要多少数据表能够完成基本设计？并简单画出设计。提示：表与表的关系。

Asp&ADO

1．写出您最常用的ASP内置对象（个数不限）和最常用的ADO对象。提示：接受参数等。

2．<%=rs.fields.item(“Name”)%> 是什么意思？

面对对象&B/S

1．面向对象的思想包括？在ASP中如何简单实现面对对象？

2．如何看待分层？ASP如何实现简单分层？

上机操作（任选）

1．利用DreamWeaver建立简单的动态站点。实现功能自定。主要是对数据库的操作。

2．利用作图软件（FireWorks）或CSS制作圆角表格（表格四角为圆润）。

**华为摸底考试试卷**

班 级：\_\_\_\_\_\_\_\_\_\_ 学号：\_\_\_\_\_\_\_\_\_\_\_ 姓 名：\_\_\_\_\_\_\_\_\_\_ 分数：

**说明：1-12题所有学员必须做，13-15题只用Java班学员做。**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 题目 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 总分 |
| 分数 |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |

1．请你分别画出OSI的七层网络结构图和TCP/IP的五层结构图。（6分）  
2．请你详细地解释一下IP协议的定义，在哪个层上面？主要有什么作用？TCP与UDP呢？（6分）  
3．请问交换机和路由器各自的实现原理是什么？分别在哪个层次上面实现的？（6分）  
4．请问C++的类和C里面的struct有什么区别？（6分）  
5．请讲一讲析构函数和虚函数的用法和作用。（6分）  
6．全局变量和局部变量有什么区别？是怎么实现的？操作系统和编译器是怎么知道的？（6分）  
7．8086是多少位的系统？在数据总线上是怎么实现的？（6分）

8．写出调用new Child()程序运行的结果。（6分）  
 >    public class Base  
 >    {  
 >        private int a=init("init a");  
 >        private static int b=init("init b");  
 >        public Base()  
 >        {  
 >           System.out.println("init Base");  
 >         }  
 >        public void init(String str)  
 >        {  
 >           System.out.println(str);  
 >         }  
 >     }  
 >     public class Child extends Base  
 >     {  
 >        private int x=init("init x");  
 >        private static int y=init("init y");  
 >        public Child()  
 >        {  
 >           System.out.println("init child");  
 >         }  
 >     }  
 >

 >    9．看程序后回答问题（6分）  
 >      public class A  
 >      {  
 >          public void methodOne(){}  
 >          public void methosTwo(){}  
 >          public void methosThree(){}  
 >          public void methosFour(){}  
 >       }  
 >       public class B extends A  
 >       {  
 >          public static void methodOne(){}  
 >          public void method Two(){}  
 >          public void method three(){}  
 >          public static void methosFour(){}  
 >        }  
 >      问：哪些方法覆盖了父类的方法，哪些方法隐藏了父类的方法  
  
 >  
  
 >  10．写一个类A，包含一个长度为10整型数组，一个方法可以取第一个数据，一个方法可以将数据加到数组尾。写一个消费 者类，能够从a中取数据，一个生产者类，能够将数据加到a 中，且是递增的整数，能同时供多个生产者和多个消费者对a 操作。（8分） **[Java班学员用Java语言做，.Net班学员用C++语言做]**

11．写一个类A，它包含一个长度为10的整型数组，一个函数，可以插入数据到数组尾，另一个函数，可以取得数组的第一个数据。写一个类，调用A的方法插入数据。写一个类，调用A的方法得到数据。要求使用多线程；

（8分） **[Java班学员用Java语言做，.Net班学员用C++语言做]**

12．有一个二进制文件Ref.sor，已知长整型为4个字节，短整型为2个字节，写一段程序从文件中读取一个字符串，然后再读一个长整型，最后读取一个短整型；（12分）

13．ArrayList与Vectors的区别?（6分）

14．Swing有哪些特点？（6分）

15．int，string，\*point，union哪些不是Java的数据类型？（6分）

***附加题：***

1、求出n！

2、你如何对一个电梯进行测试？（站在客户的角度考虑）

**技术类问题集（想的起来的都在这里，最少有90%）**

1。谈谈抽象类和借口的区别。

2。说说二叉树。（华为的人很喜欢问数据结构的问题，这是我们的同感）

3。C和C++中如何申请和释放内存空间。

4。JAVA中为什么没有多重继承。

5。当场写个程序：写一个方法，实现如下功能，传进一个字符串，如“agery”，再传进一个字符’e‘，要求去掉字符串中和给定字符一样的所有字符，并返回，如返回’agry‘。（题目不难，可是当者面试官做心理很紧张）

6。华为很喜欢考线程问题，前两年都考了，这次也有考。题目不清楚了，就是实现两个线程一个从队列头取数据，一个往队列末尾加数据之类的东东，有些难度，对我来说，因为我是自学的，呵呵。

7。short s1 = s1+1;这个语句能正常运行吗？

**综合类问题集（想的起来的都在这里，最少有90%）**

1。谈谈你自己；

2。你有没有过很大压力的时候，为什么。

3。你失败过吗？为什么失败？解决了吗？你得到什么启发？

4。你为什么要应聘我们公司？你了解华为的企业文化吗，说说看？什么是狼性文化？

5。如果你这次应聘失败了，你想可能是什么原因，为什么？

6。你有什么弱点？

7。你在原来的公司做什么工作？为什么要离开？

8。如果调你去外地工作，你愿意服从单位分配吗

大家在回答这些问题时最好不要编造，面试官会就你的回答继续问下去，一环套一环。如果是编造的，后面很难把话说圆。

1. 请用C#语言写出与SQLSERVER访问时的具体过程。在涉及到SQLSERVER数据库编程的时候要注意哪些方面？
2. 请尽量列举出SQLSERVER数据库备份和还原的方法。
3. 请用C#语言写出在本地创建一个UDP接收端口的具体过程。
4. 请解释：什么叫异常？请谈谈C#中如何捕捉异常，处理异常。
5. 请解释：什么是线程？请描述线程时的具体步骤。在线程编程时要注意哪些方面？
6. 有一个BYTE[20]的数组，请将这种BYTE的第8个字节到第11个字节这四个字节提取出来，按照有低到高的顺序组成一个4个字节的浮点数，请具体写出c#程序。
7. 对任一张表的数据进行增，删，改，查要求，运用Webservice，体现出三层架构

1、进程是一个 概念，程序是一个 概念，进程有哪几个步骤？

2、C#语言的突出特点是什么？

3、表A（a.b.c）,表B(a.b.c),a为主链，将表A的数据更新到表B中

4、用JAVAScript\C#写编码，将数字转换为字符。（如：1-A）

5、用JAVAScript用正则表达式检测字符串是否是数字串。

6、∑=根号（X平方-Y平方）的绝对值+X 编程

7、用SQL将数据库中指定的一个表传入TABLE数据集中。

8、写一个游标实现对数据的删除

9、写一个触发器实现对数据的删除

10、面向对象的特点

11、析构函数的意义及作用

12、C#中是否可以省略

13、谈谈你的项目经验

14、谈谈你的优缺点。

请在SQL中建立表A（id,name,address,tel,sex）\表B(id,sellessonl,sellession2,sellession3)和表C(lessionid,lessionnane)A表和B表中的id为学号，sellession1、sellession2.sellession3分别是学生选修的3门课程的id。C表是课程基础信息表。

要求字段类型自己定义，尽量节省数据库存储空间。

用ASP 。NET和C#编写一个程序完成以下功能：

1. 以表格和方式显示出每个学生所选择的课程并可以通过学生姓名进行查询。
2. 以表格的方式显示出每门课程所选择的人数。
3. 以二维图表显示每门课程所选择的人数。
4. 课程基础信息表的录入和查询。

要求：表格数据的显示要用DataGrid显示要用代码绑定数据，并用代码限定每列的宽度和高度。界面要美观，简洁和清楚。

1. 找错  
   void test1()   
   {   
   char string[10]; //string的长度应该设为11，要给''留出位置  
   char\* str1="0123456789";   
   strcpy( string, str1);   
   }   
     
   void test1()   
   {   
   char string[10], str1[10];   
   for(I=0; I<10； I++ ) //变量I没定义,增加int I;   
   //I不能取到9，同上一题一样要给''留出位置  
   //应该改成for(I=0; I<9; I++)   
   {   
   str1[i] = 'a'; //变量i没定义，改为str1[I] = 'a';   
   }   
   //此处要加上结尾标识 str1[i]='';   
   strcpy(string, str1);   
   }  
     
   void test3(char\* str1)   
   {   
   char string[10];   
   if( strlen(str1) <= 10 ) //同样的问题，str1的长度不能超过9；改为   
   // if( strlen(str1) < 10 )   
   {   
   strcpy(string, str1);   
   }   
   }   
     
     
   2。   
   #define MAX\_SRM 256   
     
   DSN get\_SRM\_no()  
   {   
   static int SRM\_no; //静态变量没有初始化   
   int I;   
     
   for(I=0; I { //此处不知所云  
   SRM\_no %= MAX\_SRM;   
   if( MY\_SRM.state == IDLE ) //MY\_SRM IDLE未定义  
   {   
   break;  
   }  
   }  
   if( I>=MAX\_SRM)  
   return NULL\_SRM; //NULL\_SRM未定义  
   else  
   return SRM\_no;   
   }   
     
   3。  
   8,10,12,14,16  
     
   4.  
   题目有错吧? 是不是应该改成这样:   
   int func( int a )   
   {   
   int b;   
   switch( a )   
   {   
   case 1: b=30;   
   case 2: b=20;   
   case 3: b=16;  
   default: b=0;   
   }   
   return b;   
   }  
     
   如果是这样func(1) = 0;   
     
     
   5.   
   　　 2   
     
   6.   
   sizeof(int) \*12 = 4\*12 = 64 (byte)

4. static有什么用途？（请至少说明两种）

1.限制变量的作用域

2.设置变量的存储域

7. 引用与指针有什么区别？

1) 引用必须被初始化，指针不必。

2) 引用初始化以后不能被改变，指针可以改变所指的对象。

2) 不存在指向空值的引用，但是存在指向空值的指针。

8. 描述实时系统的基本特性

在特定时间内完成特定的任务，实时性与可靠性

9. 全局变量和局部变量在内存中是否有区别？如果有，是什么区别？

全局变量储存在静态数据库，局部变量在堆栈

10. 什么是平衡二叉树？

左右子树都是平衡二叉树 且左右子树的深度差值的绝对值不大于1

11. 堆栈溢出一般是由什么原因导致的？

没有回收垃圾资源

12. 什么函数不能声明为虚函数？

constructor

13. 冒泡排序算法的时间复杂度是什么？

O(n^2)

14. 写出float x 与“零值”比较的if语句。

if(x>0.000001&&x<-0.000001)

16. Internet采用哪种网络协议？该协议的主要层次结构？

tcp/ip 应用层/传输层/网络层/数据链路层/物理层

17. Internet物理地址和IP地址转换采用什么协议？

ARP (Address Resolution Protocol)（地址解析協議）

18.IP地址的编码分为哪俩部分？

IP地址由两部分组成，网络号和主机号。不过是要和“子网掩码”按位与上之后才能区分哪些是网络位哪些是主机位。

2.用户输入M,N值，从1至N开始顺序循环数数，每数到M输出该数值，直至全部输出。写出C程序。

循环链表，用取余操作做

3.不能做switch()的参数类型是：

switch的参数不能为实型。

華為

1、局部变量能否和全局变量重名？

答：能，局部会屏蔽全局。要用全局变量，需要使用"::"

局部变量可以与全局变量同名，在函数内引用这个变量时，会用到同名的局部变量，而不会用到全局变量。对于有些编译器而言，在同一个函数内可以定义多个同名的局部变量，比如在两个循环体内都定义一个同名的局部变量，而那个局部变量的作用域就在那个循环体内

2、如何引用一个已经定义过的全局变量？

答：extern

可以用引用头文件的方式，也可以用extern关键字，如果用引用头文件方式来引用某个在头文件中声明的全局变理，假定你将那个变写错了，那么在编译期间会报错，如果你用extern方式引用时，假定你犯了同样的错误，那么在编译期间不会报错，而在连接期间报错

3、全局变量可不可以定义在可被多个.C文件包含的头文件中？为什么？

答：可以，在不同的C文件中以static形式来声明同名全局变量。

可以在不同的C文件中声明同名的全局变量，前提是其中只能有一个C文件中对此变量赋初值，此时连接不会出错

4、语句for( ；1 ；)有什么问题？它是什么意思？

答：和while(1)相同。

5、do……while和while……do有什么区别？

答：前一个循环一遍再判断，后一个判断以后再循环

6、请写出下列代码的输出内容

#include<stdio.h>

main()

{

int a,b,c,d;

a=10;

b=a++;

c=++a;

d=10\*a++;

printf("b，c，d：%d，%d，%d"，b，c，d）;

return 0;

}

答：10，12，120

1、static全局变量与普通的全局变量有什么区别？static局部变量和普通局部变量有什么区别？static函数与普通函数有什么区别？

全局变量(外部变量)的说明之前再冠以static 就构成了静态的全局变量。全局变量本身就是静态存储方式，

静态全局变量当然也是静态存储方式。这两者在存储方式上并无不同。这两者的区别虽在于非静态全局变量的作用域是整个源程序，当一个源程序由多个源文件组成时，非静态的全局变量在各个源文件中都是有效的。

而静态全局变量则限制了其作用域，

即只在定义该变量的源文件内有效，在同一源程序的其它源文件中不能使用它。由于静态全局变量的作用域局限于一个源文件内，只能为该源文件内的函数公用，因此可以避免在其它源文件中引起错误。

从以上分析可以看出， 把局部变量改变为静态变量后是改变了它的存储方式即改变了它的生存期。把全局变量改变为静态变量后是改变了它的作用域， 限制了它的使用范围。

static函数与普通函数作用域不同。仅在本文件。只在当前源文件中使用的函数应该说明为内部函数(static)，内部函数应该在当前源文件中说明和定义。对于可在当前源文件以外使用的函数，应该在一个头文件中说明，要使用这些函数的源文件要包含这个头文件

static全局变量与普通的全局变量有什么区别：static全局变量只初使化一次，防止在其他文件单元中被引用;

static局部变量和普通局部变量有什么区别：static局部变量只被初始化一次，下一次依据上一次结果值；

static函数与普通函数有什么区别：static函数在内存中只有一份，普通函数在每个被调用中维持一份拷贝

2、程序的局部变量存在于（堆栈）中，全局变量存在于（静态区 ）中，动态申请数据存在于（ 堆）中。

3、设有以下说明和定义：

typedef union {long i; int k[5]; char c;} DATE;

struct data { int cat; DATE cow; double dog;} too;

DATE max;

则语句 printf("%d",sizeof(struct date)+sizeof(max));的执行结果是：\_\_\_52\_\_\_\_

答：DATE是一个union, 变量公用空间. 里面最大的变量类型是int[5], 占用20个字节. 所以它的大小是20

data是一个struct, 每个变量分开占用空间. 依次为int4 + DATE20 + double8 = 32.

所以结果是 20 + 32 = 52.

当然...在某些16位编辑器下, int可能是2字节,那么结果是 int2 + DATE10 + double8 = 20

4、队列和栈有什么区别？

队列先进先出，栈后进先出

5、写出下列代码的输出内容

#include<stdio.h>

int inc(int a)

{

return(++a);

}

int multi(int\*a,int\*b,int\*c)

{

return(\*c=\*a\*\*b);

}

typedef int(FUNC1)(int in);

typedef int(FUNC2) (int\*,int\*,int\*);

void show(FUNC2 fun,int arg1, int\*arg2)

{

INCp=&inc;

int temp =p(arg1);

fun(&temp,&arg1, arg2);

printf("%d\n",\*arg2);

}

main()

{

int a;

show(multi,10,&a);

return 0;

}

答：110

7、请找出下面代码中的所以错误

说明：以下代码是把一个字符串倒序，如“abcd”倒序后变为“dcba”

1、#include"string.h"

2、main()

3、{

4、 char\*src="hello,world";

5、 char\* dest=NULL;

6、 int len=strlen(src);

7、 dest=(char\*)malloc(len);

8、 char\* d=dest;

9、 char\* s=src[len];

10、 while(len--!=0)

11、 d++=s--;

12、 printf("%s",dest);

13、 return 0;

14、}

答：

方法1：

int main(){

char\* src = "hello,world";

int len = strlen(src);

char\* dest = (char\*)malloc(len+1);//要为\0分配一个空间

char\* d = dest;

char\* s = &src[len-1];//指向最后一个字符

while( len-- != 0 )

\*d++=\*s--;

\*d = 0;//尾部要加\0

printf("%s\n",dest);

free(dest);// 使用完，应当释放空间，以免造成内存汇泄露

return 0;

}

方法2：

#include <stdio.h>

#include <string.h>

main()

{

char str[]="hello,world";

int len=strlen(str);

char t;

for(int i=0; i<len/2; i++)

{

t=str[i];

str[i]=str[len-i-1]; str[len-i-1]=t;

}

printf("%s",str);

return 0;

}

1.-1,2,7,28,,126请问28和126中间那个数是什么？为什么？

第一题的答案应该是4^3-1=63

规律是n^3-1(当n为偶数0，2，4)

n^3+1(当n为奇数1，3，5)

答案：63

2.用两个栈实现一个队列的功能？要求给出算法和思路！

设2个栈为A,B, 一开始均为空.

入队:

将新元素push入栈A;

出队:

(1)判断栈B是否为空；

(2)如果不为空，则将栈A中所有元素依次pop出并push到栈B；

(3)将栈B的栈顶元素pop出；

这样实现的队列入队和出队的平摊复杂度都还是O(1),

比上面的几种方法要好。3.在c语言库函数中将一个字符转换成整型的函数是atool()吗，这个函数的原型是什么？

函数名: atol

功 能: 把字符串转换成长整型数

用 法: long atol(const char \*nptr);

程序例:

#include <stdlib.h>

#include <stdio.h>

int main(void)

{

long l;

char \*str = "98765432";

l = atol(lstr);

printf("string = %s integer = %ld\n", str, l);

return(0);

}

2.对于一个频繁使用的短小函数,在C语言中应用什么实现,在C++中应用什么实现?

c用宏定义，c++用inline

3.直接链接两个信令点的一组链路称作什么?

PPP点到点连接

4.接入网用的是什么接口?

5.voip都用了那些协议?

6.软件测试都有那些种类?

黑盒：针对系统功能的测试 白合：测试函数功能，各函数接口

7.确定模块的功能和模块的接口是在软件设计的那个队段完成的?

概要设计阶段

8.enum string

{

x1,

x2,

x3=10,

x4,

x5,

}x;

问x= 0x801005，0x8010f4 ;

9.unsigned char \*p1;

unsigned long \*p2;

p1=(unsigned char \*)0x801000;

p2=(unsigned long \*)0x810000;

请问p1+5= ;

p2+5= ;

三.选择题:

1.Ethternet链接到Internet用到以下那个协议?

A.HDLC;B.ARP;C.UDP;D.TCP;E.ID

2.属于网络层协议的是:

A.TCP;B.IP;C.ICMP;D.X.25

3.Windows消息调度机制是:

A.指令队列;B.指令堆栈;C.消息队列;D.消息堆栈;

4.unsigned short hash(unsigned short key)

{

return (key>>)%256

}

请问hash(16),hash(256)的值分别是:

A.1.16;B.8.32;C.4.16;D.1.32

四.找错题:

1.请问下面程序有什么错误?

int a[60][250][1000],i,j,k;

for(k=0;k<=1000;k++)

for(j=0;j<250;j++)

for(i=0;i<60;i++)

a[i][j][k]=0;

把循环语句内外换一下

2.#define Max\_CB 500

void LmiQueryCSmd(Struct MSgCB \* pmsg)

{

unsigned char ucCmdNum;

......

for(ucCmdNum=0;ucCmdNum<Max\_CB;ucCmdNum++)

{

......;

}

死循环

3.以下是求一个数的平方的程序,请找出错误:

#define SQUARE(a)((a)\*(a))

int a=5;

int b;

b=SQUARE(a++);

4.typedef unsigned char BYTE

int examply\_fun(BYTE gt\_len; BYTE \*gt\_code)

{

BYTE \*gt\_buf;

gt\_buf=(BYTE \*)MALLOC(Max\_GT\_Length);

......

if(gt\_len>Max\_GT\_Length)

{

return GT\_Length\_ERROR;

}

.......

}

五.问答题:

1.IP Phone的原理是什么?

IPV6

2.TCP/IP通信建立的过程怎样，端口有什么作用？

三次握手，确定是哪个应用程序使用该协议

3.1号信令和7号信令有什么区别，我国某前广泛使用的是那一种？

4.列举5种以上的电话新业务？

微软亚洲技术中心的面试题！！！

1．进程和线程的差别。

线程是指进程内的一个执行单元,也是进程内的可调度实体.

与进程的区别:

(1)调度：线程作为调度和分配的基本单位，进程作为拥有资源的基本单位

(2)并发性：不仅进程之间可以并发执行，同一个进程的多个线程之间也可并发执行

(3)拥有资源：进程是拥有资源的一个独立单位，线程不拥有系统资源，但可以访问隶属于进程的资源.

(4)系统开销：在创建或撤消进程时，由于系统都要为之分配和回收资源，导致系统的开销明显大于创建或撤消线程时的开销。

2.测试方法

人工测试：个人复查、抽查和会审

机器测试：黑盒测试和白盒测试

2．Heap与stack的差别。

Heap是堆，stack是栈。

Stack的空间由操作系统自动分配/释放，Heap上的空间手动分配/释放。

Stack空间有限，Heap是很大的自由存储区

C中的malloc函数分配的内存空间即在堆上,C++中对应的是new操作符。

程序在编译期对变量和函数分配内存都在栈上进行,且程序运行过程中函数调用时参数的传递也在栈上进行

3．Windows下的内存是如何管理的？

4．介绍.Net和.Net的安全性。

5．客户端如何访问.Net组件实现Web Service？

6．C/C++编译器中虚表是如何完成的？

7．谈谈COM的线程模型。然后讨论进程内/外组件的差别。

8．谈谈IA32下的分页机制

小页(4K)两级分页模式，大页(4M)一级

9．给两个变量，如何找出一个带环单链表中是什么地方出现环的？

一个递增一，一个递增二，他们指向同一个接点时就是环出现的地方

10．在IA32中一共有多少种办法从用户态跳到内核态？

通过调用门，从ring3到ring0，中断从ring3到ring0，进入vm86等等

11．如果只想让程序有一个实例运行，不能运行两个。像winamp一样，只能开一个窗口，怎样实现？

用内存映射或全局原子（互斥变量）、查找窗口句柄..

FindWindow，互斥，写标志到文件或注册表,共享内存。.

12．如何截取键盘的响应，让所有的‘a’变成‘b’？

键盘钩子SetWindowsHookEx

　13．Apartment在COM中有什么用？为什么要引入？

　14．存储过程是什么？有什么用？有什么优点？

我的理解就是一堆sql的集合，可以建立非常复杂的查询，编译运行，所以运行一次后，以后再运行速度比单独执行SQL快很多

　15．Template有什么特点？什么时候用？

16．谈谈Windows DNA结构的特点和优点。

网络编程中设计并发服务器，使用多进程 与 多线程 ，请问有什么区别？

1，进程：子进程是父进程的复制品。子进程获得父进程数据空间、堆和栈的复制品。

2，线程：相对与进程而言，线程是一个更加接近与执行体的概念，它可以与同进程的其他线程共享数据，但拥有自己的栈空间，拥有独立的执行序列。

两者都可以提高程序的并发度，提高程序运行效率和响应时间。

线程和进程在使用上各有优缺点：线程执行开销小，但不利于资源管理和保护；而进程正相反。同时，线程适合于在SMP机器上运行，而进程则可以跨机器迁移。

思科

1. 用宏定义写出swap（x，y）

#define swap(x, y)\

x = x + y;\

y = x - y;\

x = x - y;

2.数组a[N]，存放了1至N-1个数，其中某个数重复一次。写一个函数，找出被重复的数字.时间复杂度必须为o（N）函数原型：

int do\_dup(int a[],int N)

3 一语句实现x是否为2的若干次幂的判断

int i = 512;

cout << boolalpha << ((i & (i - 1)) ? false : true) << endl;

4.unsigned int intvert(unsigned int x,int p,int

n)实现对x的进行转换,p为起始转化位,n为需要转换的长度,假设起始点在右边.如x=0b0001 0001,p=4,n=3转换后x=0b0110 0001

unsigned int intvert(unsigned int x,int p,int n){

unsigned int \_t = 0;

unsigned int \_a = 1;

for(int i = 0; i < n; ++i){

\_t |= \_a;

\_a = \_a << 1;

}

\_t = \_t << p;

x ^= \_t;

return x;

}

慧通：

什么是预编译

何时需要预编译：

１、总是使用不经常改动的大型代码体。

２、程序由多个模块组成，所有模块都使用一组标准的包含文件和相同的编译选项。在这种情况下，可以将所有包含文件预编译为一个预编译头。

char \* const p;

char const \* p

const char \*p

上述三个有什么区别？

char \* const p; //常量指针，p的值不可以修改

char const \* p；//指向常量的指针，指向的常量值不可以改

const char \*p； //和char const \*p

char str1[] = "abc";

char str2[] = "abc";

const char str3[] = "abc";

const char str4[] = "abc";

const char \*str5 = "abc";

const char \*str6 = "abc";

char \*str7 = "abc";

char \*str8 = "abc";

cout << ( str1 == str2 ) << endl;

cout << ( str3 == str4 ) << endl;

cout << ( str5 == str6 ) << endl;

cout << ( str7 == str8 ) << endl;

结果是：0 0 1 1

解答：str1,str2,str3,str4是数组变量，它们有各自的内存空间；

而str5,str6,str7,str8是指针，它们指向相同的常量区域。

12. 以下代码中的两个sizeof用法有问题吗？[C易]

void UpperCase( char str[] ) // 将 str 中的小写字母转换成大写字母

{

for( size\_t i=0; i<sizeof(str)/sizeof(str[0]); ++i )

if( 'a'<=str[i] && str[i]<='z' )

str[i] -= ('a'-'A' );

}

char str[] = "aBcDe";

cout << "str字符长度为: " << sizeof(str)/sizeof(str[0]) << endl;

UpperCase( str );

cout << str << endl;

答：函数内的sizeof有问题。根据语法，sizeof如用于数组，只能测出静态数组的大小，无法检测动态分配的或外部数组大小。函数外的str是一个静态定义的数组，因此其大小为6，函数内的str实际只是一个指向字符串的指针，没有任何额外的与数组相关的信息，因此sizeof作用于上只将其当指针看，一个指针为4个字节，因此返回4。

一个32位的机器,该机器的指针是多少位

指针是多少位只要看地址总线的位数就行了。80386以后的机子都是32的数据总线。所以指针的位数就是4个字节了。

main()

{

int a[5]={1,2,3,4,5};

int \*ptr=(int \*)(&a+1);

printf("%d,%d",\*(a+1),\*(ptr-1));

}

输出：2,5

\*(a+1）就是a[1]，\*(ptr-1)就是a[4],执行结果是2，5

&a+1不是首地址+1，系统会认为加一个a数组的偏移，是偏移了一个数组的大小（本例是5个int）

int \*ptr=(int \*)(&a+1);

则ptr实际是&(a[5]),也就是a+5

原因如下：

&a是数组指针，其类型为 int (\*)[5];

而指针加1要根据指针类型加上一定的值，

不同类型的指针+1之后增加的大小不同

a是长度为5的int数组指针，所以要加 5\*sizeof(int)

所以ptr实际是a[5]

但是prt与(&a+1)类型是不一样的(这点很重要)

所以prt-1只会减去sizeof(int\*)

a,&a的地址是一样的，但意思不一样，a是数组首地址，也就是a[0]的地址，&a是对象（数组）首地址，a+1是数组下一元素的地址，即a[1],&a+1是下一个对象的地址，即a[5].

1.请问以下代码有什么问题：

int main()

{

char a;

char \*str=&a;

strcpy(str,"hello");

printf(str);

return 0;

}

没有为str分配内存空间，将会发生异常

问题出在将一个字符串复制进一个字符变量指针所指地址。虽然可以正确输出结果，但因为越界进行内在读写而导致程序崩溃。

char\* s="AAA";

printf("%s",s);

s[0]='B';

printf("%s",s);

有什么错？

"AAA"是字符串常量。s是指针，指向这个字符串常量，所以声明s的时候就有问题。

cosnt char\* s="AAA";

然后又因为是常量，所以对是s[0]的赋值操作是不合法的。

1、写一个“标准”宏，这个宏输入两个参数并返回较小的一个。

.#define Min(X, Y) ((X)>(Y)?(Y):(X))//结尾没有;

2、嵌入式系统中经常要用到无限循环，你怎么用C编写死循环。

while(1){}或者for(;;)

3、关键字static的作用是什么？

定义静态变量

4、关键字const有什么含意？

表示常量不可以修改的变量。

5、关键字volatile有什么含意？并举出三个不同的例子？

提示编译器对象的值可能在编译器未监测到的情况下改变。

int (\*s[10])(int) 表示的是什么啊

int (\*s[10])(int) 函数指针数组，每个指针指向一个int func(int param)的函数。

1.有以下表达式：

int a=248; b=4;int const c=21;const int \*d=&a;

int \*const e=&b;int const \*f const =&a;

请问下列表达式哪些会被编译器禁止？为什么？

\*c=32;d=&b;\*d=43;e=34;e=&a;f=0x321f;

\*c 这是个什么东东，禁止

\*d 说了是const， 禁止

e = &a 说了是const 禁止

const \*f const =&a; 禁止

2.交换两个变量的值，不使用第三个变量。即a=3,b=5,交换之后a=5,b=3;

有两种解法, 一种用算术算法, 一种用^(异或)

a = a + b;

b = a - b;

a = a - b;

or

a = a^b;// 只能对int,char..

b = a^b;

a = a^b;

or

a ^= b ^= a;

3.c和c++中的struct有什么不同？

c和c++中struct的主要区别是c中的struct不可以含有成员函数，而c++中的struct可以。c++中struct和class的主要区别在于默认的存取权限不同，struct默认为public，而class默认为private

4.#include <stdio.h>

#include <stdlib.h>

void getmemory(char \*p)

{

p=(char \*) malloc(100);

strcpy(p,"hello world");

}

int main( )

{

char \*str=NULL;

getmemory(str);

printf("%s/n",str);

free(str);

return 0;

}

程序崩溃，getmemory中的malloc 不能返回动态内存， free（）对str操作很危险

5.char szstr[10];

strcpy(szstr,"0123456789");

产生什么结果？为什么？

长度不一样，会造成非法的OS

6.列举几种进程的同步机制，并比较其优缺点。

原子操作

信号量机制

自旋锁

管程，会合，分布式系统

7.进程之间通信的途径

共享存储系统

消息传递系统

管道：以文件系统为基础

11.进程死锁的原因

资源竞争及进程推进顺序非法

12.死锁的4个必要条件

互斥、请求保持、不可剥夺、环路

13.死锁的处理

鸵鸟策略、预防策略、避免策略、检测与解除死锁

15. 操作系统中进程调度策略有哪几种？

FCFS(先来先服务)，优先级，时间片轮转，多级反馈

8.类的静态成员和非静态成员有何区别？

类的静态成员每个类只有一个，非静态成员每个对象一个

9.纯虚函数如何定义？使用时应注意什么？

virtual void f()=0;

是接口，子类必须要实现

10.数组和链表的区别

数组：数据顺序存储，固定大小

连表：数据可以随机存储，大小可动态改变

12.ISO的七层模型是什么？tcp/udp是属于哪一层？tcp/udp有何优缺点？

应用层

表示层

会话层

运输层

网络层

物理链路层

物理层

tcp /udp属于运输层

TCP 服务提供了数据流传输、可靠性、有效流控制、全双工操作和多路复用技术等。

与 TCP 不同， UDP 并不提供对 IP 协议的可靠机制、流控制以及错误恢复功能等。由于 UDP 比较简单， UDP 头包含很少的字节，比 TCP

负载消耗少。

tcp: 提供稳定的传输服务，有流量控制，缺点是包头大，冗余性不好

udp: 不提供稳定的服务，包头小，开销小

1：(void \*)ptr 和 (\*(void\*\*))ptr的结果是否相同？其中ptr为同一个指针

.(void \*)ptr 和 (\*(void\*\*))ptr值是相同的

2：int main()

{

int x=3;

printf("%d",x);

return 1;

}

问函数既然不会被其它函数调用，为什么要返回1？

mian中，c标准认为0表示成功，非0表示错误。具体的值是某中具体出错信息

1，要对绝对地址0x100000赋值，我们可以用

(unsigned int\*)0x100000 = 1234;

那么要是想让程序跳转到绝对地址是0x100000去执行，应该怎么做？

\*((void (\*)( ))0x100000 ) ( );

首先要将0x100000强制转换成函数指针,即:

(void (\*)())0x100000

然后再调用它:

\*((void (\*)())0x100000)();

用typedef可以看得更直观些:

typedef void(\*)() voidFuncPtr;

\*((voidFuncPtr)0x100000)();

2，已知一个数组table，用一个宏定义，求出数据的元素个数

#define NTBL

#define NTBL (sizeof(table)/sizeof(table[0]))

面试题: 线程与进程的区别和联系? 线程是否具有相同的堆栈? dll是否有独立的堆栈?

进程是死的，只是一些资源的集合，真正的程序执行都是线程来完成的，程序启动的时候操作系统就帮你创建了一个主线程。

每个线程有自己的堆栈。

DLL 中有没有独立的堆栈，这个问题不好回答，或者说这个问题本身是否有问题。因为DLL中的代码是被某些线程所执行，只有线程拥有堆栈，如果DLL中的代码是

EXE中的线程所调用，那么这个时候是不是说这个DLL没有自己独立的堆栈？如果DLL中的代码是由DLL自己创建的线程所执行，那么是不是说DLL有独立的堆栈？

以上讲的是堆栈，如果对于堆来说，每个DLL有自己的堆，所以如果是从DLL中动态分配的内存，最好是从DLL中删除，如果你从DLL中分配内存，然后在EXE中，或者另外一个DLL中删除，很有可能导致程序崩溃

unsigned short A = 10;

printf("~A = %u\n", ~A);

char c=128;

printf("c=%d\n",c);

输出多少？并分析过程

第一题，～A ＝0xfffffff5,int值 为－11，但输出的是uint。所以输出4294967285

第二题，c＝0x10,输出的是int，最高位为1，是负数，所以它的值就是0x00的补码就是128，所以输出－128。

这两道题都是在考察二进制向int或uint转换时的最高位处理。

分析下面的程序：

void GetMemory(char \*\*p,int num)

{

\*p=(char \*)malloc(num);

}

int main()

{

char \*str=NULL;

GetMemory(&str,100);

strcpy(str,"hello");

free(str);

if(str!=NULL)

{

strcpy(str,"world");

}

printf("\n str is %s",str);

getchar();

}

问输出结果是什么？希望大家能说说原因，先谢谢了

输出str is world。

free 只是释放的str指向的内存空间,它本身的值还是存在的.

所以free之后，有一个好的习惯就是将str=NULL.

此时str指向空间的内存已被回收,如果输出语句之前还存在分配空间的操作的话,这段存储空间是可能被重新分配给其他变量的,

尽管这段程序确实是存在大大的问题（上面各位已经说得很清楚了），但是通常会打印出world来。

这是因为，进程中的内存管理一般不是由操作系统完成的，而是由库函数自己完成的。

当你malloc一块内存的时候，管理库向操作系统申请一块空间（可能会比你申请的大一些），然后在这块空间中记录一些管理信息（一般是在你申请的内存前面一点），并将可用内存的地址返回。但是释放内存的时候，管理库通常都不会将内存还给操作系统，因此你是可以继续访问这块地址的，只不过。。。。。。。。楼上都说过了，最好别这么干。

char a[10],strlen(a)为什么等于15？运行的结果

#include "stdio.h"

#include "string.h"

void main()

{

char aa[10];

printf("%d",strlen(aa));

}

sizeof()和初不初始化，没有关系；

strlen()和初始化有关。

char (\*str)[20];/\*str是一个数组指针，即指向数组的指针．\*/

char \*str[20];/\*str是一个指针数组，其元素为指针型数据．\*/

long a=0x801010;

a+5=?

0x801010用二进制表示为：“1000 0000 0001 0000 0001 0000”，十进制的值为8392720，再加上5就是8392725罗

1)给定结构struct A

{

char t:4;

char k:4;

unsigned short i:8;

unsigned long m;

};问sizeof(A) = ?

给定结构struct A

{

char t:4; 4位

char k:4; 4位

unsigned short i:8; 8位

unsigned long m; // 偏移2字节保证4字节对齐

}; // 共8字节

2)下面的函数实现在一个数上加一个数，有什么错误？请改正。

int add\_n ( int n )

{

static int i = 100;

i += n;

return i;

}

当你第二次调用时得不到正确的结果，难道你写个函数就是为了调用一次？问题就出在 static上？

// 帮忙分析一下

#include<iostream.h>

#include <string.h>

#include <malloc.h>

#include <stdio.h>

#include <stdlib.h>

#include <memory.h>

typedef struct AA

{

int b1:5;

int b2:2;

}AA;

void main()

{

AA aa;

char cc[100];

strcpy(cc,"0123456789abcdefghijklmnopqrstuvwxyz");

memcpy(&aa,cc,sizeof(AA));

cout << aa.b1 <<endl;

cout << aa.b2 <<endl;

}

答案是 -16和１

首先sizeof(AA)的大小为4,b1和b2分别占5bit和2bit.

经过strcpy和memcpy后,aa的4个字节所存放的值是:

0,1,2,3的ASC码，即00110000,00110001,00110010,00110011

所以，最后一步：显示的是这４个字节的前５位，和之后的２位

分别为：10000,和01

因为int是有正负之分　　所以：答案是-16和１

求函数返回值，输入x=9999;

int func （ x ）

{

int countx = 0;

while ( x )

{

countx ++;

x = x&(x-1);

}

return countx;

}

结果呢？

知道了这是统计9999的二进制数值中有多少个1的函数，且有

9999＝9×1024＋512＋256＋15

9×1024中含有1的个数为2；

512中含有1的个数为1；

256中含有1的个数为1；

15中含有1的个数为4；

故共有1的个数为8，结果为8。

1000 - 1 = 0111，正好是原数取反。这就是原理。

用这种方法来求1的个数是很效率很高的。

不必去一个一个地移位。循环次数最少。

int a,b,c 请写函数实现C=a+b ,不可以改变数据类型,如将c改为long int,关键是如何处理溢出问题

bool add (int a, int b,int \*c)

{

\*c=a+b;

return (a>0 && b>0 &&(\*c<a || \*c<b) || (a<0 && b<0 &&(\*c>a || \*c>b)));

}

分析：

struct bit

{ int a:3;

int b:2;

int c:3;

};

int main()

{

bit s;

char \*c=(char\*)&s;

cout<<sizeof(bit)<<endl;

\*c=0x99;

cout << s.a <<endl <<s.b<<endl<<s.c<<endl;

int a=-1;

printf("%x",a);

return 0;

}

输出为什么是

4

1

-1

-4

ffffffff

因为0x99在内存中表示为 100 11 001 , a = 001, b = 11, c = 100

当c为有符合数时, c = 100, 最高1为表示c为负数，负数在计算机用补码表示，所以c = -4;同理

b = -1;

当c为有符合数时, c = 100,即 c = 4,同理 b = 3

位域 ：

有些信息在存储时，并不需要占用一个完整的字节， 而只需占几个或一个二进制位。例如在存放一个开关量时，只有0和1

两种状态，用一位二进位即可。为了节省存储空间，并使处理简便，Ｃ语言又提供了一种数据结构，称为“位域”或“位段”。所谓“位域”是把一个字节中的二进位划分为几个不同的区域，

并说明每个区域的位数。每个域有一个域名，允许在程序中按域名进行操作。这样就可以把几个不同的对象用一个字节的二进制位域来表示。一、位域的定义和位域变量的说明位域定义与结构定义相仿，其形式为：

struct 位域结构名

{ 位域列表 };

其中位域列表的形式为： 类型说明符 位域名：位域长度

例如：

struct bs

{

int a:8;

int b:2;

int c:6;

};

位域变量的说明与结构变量说明的方式相同。 可采用先定义后说明，同时定义说明或者直接说明这三种方式。例如：

struct bs

{

int a:8;

int b:2;

int c:6;

}data;

说明data为bs变量，共占两个字节。其中位域a占8位，位域b占2位，位域c占6位。对于位域的定义尚有以下几点说明：

1. 一个位域必须存储在同一个字节中，不能跨两个字节。如一个字节所剩空间不够存放另一位域时，应从下一单元起存放该位域。也可以有意使某位域从下一单元开始。例如：

struct bs

{

unsigned a:4

unsigned :0 /\*空域\*/

unsigned b:4 /\*从下一单元开始存放\*/

unsigned c:4

}

在这个位域定义中，a占第一字节的4位，后4位填0表示不使用，b从第二字节开始，占用4位，c占用4位。

2. 由于位域不允许跨两个字节，因此位域的长度不能大于一个字节的长度，也就是说不能超过8位二进位。

3. 位域可以无位域名，这时它只用来作填充或调整位置。无名的位域是不能使用的。例如：

struct k

{

int a:1

int :2 /\*该2位不能使用\*/

int b:3

int c:2

};

从以上分析可以看出，位域在本质上就是一种结构类型， 不过其成员是按二进位分配的。

A. **(C/C++程序员)** 编写一个程序完成以下功能：

- 从磁盘上读入一个文件，这个文件可能是一个HTML格式的文件或者其他格式的

- parse此文件中可能包含的RSS链接 （可能0或多个）。

- 列出找到的RSS链接

注意：请不要使用任何Parse HTML, XML, RSS相关的库，直接写C/C++代码来处理读入的内容。请注意代码的风格，文件的分布安排、文件名、变量、样式等的命名、页面的美观等，这些都将影响您的得分。

提示：1、注意各种异常情况的处理。 2、注意代码的简洁和效率。

1英文解释：Appdomain,Assembly.Reflection,Actribute,Delegate,CLR.Postback..Dispose

2编写一个类，（构造，公有，私有方法，静态，私有变量）

3类和对象的区别

4简单的SQL子查询，左右连接语句

5SQL server事务处理机制

6DataReader和DatoAdpter的区别

7Dataset和DataAdpter的区别

8ADO和ADO。NET的区别

9，3个以上ASP。NET页面生命周期页面事件的名称

10ASP。NET页面缓存数据的几种形式

11开发中对XML技术的使用

12列出数据结构中几种树的名称

13HTTP1。0和HTTP1。1协议的区别

14列出2个以上UML图的名称

15列出您所知道的设计模式名称，并简述

16列出您平时喜欢看的技术书籍名称

17补全代码，不能用String,lenth等系统方法

获得传入字符串的长度，返回字符串长度

public int Getstring lenth(string sourcesting)

{

//请补全---------

}

第一部分：智力测试

（说明：此部分考察应聘者综合能力）

1. 小明一家过一座桥，过桥时是黑夜，所以必须有灯。现在小明过桥要1秒，小明的弟弟要3秒，小明的爸爸要6秒，小明的妈妈要8秒，小明的爷爷要12秒。每次此桥最多可过两人，而过桥的速度依过桥最慢者而定，而且灯在点燃后30秒就会熄灭。问小明一家如何过桥？
2. 100美元哪去了？

三个朋友住进了一家宾馆。结帐时，帐单总计3000美元。三个朋友每人分摊1000美元，并把这3000美元如数交给了服务员，委托他代到总台交帐，但在交帐时，正逢宾馆实施价格优惠，总台退还给服务员500美元，实收2500美元，服务员从这500美元中扣下了200美元，只退还了三位客人300美元，三人平分了这300美元，每人取回了100美元。这样，三个客人每人实际支付900美元，共支付2700美元，加上服务员扣的200美元，共计2900美元。那么这100美元的差额到哪里去了呢？

第二部分：知识面考察

（说明：此部分为考察应聘者知识面广度，包含了25条判断题。每一个应聘技术岗位的面试者都要回答本部分问题）

1. 目前市面上最新的PC机仍旧是基于数十年前提出的冯。诺伊曼体系计算机系统。
2. Unix操作系统和WindowsXP系统一样都是多任务系统，但不是实时操作系统。
3. 设计模式（design pattern）是一种用于解决模式匹配设计时的通用算法。
4. UML（Unified Modeling Language）只能描述系统要实现什么，而不能描述系统应该如何实现。
5. SOAP是一种新型的ROC的规范，他是由XML RPC发展演进过来的。
6. 一个域名唯一对应着一个IP地址。
7. 在现行的网络结构下，VOLP技术主要基于UDP协议来传输媒体流。
8. 在IP网络上的计算机通过广播ARP报文来检查是否有其他计算机使用了相同的IP地址。
9. TCP连接采用了3次握手机制来确认连接，但理论上，任意多次握手都不能完美地确认连接已经成功建立。
10. AT是一种网络地址翻译的技术，他能使得多台没有合法地址的计算机可以共享1个合法的IP地址访问Inernet.
11. 软件开发中往往测试和修改错误的时间会超过编码的时间。
12. 软件测试最重要的一件事情就是考虑各种出错的可能性，尤其要考虑一引起超出常规的操作。
13. indowsNT网络中域控制器（Domain Controller）起控制域名解析的作用。
14. WindowsNT组成的企业网络上Domain和work group模式可以并存。
15. 理论上最完美的防火墙也不能确保网络的绝对安全。
16. 启动一个线程是可以使用run（）或者start（）
17. sleep()和wait()方法在线程处理中作用是一样的。
18. java 中没有goto语句。
19. 数组和String 都有length()方法，分别用于获取 String的长度以及数组的长度。
20. 构造器Constructor不能被继承，因此不能重写Overriding ，但可以被重载Overloading。
21. SET中的元素是不允许重复的，可以通过==判断是否重复。
22. 当一个线程进入一个对象的一个Synchronized方法后，其它任何线程都不允许进入此对象的方法。
23. &和&&的区别在于&是位运算符，&&是布尔逻辑运算符。
24. HashMap和 Hashtable都实现了 Map接口，都不允许 null键，只是 Hashtable是同步的，因此效率低。
25. String=new String(“xyz”) 语句创建了一个String Object。

第三部分：问答题

（这一部分是笔试的重点）

1. 描述一下java中io包的结构以及处理的原理机制
2. overload和override的区别。Overloaded的方法是否可以改变返回值的类型？
3. 请描述一下java 中的collection frame work的结构以及分别如何使用。

第四部分：编程题

（这一部分是笔试的重点）

1. 写一个程序，接受从console输入n个数字（每个数字以“，”隔开），按照数字从大到小的顺序打印输出到console。请尽量使用高效率的方法。
2. singleton模式主要作用是保证在java 应用程序中，一个类class只有一个实例存在；请写一个程序实现该模式。

第五部分：附加题

（如果您觉得自己的专业知识比较优秀，而知识面不足，可以通过附加题来加分）

1. 使用java socket编程，实现一个echoserver：即客房端连接到服务器，发送任何命令，服务器收到命令后将该命令原样返回到客户端，然后断开连接。

注意：1）需要写出服务器以及客户端两端的程序

2）服务器应该可以支持多个客户端同时连接。

2、描述一下jvm加载class文件的原理机制。

描述一下你对dot net 的理解（越详细越好，最好有图形描述）

Java Script 与 Java 的区别？

Java Script 类型与dot net 类型的区别

请列举你所知道的文件上传实现方式

dot net 组件如何与JavaScript脚本交互；dot net 组件如何调用JavaScript函数；JavaScript如何调用dot net 公共属性和事件？

请说说你的优点和缺点（各3条）

能否适应经常出差

你的薪资要求是多少

公司如果选择你 你打算如何开展工作？

你对公司还有什么其他要求，例如福利？

介绍一下你的家人，他们对你的这份工作有什么看法？

你的亲戚朋友是怎么评价你的？

请做一个简单的自我介绍。

你是如何评价自己的？

大陆软件面试题

一,基本情况:

1. 从事开发工作\_\_\_\_\_年?
2. 是否是DBA?
3. 目前岗位?
4. 所做项目介绍.

二:数据库.

1. 你对哪种数据库最熟悉?使用该数据库多少年?简单描述对该数据库的理解.
2. 对其它较熟悉的商业数据库的了解及使用年限.
3. 如何将数据库(实例)备份和还原?(针对自己最熟悉的)
4. 如何知道一个表或视图的结构?
5. 描述INNER JOIN的功能和已在何种情况下使用INNER JOIN.
6. 如何让你的一个系统能支持异构数据库.你会怎么样设计?

三:DOT.NET

1. 使用DOT.NET多少年?主要开发哪些类型的系统?
2. 除DOT.NET外,你还会\_\_\_\_语言?
3. 描述你对微软DOT.NET的了解(最好图形描述)
4. 写一个函数计算当参数为N的值:1-2+3-4+5-6+7……+N
5. 描述DOT.NET插件在客户端的自动部署,包括自动下载和安全性自动设置.
6. 如何实现DOT.NET和WIN32API的调用.
7. 如何从WEB.SERVICE中取得用户的SESSION ID.
8. 如何通过WEB SERVICE启动或调用服务器端的WIN SERVICE安全机制的实现方式.
9. 如何从A.ASPX中将大量数据传给B.ASPX.且不能使用SESSION和APPLICATION.
10. 描述ASP.NET自定义控件的作用,开发过程安全配置过程.

写一个SQL存储过程，建立一个表USER 字段是姓名，年龄，职位，权限，然后向里面插入6条数据，然后查询出年龄大于18的所有信息。

用C语言写个函数，完成去掉字符串左边的空格，把右边的空格用0填充。调用之，然后打印出来。

用VB写出和数据库连接的过程。

选择题：

1. 从供选择的答案中，选出应填入下面叙述中（？）内的最确切的解答，把相应编号写在答卷的对应栏内。（第1到第3题）

在C++语言中引进了类的概念。类的定义包括类名，类的说明和类的实现。（A）是类的外部接口，（B）是类的内产表示，类具有（C）、（D）和（E）。有了（C）可以隐藏类对象内部实现的复杂细节，有效地保护内部所有数据不受外部破坏；（D）增强了类的共享机制，实现了软件的可重用性，简化系统的开发工作；（E）可实现函数重载和运算符重载。

供选择的答案

A、B：1、类的引用 2、类的说明 3、类的实现 4、类的标识 5、类的构造 6、类的成员说明

C~E：1、开放性 2、封装性 3、兼容性 4、继承性 5、多态性 6、可扩充性

1. 各种程序设计语言中都有多种数据类型，纯量数据类型和结构类型是两大基本数据类型。例如，（A）等都是前者，而（B）都是后者。

子程序通常分为两类：（C）和（D），前者是命令的抽象，后者是为了求值。

一些重要的程序语言（如C和PASCAL）允许过程的递归调用。通常用（E）来实现递归调用中的存储分配。

供选择的答案

A、B：1、双精度型、枚举和数组 2、数组、记录和联合 3、指针、数组和记录 4、实型、布尔型和指针 5、复型、指针和记录 6、实型、整型和联合

C、D：1、标准子程序 2、进程 3、过程 4、用户定义子程序 5、函数 6、分程序 7、模块

E：1、栈 2、堆 3、数组 4、链表

1. 每一棵树都能唯一地转换为它所对应的二叉树，树的这种二叉树表示对树的运算带来很大的好处。遍历（周游）是树形结构的一种重要运算，二叉树的基本组成部分是：根（N）、左子树（L）和右子树（R）。因而二叉树的遍历次序有六种。最常用的是三种：前序法（即按（A）次序），后序法（即按（B）次序）和中序法（也称对称序法，即按（C）次序）。这三种方法相互之间的关联。若已知一棵二叉树的前序序列是BEFCGDH，中序序列是FEBGCHD，则它的后序序列必是（D），而且可得该二叉树所表示的树的先根次序序列是（B）。

A~C：1、RLN 2、RNL 3、LRN 4、LNR 5、NLR 6、NRL

D、E：1、EFGHBCD 2、FEGHDCB 3、BCDEFGH 4、EFBGCHD 5、BEFCGDH 6、FEGBHDC

简答题：

1. 在公司人事数据库中有两张表：EMP表记录公司雇员的情况，DEPT表记录公司各个部门的情况，在下图中简要说明了这两张表的内容。

EMP表：

EMPNO ENAME JOB MGR HIREDATE SALARY DEPTNO

7001 SMITH SALEMAN 1001 17-DEC-98 800。00 10

7002 JONES MANAGER 1002 02-JAN-96 1500。00 20

7003 WILLIAM CLERK 1001 21-MAY-94 1200。00 20

（EMPNO：雇员编号、ENAME：雇员姓名、JOB：职务、MGR：所属部门经理编号、HIREDATE：起始工作日期、DEPTNO：所属部门编号）

DEPT表：

DEPTNO DNAME LOC

10 ACCOUNTING GUANGZHOU

20 SALES GUANGZHOU

30 RESARCH NANJING

请使用合适的SQL语句列出在广州工作的所有雇员的资料（包括其工作部门的名称、部门地址），结果请按部门排序。

2、。NET有那几种服务器控件？（举例说明用途，至少说明两种以上）（10分）

3、SESSION和APPLICATION各自代表什么意思？有什么不同？（5分）

1. SQLSERVER 的存储过程和触发器有什么不同之处？运行上有什么特点？（10分）
2. 请写出JAVASRIPT小程序，弹出一个带“确定”和“取消”有提示框。（10分）
3. CSS的含义是什么？有什么用途？（10分）
4. 请写出。NET分页的程序思路，或者有什么控件可以直接分页（写出该控件的名称和如何使用该控件）？
5. 请述明MVC的含义。（10分）
6. 请述明SOAP协议的含义和。NET的WEB SERVICE 服务器的架设方法。（20分）

（慧通）  
1 写出程序把一个链表中的接点顺序倒排  
typedef struct linknode  
{  
int data;  
struct linknode \*next;  
}node;  
//将一个链表逆置  
node \*reverse(node \*head)  
{  
node \*p,\*q,\*r;  
p=head;  
q=p->next;  
while(q!=NULL)  
{  
r=q->next;  
q->next=p;  
p=q;  
q=r;  
}

head->next=NULL;  
head=p;  
return head;  
}  
2 写出程序删除链表中的所有接点  
void del\_all(node \*head)  
{  
node \*p;  
while(head!=NULL)  
{  
p=head->next;  
free(head);  
head=p;  
}  
cout<<"释放空间成功!"<<endl;  
}  
3两个字符串，s,t;把t字符串插入到s字符串中，s字符串有足够的空间存放t字符串  
void insert(char \*s, char \*t, int i)  
{  
char \*q = t;  
char \*p =s;  
if(q == NULL)return;  
while(\*p!='\0')  
{  
p++;  
}  
while(\*q!=0)  
{  
\*p=\*q;  
p++;  
q++;  
}  
\*p = '\0';  
}

分析下面的代码：  
char \*a = "hello";  
char \*b = "hello";  
if(a= =b)  
printf("YES");  
else  
printf("NO");  
这个简单的面试题目,我选输出 no(对比的应该是指针地址吧),可在VC是YES 在C是NO  
lz的呢，是一个常量字符串。位于静态存储区，它在程序生命期内恒定不变。如果编译器优化的话，会有可能a和b同时指向同一个hello的。则地址相同。如果编译器没有优化，那么就是两个不同的地址，则不同

写一个函数，功能：完成内存之间的拷贝  
memcpy source code:  
    270 void\* memcpy( void \*dst, const void \*src, unsigned int len )  
    271 {  
    272    register char \*d;  
    273    register char \*s;  
    27  
    275    if (len == 0)  
    276       return dst;  
    277   
    278    if (is\_overlap(dst, src, len, len))  
    279       complain3("memcpy", dst, src, len);  
    280   
    281    if ( dst > src ) {  
    282       d = (char \*)dst + len - 1;  
    283       s = (char \*)src + len - 1;  
    284       while ( len >= 4 ) {  
    285          \*d-- = \*s--;  
    286          \*d-- = \*s--;  
    287          \*d-- = \*s--;  
    288          \*d-- = \*s--;  
    289          len -= 4;  
    290       }  
    291       while ( len-- ) {  
    292          \*d-- = \*s--;  
    293       }  
    294    } else if ( dst < src ) {  
    295       d = (char \*)dst;  
    296       s = (char \*)src;  
    297       while ( len >= 4 ) {  
    298          \*d++ = \*s++;  
    299          \*d++ = \*s++;  
    300          \*d++ = \*s++;  
    301          \*d++ = \*s++;  
    302          len -= 4;  
    303       }  
    304       while ( len-- ) {  
    305          \*d++ = \*s++;  
    306       }  
    307    }  
    308    return dst;  
    309 }  
公司考试这种题目主要考你编写的代码是否考虑到各种情况，是否安全（不会溢出）  
各种情况包括：  
１、参数是指针，检查指针是否有效  
２、检查复制的源目标和目的地是否为同一个，若为同一个，则直接跳出  
３、读写权限检查  
４、安全检查，是否会溢出  
memcpy拷贝一块内存，内存的大小你告诉它  
strcpy是字符串拷贝，遇到'\0'结束

/\* memcpy ─── 拷贝不重叠的内存块 \*/    
void memcpy(void\* pvTo, void\* pvFrom, size\_t size)  
{  
void\* pbTo = (byte\*)pvTo;  
void\* pbFrom = (byte\*)pvFrom;  
ASSERT(pvTo != NULL && pvFrom != NULL); //检查输入指针的有效性  
ASSERT(pbTo>=pbFrom+size || pbFrom>=pbTo+size);//检查两个指针指向的内存是否重叠  
while(size-->0)  
\*pbTo++ == \*pbFrom++;  
return(pvTo);  
}

华为面试题：怎么判断链表中是否有环？  
bool CircleInList(Link\* pHead)  
{  
if(pHead = = NULL || pHead->next = = NULL)//无节点或只有一个节点并且无自环  
return (false);  
if(pHead->next = = pHead)//自环  
return (true);  
Link \*pTemp1 = pHead;//step 1  
Link \*pTemp = pHead->next;//step 2  
while(pTemp != pTemp1 && pTemp != NULL && pTemp->next != NULL)  
{  
pTemp1 = pTemp1->next;  
pTemp = pTemp->next->next;  
}  
if(pTemp = = pTemp1)  
return (true);  
return (false);  
}

两个字符串，s,t;把t字符串插入到s字符串中，s字符串有足够的空间存放t字符串  
void insert(char \*s, char \*t, int i)  
{  
memcpy(&s[strlen(t)+i],&s[i],strlen(s)-i);  
memcpy(&s[i],t,strlen(t));  
s[strlen(s)+strlen(t)]='\0';  
}

1。编写一个 C 函数，该函数在一个字符串中找到可能的最长的子字符串，且该字符串是由同一字符组成的。  
char \* search(char \*cpSource, char ch)  
{  
         char \*cpTemp=NULL, \*cpDest=NULL;  
         int iTemp, iCount=0;  
         while(\*cpSource)  
         {  
                 if(\*cpSource == ch)  
                 {  
                          iTemp = 0;  
                          cpTemp = cpSource;  
                          while(\*cpSource == ch)   
++iTemp, ++cpSource;  
                          if(iTemp > iCount)   
iCount = iTemp, cpDest = cpTemp;  
        if(!\*cpSource)   
break;  
                 }  
                 ++cpSource;  
 }  
 return cpDest;  
}        
2。请编写一个 C 函数，该函数在给定的内存区域搜索给定的字符，并返回该字符所在位置索引值。  
int search(char \*cpSource, int n, char ch)  
{  
         int i;  
         for(i=0; i<n && \*(cpSource+i) != ch; ++i);  
         return i;  
}

一个单向链表，不知道头节点,一个指针指向其中的一个节点，问如何删除这个指针指向的节点？  
将这个指针指向的next节点值copy到本节点，将next指向next->next,并随后删除原next指向的节点。

#include <stdio.h>  
void foo(int m, int n)  
{  
    printf("m=%d, n=%d\n", m, n);  
}

int main()  
{  
    int b = 3;  
    foo(b+=3, ++b);  
    printf("b=%d\n", b);  
return 0;  
}  
输出：m=7,n=4,b=7(VC6.0)  
这种方式和编译器中得函数调用关系相关即先后入栈顺序。不过不同  
编译器得处理不同。也是因为C标准中对这种方式说明为未定义，所以  
各个编译器厂商都有自己得理解，所以最后产生得结果完全不同。  
因为这样，所以遇见这种函数，我们首先要考虑我们得编译器会如何处理  
这样得函数，其次看函数得调用方式，不同得调用方式，可能产生不同得  
结果。最后是看编译器优化。

2.写一函数，实现删除字符串str1中含有的字符串str2.  
第二个就是利用一个KMP匹配算法找到str2然后删除（用链表实现的话，便捷于数组）

/\*雅虎笔试题(字符串操作)   
给定字符串A和B,输出A和B中的最大公共子串。  
比如A="aocdfe" B="pmcdfa" 则输出"cdf"  
\*/  
//Author: azhen  
#include<stdio.h>  
#include<stdlib.h>  
#include<string.h>

char \*commanstring(char shortstring[], char longstring[])  
{  
int i, j;

char \*substring=malloc(256);

if(strstr(longstring, shortstring)!=NULL)              //如果……，那么返回shortstring  
return shortstring;

for(i=strlen(shortstring)-1;i>0; i--)                 //否则，开始循环计算  
{  
for(j=0; j<=strlen(shortstring)-i; j++){  
memcpy(substring, &shortstring[j], i);  
substring[i]='\0';  
if(strstr(longstring, substring)!=NULL)  
return substring;  
}  
}  
return NULL;  
}

main()  
{  
char \*str1=malloc(256);  
char \*str2=malloc(256);  
char \*comman=NULL;

gets(str1);  
gets(str2);

if(strlen(str1)>strlen(str2))                         //将短的字符串放前面  
comman=commanstring(str2, str1);  
else  
comman=commanstring(str1, str2);

printf("the longest comman string is: %s\n", comman);  
}

11.写一个函数比较两个字符串str1和str2的大小，若相等返回0，若str1大于  
str2返回1，若str1小于str2返回－1  
int strcmp ( const char \* src,const char \* dst)  
{  
        int ret = 0 ;  
        while( ! (ret = \*(unsigned char \*)src - \*(unsigned char \*)dst) && \*dst)  
{  
                ++src;  
++dst;  
}  
        if ( ret < 0 )  
                ret = -1 ;  
        else if ( ret > 0 )  
                ret = 1 ;  
        return( ret );  
}

3,求1000！的未尾有几个0（用素数相乘的方法来做，如72=2\*2\*2\*3\*3）;  
求出1->1000里,能被5整除的数的个数n1,能被25整除的数的个数n2,能被125整除的数的个数n3,  
能被625整除的数的个数n4.  
1000!末尾的零的个数=n1+n2+n3+n4;  
#include<stdio.h>  
#define NUM 1000

int find5(int num){  
int ret=0;  
while(num%5==0){  
num/=5;  
ret++;  
}  
return ret;  
}  
int main(){  
int result=0;  
int i;  
for(i=5;i<=NUM;i+=5)  
{  
result+=find5(i);  
}  
printf(" the total zero number is %d\n",result);  
return 0;  
}

1. 有双向循环链表结点定义为：   
struct node   
{ int data;   
struct node \*front,\*next;   
};   
有两个双向循环链表A，B，知道其头指针为：pHeadA,pHeadB，请写一函数将两链表中data值相同的结点删除   
BOOL DeteleNode(Node \*pHeader, DataType Value)  
{  
if (pHeader == NULL) return;

BOOL bRet = FALSE;  
Node \*pNode = pHead;  
while (pNode != NULL)  
{  
if (pNode->data == Value)  
{  
if (pNode->front == NULL)  
{  
pHeader = pNode->next;  
pHeader->front = NULL;  
}  
else  
{  
if (pNode->next != NULL)  
{  
pNode->next->front = pNode->front;  
}  
pNode->front->next = pNode->next;  
}

Node \*pNextNode = pNode->next;  
delete pNode;  
pNode = pNextNode;

bRet = TRUE;   
//不要break或return, 删除所有  
}  
else  
{  
pNode = pNode->next;  
}  
}

return bRet;  
}

void DE(Node \*pHeadA, Node \*pHeadB)  
{  
if (pHeadA == NULL || pHeadB == NULL)  
{  
return;  
}

Node \*pNode = pHeadA;  
while (pNode != NULL)  
{  
if (DeteleNode(pHeadB, pNode->data))  
{  
if (pNode->front == NULL)  
{  
pHeadA = pNode->next;  
pHeadA->front = NULL;  
}  
else  
{  
pNode->front->next = pNode->next;  
if (pNode->next != NULL)  
{  
pNode->next->front = pNode->front;  
}  
}  
Node \*pNextNode = pNode->next;  
delete pNode;  
pNode = pNextNode;  
}  
else  
{  
pNode = pNode->next;  
}  
}  
}  
2. 编程实现：找出两个字符串中最大公共子字符串,如"abccade","dgcadde"的最大子串为"cad"   
int GetCommon(char \*s1, char \*s2, char \*\*r1, char \*\*r2)  
{  
int len1 = strlen(s1);  
int len2 = strlen(s2);  
int maxlen = 0;

for(int i = 0; i < len1; i++)  
{  
for(int j = 0; j < len2; j++)  
{  
if(s1[i] == s2[j])  
{  
int as = i, bs = j, count = 1;  
while(as + 1 < len1 && bs + 1 < len2 && s1[++as] == s2[++bs])  
count++;

if(count > maxlen)  
{  
maxlen = count;  
\*r1 = s1 + i;  
\*r2 = s2 + j;  
}  
}  
}  
}  
3. 编程实现：把十进制数(long型)分别以二进制和十六进制形式输出，不能使用printf系列库函数  
char\* test3(long num) {  
char\* buffer = (char\*)malloc(11);  
buffer[0] = '0';  
buffer[1] = 'x';  
buffer[10] = '\0';

char\* temp = buffer + 2;  
for (int i=0; i < 8; i++) {  
temp[i] = (char)(num<<4\*i>>28);  
temp[i] = temp[i] >= 0 ? temp[i] : temp[i] + 16;  
temp[i] = temp[i] < 10 ? temp[i] + 48 : temp[i] + 55;  
}  
return buffer;  
}

输入N, 打印 N\*N 矩阵  
比如 N = 3，打印：

1  2  3  
8  9  4  
7  6  5

N = 4，打印：

1   2   3   4  
12  13  14  5  
11  16  15  6  
10  9   8   7  
解答：  
1 #define N 15  
int s[N][N];  
void main()  
{  
int k = 0, i = 0, j = 0;  
int a = 1;   
for( ; k < (N+1)/2; k++ )  
{  
while( j < N-k ) s[i][j++] = a++; i++; j--;  
while( i < N-k ) s[i++][j] = a++; i--; j--;  
while( j > k-1 ) s[i][j--] = a++; i--; j++;  
while( i > k )   s[i--][j] = a++; i++; j++;   
}  
for( i = 0; i < N; i++ )  
{  
for( j = 0; j < N; j++ )  
cout << s[i][j] << '\t';  
cout << endl;  
}  
}  
2 define MAX\_N  100  
int matrix[MAX\_N][MAX\_N];

/\*  
 \*（x,y）：第一个元素的坐标  
 \* start：第一个元素的值  
 \* n：矩阵的大小  
 \*/  
void SetMatrix(int x, int y, int start, int n) {  
    int i, j;

    if (n <= 0)    //递归结束条件  
        return;  
    if (n == 1) {  //矩阵大小为1时  
        matrix[x][y] = start;  
        return;  
    }  
    for (i = x; i < x + n-1; i++)   //矩阵上部  
        matrix[y][i] = start++;

    for (j = y; j < y + n-1; j++)   //右部  
        matrix[j][x+n-1] = start++;

    for (i = x+n-1; i > x; i--)     //底部  
        matrix[y+n-1][i] = start++;

    for (j = y+n-1; j > y; j--)     //左部  
        matrix[j][x] = start++;

    SetMatrix(x+1, y+1, start, n-2);   //递归  
}

void main() {  
   int i, j;  
   int n;

   scanf("%d", &n);  
   SetMatrix(0, 0, 1, n);  
     
   //打印螺旋矩阵  
   for(i = 0; i < n; i++) {  
      for (j = 0; j < n; j++)  
printf("%4d", matrix[i][j]);  
      printf("\n");  
   }  
}

斐波拉契数列递归实现的方法如下：  
 int  Funct( int n )  
{  
   if(n==0) return 1;  
   if(n==1) return 1;  
   retrurn  Funct(n-1) + Funct(n-2);  
}  
请问，如何不使用递归，来实现上述函数？  
请教各位高手！  
解答：int  Funct( int n )  //  n 为非负整数  
{  
   int a=0;  
   int b=1;  
   int c;  
   if(n==0) c=1;  
   else if(n==1) c=1;  
   else for(int i=2;i<=n;i++)  //应该n从2开始算起  
   {  
     c=a+b;  
     a=b;  
     b=c;  
   }  
   return c;  
}  
解答：  
现在大多数系统都是将低字位放在前面，而结构体中位域的申明一般是先声明高位。  
100  的二进制是 001 100 100  
低位在前   高位在后    
001----s3  
100----s2  
100----s1  
所以结果应该是 1  
如果先申明的在低位则：  
001----s1  
100----s2  
100----s3  
结果是 4  
1、原题跟little-endian，big-endian没有关系  
2、原题跟位域的存储空间分配有关，到底是从低字节分配还是从高字节分配，从Dev C++和VC7.1上看，都是从低字节开始分配，并且连续分配，中间不空，不像谭的书那样会留空位  
3、原题跟编译器有关，编译器在未用堆栈空间的默认值分配上有所不同，Dev C++未用空间分配为  
01110111b，VC7.1下为11001100b,所以在Dev C++下的结果为5，在VC7.1下为1。

注：PC一般采用little-endian，即高高低低，但在网络传输上，一般采用big-endian，即高低低高，华为是做网络的，所以可能考虑big-endian模式，这样输出结果可能为4

判断一个字符串是不是回文  
int IsReverseStr(char \*aStr)  
{  
int i,j;  
int found=1;  
if(aStr==NULL)  
return -1;  
j=strlen(aStr);  
for(i=0;i<j/2;i++)  
if(\*(aStr+i)!=\*(aStr+j-i-1))  
{  
found=0;  
break;  
}  
return found;  
}

Josephu 问题为：设编号为1，2，… n的n个人围坐一圈，约定编号为k（1<=k<=n）的人从1开始报数，数到m 的那个人出列，它的下一位又从1开始报数，数到m的那个人又出列，依次类推，直到所有人出列为止，由此产生一个出队编号的序列。

数组实现：  
#include <stdio.h>  
#include <malloc.h>  
int Josephu(int n, int m)  
{  
  int flag, i, j = 0;  
  int \*arr = (int \*)malloc(n \* sizeof(int));  
  for (i = 0; i < n; ++i)  
    arr[i] = 1;  
  for (i = 1; i < n; ++i)  
  {  
    flag = 0;  
    while (flag < m)  
    {  
      if (j == n)  
        j = 0;  
      if (arr[j])  
        ++flag;  
      ++j;  
    }  
    arr[j - 1] = 0;  
    printf("第%4d个出局的人是：%4d号\n", i, j);  
  }  
  free(arr);  
  return j;  
}  
int main()  
{  
  int n, m;  
  scanf("%d%d", &n, &m);  
  printf("最后胜利的是%d号！\n", Josephu(n, m));  
  system("pause");  
  return 0;  
}  
链表实现：  
#include <stdio.h>  
#include <malloc.h>  
typedef struct Node  
{  
  int index;  
  struct Node \*next;  
}JosephuNode;  
int Josephu(int n, int m)  
{  
  int i, j;  
  JosephuNode \*head, \*tail;  
  head = tail = (JosephuNode \*)malloc(sizeof(JosephuNode));  
  for (i = 1; i < n; ++i)  
  {  
    tail->index = i;  
    tail->next = (JosephuNode \*)malloc(sizeof(JosephuNode));  
    tail = tail->next;  
  }  
  tail->index = i;  
  tail->next = head;  
    
  for (i = 1; tail != head; ++i)  
  {  
    for (j = 1; j < m; ++j)  
    {  
      tail = head;  
      head = head->next;  
    }  
    tail->next = head->next;  
    printf("第%4d个出局的人是：%4d号\n", i, head->index);  
    free(head);  
    head = tail->next;  
  }  
  i = head->index;  
  free(head);  
  return i;  
}  
int main()  
{  
  int n, m;  
  scanf("%d%d", &n, &m);  
  printf("最后胜利的是%d号！\n", Josephu(n, m));  
  system("pause");  
  return 0;  
}

已知strcpy函数的原型是：  
        char \* strcpy(char \* strDest,const char \* strSrc);  
    1.不调用库函数，实现strcpy函数。  
    2.解释为什么要返回char \*。  
    解说：  
    1.strcpy的实现代码  
        char \* strcpy(char \* strDest,const char \* strSrc)  
        {  
                if ((strDest==NULL)||(strSrc==NULL)) <file://[/1>]  
                        throw "Invalid argument(s)"; //[2]  
                char \* strDestCopy=strDest;  <file://[/3>]  
                while ((\*strDest++=\*strSrc++)!='\0'); <file://[/4>]  
                return strDestCopy;  
        }  
    错误的做法：  
    [1]  
    (A)不检查指针的有效性，说明答题者不注重代码的健壮性。  
    (B)检查指针的有效性时使用((!strDest)||(!strSrc))或(!(strDest&&strSrc))，说明答题者对C语言中类型的隐式转换没有深刻认识。在本例中char \*转换为bool即是类型隐式转换，这种功能虽然灵活，但更多的是导致出错概率增大和维护成本升高。所以C++专门增加了bool、true、false三个关键字以提供更安全的条件表达式。  
    (C)检查指针的有效性时使用((strDest==0)||(strSrc==0))，说明答题者不知道使用常量的好处。直接使用字面常量（如本例中的0）会减少程序的可维护性。0虽然简单，但程序中可能出现很多处对指针的检查，万一出现笔误，编译器不能发现，生成的程序内含逻辑错误，很难排除。而使用NULL代替0，如果出现拼写错误，编译器就会检查出来。  
    [2]  
    (A)return new string("Invalid argument(s)");，说明答题者根本不知道返回值的用途，并且他对内存泄漏也没有警惕心。从函数中返回函数体内分配的内存是十分危险的做法，他把释放内存的义务抛给不知情的调用者，绝大多数情况下，调用者不会释放内存，这导致内存泄漏。  
    (B)return 0;，说明答题者没有掌握异常机制。调用者有可能忘记检查返回值，调用者还可能无法检查返回值（见后面的链式表达式）。妄想让返回值肩负返回正确值和异常值的双重功能，其结果往往是两种功能都失效。应该以抛出异常来代替返回值，这样可以减轻调用者的负担、使错误不会被忽略、增强程序的可维护性。  
    [3]  
    (A)忘记保存原始的strDest值，说明答题者逻辑思维不严密。  
    [4]  
    (A)循环写成while (\*strDest++=\*strSrc++);，同[1](B)。  
    (B)循环写成while (\*strSrc!='\0') \*strDest++=\*strSrc++;，说明答题者对边界条件的检查不力。循环体结束后，strDest字符串的末尾没有正确地加上'\0'。

1. 已知2张表

招标标号, 时间, 地区编码id 地区号, 地区名

announce (TitleC, CreateDate, ZoneId) 招标信息表Zone(ZoneId, ZoneNm)

本机名为host，数据库名为sql，用asp编写代码，在网页上显示出南京地区招标信息中，按时间倒序排列的前5名招标信息，要显示出标号、时间、地区名（要以表格形式显示出来）。

1. 用for… next, if… then, do… loop, select… case, exit各写一段可运行的程序。
2. 程序运行结果：

<% Test = “Text”

TestA

TestB

Response.write(Test)

Sub TestA()

Test = “TestA”

End Sub

Sub TestB()

Test = “TestB”

End Sub

%>

4、5、6、7前提：

已有一个ConnTestDB连接(Connection)对象，并已连接好数据库并open，在后面的程序中可直接使用，有两张表：

Id

Student {

Name

StuId

Score {Chinese 语文

Math 数学

1. 用ASP代码在网页上显示如下内容：

姓名 性别

AAA M

BBB F

CCC M

…… ……

总共有多少个学生

1. 同上：显示如下内容：

|  |  |  |  |
| --- | --- | --- | --- |
| 姓名 | 语文分数 | 数学分数 | 总成绩 |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

1. 插入一段数据

张三，男，语文成绩：80，数学成绩60

用一段ASP代码将以上数据库分别插入两张表中

**NET程序员上机测试题(2)**

请按下列要求设计一套网上考试系统

**一、系统总体功能要求**

1. 基于B/S或者C/S结构，前端使用VS.NET2003采用C＃语言,数据库SQL Server2000；
2. 数据库设计；
3. 实现新考生信息的增加、删除、保存、修改；
4. 考生登录；
5. 试题生成（1、防止考生刷新页面，2、随机生成试题）；

备注：由于加载试题页面时将从数据库中随机抽取试题，所以应防止考生刷新页面(刷新页面会重新生成新的试题)。

1. 交卷评分。

**二、主要数据库表单设计（请理解功能要求后，自己灵活设计，以下仅供参考）**

1、考生信息表

|  |  |  |
| --- | --- | --- |
| 序号 | 描述 | 字段名称、类型等 |
|  | 考号 |  |
|  | 考生姓名 |  |
|  | 是否登录考试 |  |
|  | 得分 |  |
|  | 随机生成的试题答案 |  |

2、多选题库表

|  |  |  |
| --- | --- | --- |
| 序号 | 描述 | 字段名称、类型等 |
|  | 编号 |  |
|  | 题目 |  |
|  | Choicel1 |  |
|  | Choicel2 |  |
|  | Choice3 |  |
|  | Choicel4 |  |
|  | 答案(0表示未选，1表示选择，如选择ABD就用1101表示) |  |

**南京机遇软件公司面试题**

1. 设集合N={0，1，2。。。n}，f为N到N的函数，且

f（x）={f(f(+11)) 0<=x<=90

x-10 x>90

}

经计算f(90)=81，f(89)=81,f(49)=\_\_\_\_\_。

1. 有一个student表（学生，姓名，系名，课程名，成绩），查询至少修了四门课程的学生学号，姓名以及平均成绩的SQL语句。
2. 内存地址从4000H到43FFH，共有\_\_\_\_个内存单元，若该内存每个存贮单元可以存储16位二进制数，并用4片存储芯片构成，则芯片容量为\_\_\_.
3. 已知递归函数f(n)的功能是计算1+2+3…n，且n>=1,应采用的代码段是\_\_\_\_\_.
4. 任何一个基于比较的内部排序算法，若对6个元素进行排序，最坏情况下所需要的比较次数是几次。
5. int a=-67,b=116,c=78;

int d=~a|b&c;

System.out.println(d);

的结果为：\_\_\_\_.

设有关系EMP（ENO，ENAME，SALARY，DNO），其中各属性的含义依次为职工号，职工姓名，工资和所在部门号，以及关系DEPT（DNO，DNAME，MANAGER），其中各属性的含义依次为部门号，部门名称，部门经理的职工号。（用SQL语句）

1. 写出查询001号职工所在的部门名称
2. 查询人数不少于10人的各部门的职工人数，平均职工工资，并按平均职工工资从低到高排序
3. 写出“销售部“的哪些工资数额低于600的职工的工资上调10%
4. 写出工资数高于6000的各部门经理的工资下调10%

将浮点型的金额值转换成为大写的金额字符串，小数超过2位的部分四舍五入，例：

1003.251🡪壹千零叁元贰角伍分。

1.面向对象的思想主要包括什么？2.什么是ASP.net中的用户控件4.列举一下你所了解的XML技术及其应用5.值类型和引用类型的区别？写出C#的样例代码。6.ADO.net中常用的对象有哪些？分别描述一下。8.C#中的接口和类有什么异同。9.。net中读写数据库需要用到哪些类？他们的作用10.UDP连接和TCP连接的异同。11.ASP.net的身份验证方式有哪些？分别是什么原理？12.进程和线程分别怎么理解？13.什么是code-Behind技术。15..net中读写XML的类都归属于哪些命名空间？16.解释一下UDDI、WSDL的意义及其作用。17.什么是SOAP,有哪些应用。18.如何部署一个ASP.net页面。19.如何理解.net中的垃圾回收机制。20.常用的调用webservice方法有哪些？21描述Cookie和Session的作用，区别和各自的应用范围，Session工作原理。

填空：

1、address bus为16Bit的CPU（8085,Z80）可能表现的address空间，用16进位来表示为OH~ H,同样，address bus 为32Bit的CPU(68040)用2进位的指数（例：2）表示为：OH~ 。

2、大容量的内存空间（Memory）的单位用Kbyte,Mbyte来表示，其换算方法是：1Mbyte= Kbyte= Byte

3、Computer的时间单位是：u sec,nsec.其换算方式是：

1Sec= msec= u sec= nsec

下列数值变换

1、0101（2）＝﹥ （10） （2进位＝﹥10进位）

2、1011（2）＝﹥ （10） （2进位＝﹥10进位）

3、15（10）＝﹥ （2） （10进位到2进位）

4、27（10）＝﹥ （2） （10进位到2进位）

5、34（10）＝﹥ （16） （10进位到16进位）

6、58（10）＝﹥ （10） （10进位到16进位）

7、54（16）＝﹥ （16） （16进位到10进位）

8、7E（16）＝﹥ （10） （16进位到10进位）

9、10A（16）＝﹥ （10） （16进位到10进位）

10、19（10）＝﹥ （8） （10进位到8进位）

下图为单方向LIST。东京处在LIST的先头，其地址存放在下一个数值的地址，名古屋在LIST的最后，其值为0

地址为150处的热海处，插入“新横滨”与“静冈”后，以下的正确答案是：

1. 热海的指针是50，东京的指针是150
2. 热海的指针是50，名古屋的指针是150
3. 热海的指针是70，静冈的指针是150
4. 热海的指针是70，新横滨的指针是150
5. 热海的指针是70，新横滨的指针是150

先头数据的指针：10

地址 数据部分 批针

10 东京 50

30 名古屋 0

50 新横滨 70

70 静冈 90

90 滨松 30

150 热海

阅读下列流程图说明及流程图，回答1到3的问题

流程图说明：

1. 将9\*9的二次元数组画像数据，按顺时针方向旋转90度后，可获得数组B
2. 下图所示实行其流程的算法（横向是j方向，纵向是i方向）

数组A

1 2 3 4 5 6 7 8 9

1. 0 0 0 0 0 0 0 0 0
2. 0 \* \* \* \* \* \* \* 0
3. 0 0 0 0 \* 0 0 0 0
4. 0 0 0 0 \* 0 0 0 0
5. 0 0 \* 0 \* \* \* 0 0
6. 0 0 \* 0 \* 0 0 0 0
7. 0 0 \* 0 \* 0 0 0 0
8. 0 \* \* \* \* \* \* \* 0
9. 0 0 0 0 0 0 0 0 0

数组B

1 2 3 4 5 6 7 8 9

1 0 0 0 0 0 0 0 0 0

1. 0 \* 0 0 0 0 0 \* 0
2. 0 \* \* \* \* 0 0 \* 0
3. 0 \* 0 0 0 0 0 \* 0
4. 0 \* \* \* \* \* \* \* 0
5. 0 \* 0 0 \* 0 0 \* 0
6. 0 \* 0 0 \* 0 0 \* 0
7. 0 \* 0 0 0 0 0 \* 0
8. 0 0 0 0 0 0 0 0 0
9. 数组A与数组B从上方第I与从左第J位的数组是：A（i,j）,B（i,j）
10. 开始🡪I方向重复I=1，2，……9🡪J方向重复J=1，2，……9

🡪★A（i,j）至B（j,10-i）🡪j方向重复🡪I方向重复🡪结束

问

* 1. 逆时针方向旋转90度（即：A（1，1）🡪B(9,1)……），上面的流程图中★处应如何更改？
  2. 旋转180度后，上面的流程图中★处应如何更改？
  3. 如下列数组所示，如果左右方向向颠倒后，上面的流程图中★处应如何更改？

数组A

1 2 3 4 5 6 7 8 9

1、0 0 0 0 0 0 0 0 0

2、0 \* \* \* \* \* \* \* 0

3、0 0 0 0 \* 0 0 0 0

4、0 0 0 0 \* 0 0 0 0

5、0 0 \* 0 \* \* \* 0 0

6、0 0 \* 0 \* 0 0 0 0

7、0 0 \* 0 \* 0 0 0 0

8、0 \* \* \* \* \* \* \* 0

9、0 0 0 0 0 0 0 0 0

数组B

1 2 3 4 5 6 7 8 9

1、0 0 0 0 0 0 0 0 0

1. 0 \* \* \* \* \* \* \* 0
2. 0 0 0 0 \* 0 0 0 0
3. 0 0 0 0 \* 0 0 0 0
4. 0 0 \* \* \* \* \* 0 0
5. 0 0 0 0 \* 0 \* 0 0
6. 0 0 0 0 \* 0 \* 0 0
7. 0 \* \* \* \* \* \* \* 0
8. 0 0 0 0 0 0 0 0 0

解答选项：

* 1. A(I,j)🡪B(10-i,10-j)
  2. A(I,j)🡪B(10-i,j)
  3. A(I,j)🡪B(10-j,10-i)
  4. A(I,j)🡪B(10-j, i)
  5. A(I,j)🡪B(10-I,10-i)
  6. A(I,j)🡪B(i,10-j)
  7. A(I,j)🡪B(i,j)
  8. A(I,j)🡪B(j,10-i)
  9. A(I,j)🡪B(j,10-j)
  10. A(I,j)🡪B(j,i)

4. #include “stdio.h” 和 include <stdio.h> 的区别

3. “abcdefghi”向后移动2位 “hicdefgh” 写一函数实现此功能 （\*p为要移动的字符串，nsteps为移动的位数）

void pp (char \*p, int nsteps)

{

//请填写

}

1. Bool bstate, float fnum, int nierp, 写出这几个数与零值的比较

专业题：C语言题目

1 写一个函数，将一个3\*3矩阵转置。

2 有一个链表，其中包含姓名，性别，年龄，请写一个子程序，输入一个年龄，如果输入的年龄是链表结点中的年龄，删除该结点。

智力题目

1 某大学的一个寝室住8个人，其中有6个人会下象棋，有5个人会下围棋，有4个人会下国际象棋，问3种棋都会下的最多有几人，最少有几人？

2 有80个球，其中有一个是假球，假球比真球轻，放在没有砝码的天平上称，只能称4次，找出假球。

3 电影院座位一排有13个座位，已经坐了若干人，新坐上去的一个人不论怎么坐都与其它人相邻，问原来坐好的至少有几人？（方法有多种，只写一种）

4 有如下表格，表格中数字为1---31

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1 |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  | 31 |

表A

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 2 |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  | 31 |

表B

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  | 31 |

表C

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 8 |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  | 31 |

表D

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  | 31 |

表E

在你心中不论想一个什么数字（在1---31之间），都能填到表格当中，并能知道你想到的那个数字是多少，比如数字11，填到上面的表A，表B，表D中，设计这样的表格

诺思信科技（南京）有限公司.NET笔试题答案

一、C#基础题

**1、传入某个属性的SET方法的隐含参数的名称是什么？**

value，它的类型和属性所声明的类型相同。

**2、C#支持多重继承吗？**

不支持。可以用接口来实现。

**3、C#用多种修饰符来表达类的不同性质。根据其保护级C#的类有五种不同的限制修饰符，请写出并指出它们之间的区别是什么？**

public 可以被任意存取

protected只可以被本类和其继承子类存取

internal只可以被本组合体（Assembly）内所有的类存取，组合体是C#语言中类被组合后的逻辑单位和物理单位，其编译后的文件扩展名往往是“.DLL”或“.EXE”。

protected internal唯一的一种组合限制修饰符，它只可以被本组合体内所有的类和这些类的继承子类所存取。

private只可以被本类所存取。

**4、C#提供一个默认的无参构造函数，当我实现了另外一个有一个参数的构造函数时，还想保留这个无参数的构造函数。这样我应该写几个构造函数？**

两个，一旦你实现了一个构造函数，C#就不会再提供默认的构造函数了，所以需要手动实现那个无参构造函数。

**5、重载和覆写有什么区别？**

重载提供了对一个方法签名的不同参数调用的实现。覆写提供了子类中改变父类方法行为的实现。

**6、在方法定义中，virtual有什么含义？**

被virtual修饰的方法可以被子类覆写

**7、能够将非静态的方法覆写成静态方法吗？**

不能，覆写方法的签名必须与被覆写方法的签名保持一致，除了将virtual改为override。

**8、可以覆写私有的虚方法吗？**

不可以，甚至子类中无法访问父类中的私有方法

**9、能够阻止某一个类被其他类继承吗？**

可以，使用关键字sealed

**10、能够实现允许某个类别继承，但不允许其中的某个方法被覆写吗？**

可以，标记这个类为public，并标记这个方法为sealed。

**11、什么是抽象类（abstract class）?**

一种不可以被实例化的类。抽象类中一般含有抽象方法，当然也可有具体实现。继承类只有实现过所有抽象类的抽象方法后才能被实例化。

**12、接口（interface）是什么？**

只含有公有抽象方法（public abstract method）的类。这些方法必须在子类中被实现。

**13、为什么不能指定接口中方法的修饰符？**

接口中的方法用来定义对象之间通信的契约，指定接口中的方法为私有或保护没有意义。它们默认为公有方法。

**14、接口和抽象类的区别是什么？**

接口中所有方法必须是抽象的，并且不能指定方法的访问修饰符。抽象类中可以有方法的实现，也可以指定方法的访问修饰符。

**15、如何区别重载方法？**

不同的参数类型，不同的参数个数，不同的参数顺序

16、解释一下ajax及实现原理

17、列出常用的使用javascript操作xml的类包

二、程序题

**1、请问，Class1.Count的值是多少？（ 3 ）**

class Class1

{

public static int Count = 0;

static Class1()

{

Count++;

}

public Class1()

{

Count++;

}

}

Class1 o1 = new Class1();

Class1 o2 = new Class1();

Console.Write(“Class1.Count的值是”+ Count)

**2、请问，此程序输出的结果是： Class1**

abstract class BaseClass

{

public virtual void MethodA()

{

Console.WriteLine("BaseClass");

}

public virtual void MethodB()

{

}

}

class Class1 : BaseClass

{

public void MethodA()

{

Console.WriteLine("Class1");

}

public override void MethodB()

{

}

}

class Class2 : Class1

{

new public void MethodB()

{

}

}

class MainClass

{

public static void Main(string[] args)

{

Class2 o = new Class2();

o.MethodA();

}

}

**3、请用你最熟悉的语言编写一个排序的函数（方法）**

**略**

三、思维题

**1**．某一密码仅使用K、L、M、N、O共5个字母，密码中的单词从左向右排列，密码单词必须遵循如下规则：

（1） 密码单词的最小长度是两个字母，可以相同，也可以不同

（2） K不可能是单词的第一个字母

（3） 如果L出现，则出现次数不止一次

（4） M不能使最后一个也不能是倒数第二个字母

（5） K出现，则N就一定出现

（6） O如果是最后一个字母，则L一定出现

问题一：下列哪一个字母可以放在LO中的O后面，形成一个3个字母的密码单词？B

A) K B)L C) M D) N

问题二：如果能得到的字母是K、L、M，那么能够形成的两个字母长的密码单词的总数是多少？A

A)1个 B)3个 C)6个 D）9个

问题三：下列哪一个是单词密码？C

A) KLLN B) LOML C) MLLO D)NMKO

问题四：3个相同的字母可以构成的不同密码单词的总数是多少？B

A) 1个 B)2个 C) 3个 D) 4个

问题五：单词密码MMLLOKN可以通过下列变成另一个密码单词，除了： C

A) 用N代替每个L B)用O代替每一个M C)用O代替N D)把O移到N的左边

1．做一个javascript定时器应用的小例子。

2．设计一个新闻发布系统，完成对新闻的添加，页面上使用DataList（控件）以列表的形式显示最新5条新闻标题，单击页面可以打开新窗口显示新闻的详细内容

1. 设计数据库

表名 NewsInfo

字段名 含义 类型

NewID 主键 int(自动增长)

NewTitle 新闻标题 nvarchar(100)

NewInfo 新闻内容 nvarchar(1000)

按照以上要求 写出SQL脚本，并且写出对表添加，更新和删除的存储过程

1. 显示页面
2. 添加新闻页面
3. 整个功能按照三层的方式来完成。

笔试：

大概分四个部分

第一部分：基本信息的填写。

第二部分：程序题。

1.有两张表（库存表和销售表），要求查出每天的放库存量（量关键一题）

2.两个VB的题目（具体内容记不清楚了）

3.在ASP中设计报表。

4.给一个需求，要求写出设计思路。

其它还有几个数据库的题，记不清了。

第三个部分：是不记分的，主要是些与编程无关的问题（比如说中国的十大钢铁企业，现在最流行的软件等）

第四个部分：是财务方面的题，上次基本上没写（名词解释）。

乾坤主要是用VB做MS系统的，用VB和SQL2000开发。

笔试：

大概分四个部分

第一部分：基本信息的填写。

第二部分：程序题。

1.有两张表（库存表和销售表），要求查出每天的放库存量（量关键一题）

2.两个VB的题目（具体内容记不清楚了）

3.在ASP中设计报表。

4.给一个需求，要求写出设计思路。

其它还有几个数据库的题，记不清了。

第三个部分：是不记分的，主要是些与编程无关的问题（比如说中国的十大钢铁企业，现在最流行的软件等）

第四个部分：是财务方面的题，上次基本上没写（名词解释）。

乾坤主要是用VB做MS系统的，用VB和SQL2000开发。

顺举：

1. 项目流程、数据结构概念；
2. 面向对象与面向过程的区别；
3. 重载和重写的区别；
4. B/S和C/S的区别
5. JavaScript、J Script和VBScript的区别
6. 补码
7. 数据库查询的应用、触发器。

四达荣合C/C++笔试题

一:填空:

1. 现有类public void CCompany,请写出它的拷贝构造函数\_\_\_\_\_\_\_\_\_\_\_\_\_,析构函数\_\_\_\_\_\_\_\_\_\_\_,运算符+=的重载函数\_\_\_\_\_\_\_\_\_\_\_\_.
2. int i = 10, j = 10, k = 3, k /= i + j, k = \_\_\_\_\_\_\_\_\_\_\_.
3. 现有函数Setattr(int a, int b, int c). 已知b无初始值, a的初始值等于1, c的初始值等于0, 那么此函数的声明形式为:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.
4. 在C中,预编译定义的关键字是\_\_\_\_\_\_\_\_\_\_, 常量定义的关键字是:\_\_\_\_\_\_\_\_\_\_\_\_\_.
5. 在C中,检测字符串长度的函数是:\_\_\_\_\_\_\_\_\_\_\_\_\_\_, 拷贝字符串的函数是:\_\_\_\_\_\_\_\_\_,

比较两个字符串的函数是:\_\_\_\_\_\_\_\_\_\_\_\_\_\_,连接字符串的函数是:\_\_\_\_\_\_\_\_\_\_\_,使用这些函数需调用的头文件是:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_.

1. 已知字符串char[] topic = “hello,hello hello money!”, sizeof(topic) = \_\_\_\_\_\_\_\_\_, strlen(topic)=\_\_\_\_\_\_\_\_\_\_\_.

二:问答题:

1. main函数的参数argc和argv的作用是什么?如何获得某个应用程序的第三个命令行参数.
2. if( n == 10) ---1.

if( 10 == n) ---2

哪一种写法是正确的, 为什么?

1. -🡪1.const char \*p;

-🡪2.char const \*p;

-🡪3.char \* const p;

-🡪4 const char \* const p;

请描述一下以上四种写法的区别.

1. 请编写一个类, 使其具有整形变量i的i++以及++i的功能.
2. 请写出C++中的所有访问级别,并比较它们的不同.
3. 现有表department(dep\_Id, depart\_name, wage), 查询除human resource以外的部门名及该部门的平均工资,并按字符的顺序对查询结果进行排序.

三:分析题:

考点:函数的按值传递及引用传递.

多态(基类指针指向派生类对象, 基类引用指向派生类对象).

函数重载,

静态成员(静态方法).

四.编程题.

1. 已知字符串操作函数中有字符串拷贝函数strcpy().现有自定义字符串拷贝函数Strcpy的声明: char\* Strcpy(char\* strDest, const char\* strSrc).

-🡪1.实现该自定义字符串拷贝函数.

-🡪2.该函数具备拷贝字符串的功能,为什么要返回char\*型变量.

2.已知一个链表中存放了10个数,请对该链表进行逆置.以下是该链表结点的声明:

struct node

{

int num;

struct node\* next;

}

3.-->1.为一个整型数组添加一组随机数.

--🡪2.分区排序:从数组中任意选取一个数作为参照数,遍历该数组,将大于此数的数的数放在一边, 将小于此数的数的数放在另一边, 依次类推,完成排序.

**苏慧面试题**

1. 一个字长的二进制位数
2. 某二叉树的先序和后序序列正好相反，则该二叉树一定是什么的二叉树？
3. 什么是XML？请说说它的特点。
4. 什么是纯虚函数，什么是多态？关系？C++和JAVA的区别是什么？
5. 数据库中事务的概念及特点是什么？
6. 什么是UML？说说它的种类。
7. 什么是模态窗口？什么是非模态窗口？
8. 输出Helloworld
   1. EJB
   2. .net webservice

已知链表的头结点head,写一个函数把这个链表逆序。

1. 简单描述包和dll异同
2. 简述parent和owner的区别
3. 谈一谈你对vcl framework的理解
4. 简单谈一下你对delphi接口机制的理解
5. web开发客户端程序和服务器端程序的区别
6. ASP 的几大内置对象。
7. 谈谈你以前所做项目对文件和图片的上传和显示所用的方法。
8. 谈你对ASP的理解。
9. 简单描述包和dll异同
10. 简述parent和owner的区别
11. 简述delphi是如何封装windows消息机制
12. 简单介绍一下delphi模式下三层开发
13. 谈一谈你对vcl framework的理解
14. 简单谈一下你对delphi接口机制的理解
15. web开发客户端程序和服务器端程序的区别
16. ASP 的几大内置对象。
17. 谈谈你以前所做项目对文件和图片的上传和显示所用的方法。
18. 谈你对ASP的理解。

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| 学号 | 姓名 | 年龄 | 班级 | 总成绩 | |
| 课程1 | 课程2 |
|  |  |  |  |  | |
|  |  |  |  |  | |

1. 用HTML语句写该表,标题为粗体

再用CSS描述该表

1. 用脚本语言在一个文本框中判断如果输入非数字,在失去焦点时则弹出对话框”请输入非数字”以提醒用户.
2. 有一个客户表在TEST数据库中创建,有如下字段,

客户ID,客户姓名,联系人,联系人地址,邮编

4. 创建该表后,用SQL语句插入两条数据:0001,张三 0002,李四

5. 有一个订单信息表,

客户ID,商品名称,商品数量,订购日期

1. INNER JOIN与LEFT JOIN有什么区别
2. 用该表与客户表建立一个INNER JOIN.
3. 根据客户和商品名称统计商品数量,最后按客户名排序
4. 按客户和日期查询商品数量

6. NET

1. 建立客户表的操作类,有添加,更新,删除操作
2. .ascx 的后缀名和.asmx后缀名各表示什么?
3. 什么是面向对象

答：万物都是对象，其主要特征：封装、继承、多态

1. 怎样实现多态

答：1.通过对象直接调用成员函数时，始终默认使用该对象的类的成员函数（除非用::显示指定类名）。2.通过指向对象的指针或引用调用成员函数时：如果该函数是实函数，则调用该指针或引用的类的成员函数；如果该函数是虚函数，则调用该指针或引用指向的对象的类的成员函数。

1. out保留字怎么使用，什么时候使用

答：有时为了从一个函数中返回多个值，我们需要使用out关键字，把输出值赋给通过引用传递给方法的变量（也就是参数）。但C#要求变量再被引用的前必须初始化。在调用该方法时，还需要添加out关键字

1. 什么时候使用抽象类，什么时候用接口

答：接口用于规范，抽象类用于共性。接口中只能声明方法，属性，事件，索引器。而抽象类中可以有方法的实现，也可以定义非静态的类变量。抽象类是类，所以只能被单继承，但是接口却可以一次实现多个。抽象类可以提供某些方法的部分实现，接口不可以.抽象类的实例是它的子类给出的。接口的实例是实现接口的类给出的。再抽象类中加入一个方法，那么它的子类就同时有了这个方法。而在接口中加入新的方法，那么实现它的类就要重新编写（这就是为什么说接口是一个类的规范了）。接口成员被定义为公共的，但抽象类的成员也可以是私有的、受保护的、内部的或受保护的内部成员（其中受保护的内部成员只能在应用程序的代码或派生类中访问）。此外接口不能包含字段、构造函数、析构函数、静态成员或常量。

1. 访问修饰符各种访问范围

答：public 不受限制的访问

Private 只能在本类中访问

Protected 本类和继承与他的类中的都能访问

Internal 同一个应用程序或类库中

1. new的两种用法

答：实例话对象，隐藏基类方法。

1. c#的五种类型
2. 数据类型装换的运算符是什么

答：as，is

1. 虚函数的用法

答：

1. virtual指明一成员函数为虚函数,而virtual仅用于类的定义里,在类外可不加此关键字.
2. 一个类的成员函数被定义为虚函数0时,子类该函数仍保持虚函数特征.
3. 子类覆盖此函数时,定义里可不加virtual关键字,但函数声明要和基类的完全一致!且此声明是必须的.
4. 不是纯虚函数时,父类的虚函数必须要实现; 而若将父类的虚函数设定为纯虚函数时,子类必需要覆盖之而且必须要实现之!
5. 委托与事件的用法

答：public delegate void handels();//返回值是void，没有参数

public event handels eventHandels;

his.eventHandels = new handels(fun);

public void fun()

{

}

面试的时候全是口述的，所以记的不全

名词解释

1. 白盒测试和黑盒测试
2. 死锁（操作系统）
3. **OSI网络结构的七层模型及其核心思想是什么**

第七层：应用层

* 定义了用于在网络中进行通信和数据传输的接口 - 用户程式；
* 提供标准服务，比如虚拟终端、文件以及任务的传输和处理；

　　第六层：表示层

* 掩盖不同系统间的数据格式的不同性；
* 指定独立结构的数据传输格式；
* 数据的编码和解码；加密和解密；压缩和解压缩

　　第五层：会话层

* 管理用户会话和对话；
* 控制用户间逻辑连接的建立和挂断；
* 报告上一层发生的错误

　　第四层：传输层

* 管理网络中端到端的信息传送；
* 通过错误纠正和流控制机制提供可靠且有序的数据包传送；
* 提供面向无连接的数据包的传送；

　　第三层：网络层

* 定义网络设备间如何传输数据；
* 根据唯一的网络设备地址路由数据包；
* 提供流和拥塞控制以防止网络资源的损耗

　　第二层：数据链路层

* 定义操作通信连接的程序；
* 封装数据包为数据帧；
* 监测和纠正数据包传输错误

　　第一层：物理层

* 定义通过网络设备发送数据的物理方式；
* 作为网络媒介和设备间的接口；
* 定义光学、电气以及机械特性。

1. 数据的实体完整性和参照完整性的区别
2. 一段程序字符串转换成整形（十六进制）（1A🡪26）
3. int \*Fun() （检查错误）

{

int i = 1;

int \*p = &I;

return p;

}

1. **在web页面中不论显示器的分辨率的大小是800×600，还是1024×768使得页面都显示在屏幕中间。**
2. **请利用正则表达式验证一个字符串是否为数字串**

**（请用javascript或csharp）。**

function RegExp(nstr){

var REstr = /^\d+$/g;

return REstr.test(nstr);

}

System.Text.RegularExpressions.Regex

reg=new ystem.Text.RegularExpressions.Regex(@"/^\d+$/g ");

System.Text.RegularExpressions.MatchCollection item=reg.Matches(str);

1. **请用HTML在客户端画出一个5行4列的表格，并用Javascript访问表格第1行的2列的内容**

<table id = tableNode>

<tr><td></td><td></td><td></td><td></td></tr>

<tr><td></td><td></td><td></td><td></td></tr>

<tr><td></td><td></td><td></td><td></td></tr>

<tr><td></td><td></td><td></td><td></td></tr>

<tr><td></td><td></td><td></td><td></td></tr>

</table>

tableNode.firstchild.firstchild.childNodes[1].firstchild

1. **请用javascript脚本实现弹出无边框窗口显示a.htm页**

<script laguage="JavaScript">

var win=window.open("PPA.htm","","fullscreen=1")

</script>

1. **运用ASP.NET的输出缓冲来存储动态页面**

@OutputCache

列<%@ OutputCache Duration="60" VaryByParam="None" %>

1. **本地计算机上有SQL 2000登录密码式 password 请用CSharp 创建与SQL的连接并把Northwind数据库中的sysusers表中的数据读出放入Table数据集。**

string stconnString = “ provider=SQLOLEDB;Data Source=localhost;Initial Catalog=Northwind;User ID=sa;Password=password;”;

System.Data.Sqlclient.SqlConnection myconn

= newSystem.Data.Sqlclient.SqlConnection(stconnString);

myconn.Open();

string stselString = “select \* from Northwind”;

System.Data.Sqlclient.SqlDataAdapter myadapter

= new System.Data.Sqlclient.SqlDataAdapter(stselString,myconn);

System.Data.Sqlclient.DataTable mydt = new System.Data.Sqlclient.DataTable();

myadapter.Fill(mydt);

1. **请用javascript或csharp设计一个算法，实现将数字转化为字符。**

**如（0－A，1－B，26－AA）。**

function getKey(ikey){

var quotient = ikey/26;

var residue = ikey%26;

var s = “”;

if(quotient>25)

s = “0”;

else

{

if(quotient>0)

s = String.formCharCode(quotient+65-1);

s = s+String.formCharCode(residue+65);

var st = s.split(‘@’)

if(st[0] != ‘’)

result = st[0];

else

result = st[1];

}

return result;

}

1. **Session的作用是什么，当不能使用Session会话时最可能出现的原因是什么？**

Session 对象存储特定用户会话所需的信息. 会话状态仅在支持 cookie 的浏览器中保留

1. **Application 和 Session 两种会话有什么不同。Application会话自身具有那些属性？**
2. **请简述一下你对WebService的了解。**
3. **编写代码防止内容被拷贝（javascript）**

<body oncontextmenu="return false" onselectstart="return false">

1. **.net的机制是什么?**

一次编译多次使用

1. **请说出ASP，ASP.NET的区别**
2. **． 说出下面几个函数的区别:**

**private void test(string str){…}**

**private void test(ref string str){…}**

**private void test(out string str){…}**

private void test(string str){…}　常规方式，传值方式

private void test(ref string str){…}　以引用方式传参数

private void test(out string str){…}　只用于从函数向外传值

1. **如果一个页面与数据库有频繁的交互，但又不能让页面刷新的太厉害，您有什么好的方法？**

中税科技

1、VBScript的数据类型？ASP有几个内置对象？分别是什么？

2、SQL注入漏洞是什么，带来什么危害，怎么防止

3、整篇页面居中有几种方法?各自的利弊

4、DIV和Table两种页面排版的优缺点

5、错误号080x

1. ASP.net 可以在哪些系统下运行
2. Aspx文件可以用一种以上的语言吗？
3. Asp.net可以有多个Form吗？
4. Dataset和ADO的记录集有什么区别 有什么特点，有什么不足之处

5. ADO和ADO.net 有什么不同？ADO.net 是ADO的后续版本吗？

* 1. Dataset,Datatable,Datareder,DataADapter有什么联系？
  2. C#支持多继承吗？假设有一个人类，有名字和年龄2个属性，用c#完成这个类,C#比vb.net的效率更高吗?
  3. 写一个函数,当参数N很大的时候,计算下列算式的结果:1-2+3-4+5-6+7-8+…n
  4. 用程序打印出哈希表
  5. 用列表输出M叉树,例如a(b(c,d)e,f(g,h,I))….(不记得了,反正是用程序实现~)
  6. asp.net的命名空间有哪些?

西安三石软件科技有限公司

1. struts的结构体系，每个结构的简介
2. jsp有几个内置对象，简要介绍它们的功能
3. xml的解析方法有几种，简要说明。

**陕西三毅科技有限公司**

1.有10个整数，选择一个效率比较高的方法进行排序，按升序排列。

2.有10个已经按升序排好序的整数，输入任意整数，在这10个数中查询，如果存在，返回位置，如果不存在，返回-1，选择一个效率比较高的方法进行查询。

3.有4个任意整数：A，B，C，D,通过+-\*/ 4种运算，使整个算式结果等于24，例如：1\*（2+2）\*6=24；

输出结果例如：“A”，“\*”，“（”，“B”，“+”，“C”，“）”，“\*”，“D”，“=24”；

**陕西网源电子科技有限公司笔试题**

1：java中是否可以一个类继承自多个类？

2：怎样表示数组及字符串的长度？

3：web中表单数据的提交有几种？有何区别？

4：说出page，request，session，application的作用范围？

5：session和cookie之间的联系？

6：什么是MVC，结合struts说说？

7：数据库连接池的优点及原理？

8：说一说struts1.2的工作流程？

9：Spring在开发中的优点？

10：用递归实现n!=n\*(n-1)\*……\*2\*1？

11：用图形说出购物网站是怎样设计的？

12：用HTML及javascript实现两个数的相加？

13：switch是否可以用在字符串中？

14：下面直接可以实例化的是——？

A 类 B 抽象类 C 接口

C方向

1.在X86系统中，如下变量在内存中的字节存放顺序。

DWORD Value1=0x8010011c

UINT Value2=12801

2.ASCII和Unicode有什么区别？汉字是如何显示的？

3.说出你最熟悉的三种排序方法，用你最熟悉的语言写出其中一种。

4.Bool flag

Floot x 与“零值”比较

Char \*p

5.是否了解MVC模式,其中M、V、C各表示什么？

6.列举出你曾经用过或者了解的几种设计模式？

7.int iVal1 = 0, iVal2 = 0;

int \* ipVal;

ipVal = &iVal1; What is the Value of ipVal = \*ipVal =

ipVal = (int\*)iVal1; What is the Value of ipVal = \*ipVal =

iVal2 = (int)&iVal1; What is the Value of ipVal = \*ipVal =

8.void SwapAB(int A, int B)

{

A=B;

B=A;

}

int main()

{

int A,B;

A=1;

B=2;

SwapAB(A,B);

return 0; // A= ?, B= ?

}

What is the value of A and B ?

What is the difference between STRUCTURE and OBJECT ORIENTED PROGRAMMING ?

What is the difference between C and C++ ?

What is an interpreter ?

NET方向

1.完成如下的程序语句

WCHAR A[MAX\_PATH]；

memset(A,NULL， )；//清空整个数组A

CString B；

Char \*Buff；

B.Format(“Embedded-Tech\0”)；

Buff＝ ； //将Buff指向CString B中保存的字符串

2. .NET框架概述

3. ADO与ADO.NET的区别

4. ASP与ASP.NET的区别

5. XML的操作

6. 接口、属性、事件

7. C#、ListView、DataGrid的操作

8. 双向链表、单向链表、B树、二杈树

9．COM服务器有几种类型？程序在使用一个COM服务时，COM服务器如何响应并提供服务？你认为在COM开发和使用操作时，要注意那些方面？

10.请描述中断的处理过程？

11..net Franwork:公共公共语言运行规范，.net类库SQL。ADO.net与XML ASP.NET与win forns。公共语言运行时，其它重要技术

（如CTS、CAS与线程技术）

12.ASP的概念；ASP.NET新特性；ASP.NET的运行原理；ASP的基本语法。

13.Web.config的配置

14.ASP.NET中的内建对象：Request、Response、Sever、appliation、session、Cookies、 viewstate

15.Global.asax文件

16.ASP.NET的控件类

17.组件的使用：文件上传，邮件发送，语言发送，运程控制，office组件编程

18.什么是Web Service（SOAP、UDDI、WSDL）

19.Web service的设置、跟踪与安全

20.ASP.NET的缓冲机制

21.分布式支持系统的实现

1. net Franwork:公共公共语言运行规范，.net类库SQL。ADO.net与XML ASP.NET与win forns。公共语言运行时，其它重要技术（如CTS、CAS与线程技术）
2. ASP的概念；ASP.NET新特性；ASP.NET的运行原理；ASP的基本语法。
3. Web.config的配置
4. ASP.NET中的内建对象：Request、Response、Sever、appliation、session、Cookies、 viewstate
5. Global.asax文件
6. ASP.NET的控件类
7. 组件的使用：文件上传，邮件发送，语言发送，运程控制，office组件编程
8. 什么是Web Service（SOAP、UDDI、WSDL）
9. Web service的设置、跟踪与安全
10. ASP.NET的缓冲机制
11. 分布式支持系统的实现
12. SQL Server存储过程，与触发器的作用，与优缺点
13. SQL语句
14. .NET框架概述
15. ADO与ADO.NET的区别
16. ASP与ASP.NET的区别
17. XML的操作
18. 接口、属性、事件
19. C#、ListView、DataGrid的操作
20. 双向链表、单向链表、B树、二杈树
21. 29、当用到SQL语句时，用sqlca.Sqlcode等于0判断是否成功？如不成功，可用sqlca.sqlerrtext获得数据库的返回的错误信息，但当我们用数据窗口更新时，（语句为：dw\_1.UPdate()）,用sqlca.sqlerrtext是得不到数据库返回的错误信息的,请写出数据窗口更新数据时,如何才能得到数据库的返回信息?  
    30、当多个数据窗口在更新时需要对事物进行控制，以确保数据的完整性。假设有两个数据窗口dw\_1，dw\_2需要同时更新，若其中一个失败则需要回滚，请编写同时更新两个数据窗口的代码  
    31、命名规范：请填写你认为规范的变量命名前缀  
         string    Interger     Long     Date  
    全名  
    共享  
    实例  
    局部  
    参数  
    32、三层架构如何完成？是哪三层？代表什么意思？  
    33、变量、存量的命名规则？  
    34、Frame work   a. ASP.net  
       b.  
                        c.  
     b与ｃ分别是什么？
22. 设<=800元税率为0。800元以上税率为5%。请计算某公司每个员工的某月纳税额和公司该月纳税总额（可用任何语言，初始数据可以放在数据组，数据集成数据中变量等自行定义）
23. 请列出你曾经使用过的操作系统、数据库、开发工具、计算工具以及使用的时间掌握程序
24. 请列出你熟悉的工具中，曾经使用过的控件（例如按扭、编辑框等），超过15分钟可以不列（不包括数据库控件）
25. 请给出以下数据库概念，可以举例子说明
    1. 4.1 完整性 主键 外键 索引
    2. 4.2 事务
    3. 4.3 第一凡范式 第二凡范式 第三凡范式
26. 员工数字段如下：部门代码（dept）、姓名（name）、年龄（age）。设姓名为主键。请用SQL（PL/SQL亦可）语句解决如下问题：
    1. 5.1 请创建表
    2. 5.2 请按部门代码和年龄排序列出所以员工
    3. 5.3 请列出所以年龄小于25岁的员工
    4. 5.4 请统计每个部门的员工数
    5. 5.5 请列出平均年龄小于25岁的部门
27. 请列出软件开发过程中各个阶段需要完成的任务（可以通过你参与的具体项目予以简要说明）
28. 如果需要一个棵目录树（不包括文件）存到数据库中请给出数据库设计和简要说明。
29. 请给出以下面向对象的概要说明，可以举例子说明
    1. 类 成员变量 方法 继承 封装 对象等
30. 什么是串行化
31. 用XML文件描述自己
32. 存储过程可以清空任意一个表数据
33. UML中视图名称，并解释其中两个视图作用
34. 设计模式的含义，写出几种设计模式的名称
35. 软件项目通常分为哪几个阶段
36. ADO.net的常用对象，分别描述一下
37. 如何部署一个ASP.NET项目
38. 中国手机号码正确表达式
39. 一个类Cstudent、一个私有变量intAge、一个私有变量strName、一个属性Age可写可读，一个属性Name（只读，不能被继承），一个根据年龄返回是否录取的方法Permit（录取年龄在18—25岁之间），请用C#式VB.net表达出这个类
40. 抽象与接口的区别？
41. Session的保存方式有哪几种？
42. 。NET编程语言与其他语言相比有哪些优势？
43. 在.net环境下，如果把一个正常运行Asp.net项目名称修改后，要进行怎样配置才能再次正常运行。
44. SQL、Access、Oracle三种数据库之间的区别？

1、自定义控件如何做？  
2、界面的布局？  
3、程序的执行过程  
4、如何理解三层架构？  
5、验证控件有哪些？具体用法？  
7、什么是ERP？ERP的演变过程？  
8、Asp和Asp.net的区别？  
9、谈谈存储过程和触发器以及他们的区别和项目的应用？  
10、ADO.net的七大对象  
11、谈谈数据完整性和C#的安全性  
12、谈谈C/S和B/S的特点  
13、变量的命名规则有哪些？  
14、ADO和ADO.NET的区别？  
15、ASP的六个内置对象，JSP中的几大对象？  
16、谈谈SQLSEVER的安全性

17、SQLSEVER有哪些函数？  
18、用户登录验证是写数据库连接之前或之后？  
19、Session和Appliction的区别？  
20、不对数据库进行操作不用session和Appliction传值，你怎样把值传到其他页面？  
21、C#和C++有什么区别？  
22、谈谈具体的项目开发流和立项  
23、function（）与BUG（）的区别？  
24、传值与传址的区别？  
25、谈谈你是怎么理解面向对象的？  
26、你对软件工程与软件管理的看法？  
27、给你两张表（原料表与成品表）（空表），查询出成品（汽车）所需要的原料。可再建表。  
28、数据窗口更新函数UPdate（）使用语法  
dwcontro1.UPdate（{accept{，reseflag}}）含义？  
当accept=True时  
当accept=false时  
当reseflag=True时  
当resefLag=faLSe时

试题一：

测试内容：

1、用MapInfo Professional新建一个包含至少2个字段的图层文件

A.tab；（5）

2、用MapInfo Professional在A.tab中添加两个折线对象Line1

和Line2，要求这两个对象的节点数都不少于5个，Line1和Line2必

须相交；（5）

3、用MapX + VB( or VC )编写一段程序，在程序中实现：

3.1、打开A.tab并显示在mapx控件中 ；（10）

3.2、创建一个新的物理图层文件B.tab，B.tab与A.tab在同一个

目录下；（15）

3.3、将Line1和Line2在交点处各自截断，生成4个新的折线对象

Line1\_1、Line1\_2、Line2\_1、Line2\_2；（35）

3.4、将截断产生的4条新线段添加到B上，并将这些线段按它们在

B上的生成顺序标上序号，要求在图上能看到标注；（30）

试题二：

测试内容：

1、用MapInfo Professional新建一个包含至少2个字段的图层文件

A.tab，其中一个字段为Name（Char 50），并在图层中创建至少10个

对象；（5）

2、用Access创建一个新的数据库A.mdb，在A.mdb中创建一个包含至少2

个字段的表T，其中一个字段为Name（Char 50），另一个字段为

Value（Numeric）；（5）

3、在T中添加至少10条记录，每条记录的Name值应与A中的Name值相对

应，每条记录的Value值可以任取0-100中的整数，但不允许重复；

（5）

4、用MapX + VB( or VC )编写一段程序，在程序中实现：

4.1、打开A.tab并显示在mapx控件中 ；（10）

4.2、通过ADO连接T并显示在Grid或List等其它控件

中；（10）

4.3、在界面上实现由用户自定义查询条件Value > xx，xx为100

以内的整数，然后由用户在地图上单击鼠标或是拉框，按用

户输入的条件及鼠标选定的范围查询图层A上所有满足条件

的对象；（实现单击+20 ，实现拉框+25）

4.4、以MessageBox的形式，显示所有符合条件的对象的地理属

性，包括对象的Name、类型（点|线|面）、坐标（如果对象

非点类型，取第一个节点坐标即可）（20）；

4.5、以MessageBox的形式，显示所有符合条件的对象的来自表T

的Name和Value值；（20）

# VISUAL BASIC SECTION

*Q1 : Grade yourself on your Visual Basic Skill ?*

*1 – poor …….. 10- Advance player*

Ans:

*Q2 : Declare Function Multiply\_Add Lib “test” (ByRef A as integer ,VeRef B as integer , ByVal c as integer ) as integer*

*Q2.1: Explain what is Declare Function used for ?*

Ans :

*Q2.2 : Explain the difference between ByRef and ByVal ?*

Ans :

Q3 : Is Visual Basic a COMPILER or an INTERPERTER ?

Ans

# C++ SECTION

*Q4 :*

int iVal1 = 0, iVal2 = 0;

int \* ipVal;

ipVal = &iVal1; What is the Value of ipVal = \*ipVal =

ipVal = (int\*)iVal1; What is the Value of ipVal = \*ipVal =

iVal2 = (int)&iVal1; What is the Value of ipVal = \*ipVal =

*Q5:*

void SwapAB(int A, int B)

{

A=B;

B=A;

}

int main()

{

int A,B;

A=1;

B=2;

SwapAB(A,B);

return 0; // A= ?, B= ?

}

*What is the value of A and B ?*

What is the difference between STRUCTURE and OBJECT ORIENTED PROGRAMMING ?

*What is the difference between C and C++ ?*

*What is an interpreter ?*

.

# Q6. Step through the following C code and answer the questions

void FillMem(unsigned char \*pData, unsigned char len)

{

while (len--) \*pData++ = len;

}

void main()

{

unsigned char buffer1[8];

unsigned char buffer2[8];

unsigned char \* pData;

unsigned char cnt1, cnt2;

for (cnt1 = 0; cnt1 < 8; cnt1++)

buffer1[cnt1] = 0xFF;

for (cnt1 = 0; cnt1 < 8; cnt1++)

buffer2[cnt1] = 0xFF;

for (cnt1 = 0, cnt2 = 4; cnt1 < 4; cnt1++, cnt2--)

buffer1[cnt1] = cnt1;

buffer2[cnt2] = cnt2;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Q What is the content of buffer1 & buffer2 ?

buffer1[0] = ?? buffer2[0] = ??

buffer1[1] = ?? buffer2[1] = ??

buffer1[2] = ?? buffer2[2] = ??

buffer1[3] = ?? buffer2[3] = ??

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

for (cnt1 = 0; cnt1 < 8; cnt1++)

buffer1[cnt1] = 0xFF;

FillMem(&buffer1[0], 4);

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Q. What is the content of buffer1 ?

buffer1[0] = ??

buffer1[1] = ??

buffer1[2] = ??

buffer1[3] = ??

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

for (cnt1 = 0; cnt1 < 8; cnt1++)

buffer1[cnt1] = 0xFF;

pData = &buffer1[4];

pData = pData - 2;

for (cnt1 = 0; cnt1 < 2; cnt1++)

\*pData++ = cnt1;

/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Q. What is the content of buffer1 ?

buffer1[0] = ??

buffer1[1] = ??

buffer1[2] = ??

buffer1[3] = ??

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

}

# Visual Basic / C++ Section:

You can use C/C++ or Visual Basic to do the following

Please write a simple function to access the momory card 8x8 bytes

The input parameter for memory location is between 0-63

If user want to write ‘1’ at the memory location 9, then this function will write ‘1’ at cell (1,1)

Eg.

MemAccess(9,1); // MemAccess(int Position, int Value);

MemAccess(2,5);

Result

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 |
| 0 |  |  |  |  |  |  |  |  |
| 1 |  |  |  |  |  |  |  |  |
| 2 |  |  |  |  |  |  |  |  |
| 3 |  |  |  |  |  |  |  |  |
| 4 |  |  |  |  |  |  |  |  |
| 5 |  |  |  |  |  |  |  |  |
| 6 |  |  |  |  |  |  |  |  |
| 7 |  |  |  |  |  |  |  |  |

Note: you can use array type to define this memory card.

Eg.

*Int MemoryCard[8][8]; // Array definition in C*

{

}

# BACK TO FOUNDATION

1 BYTE = \_\_\_\_\_\_\_\_\_BITS (based on 8 bits processor)

1 INTEGER = \_\_\_\_\_\_\_BYTES (based on 8 bits processor)

1 LONG = \_\_\_\_\_\_\_\_\_BYTES (based on 8 bits processor)

# 8051 ASSEMBLY LANGUAGE PROGRAMMING

*Add a comment on each line !*

MOV DPTR,#atable

MOV A,R7

MOVC A, @A+DPTR

CJNE A,#EEPDELIVERYSTATE,\_JUMP1

RET

\_JUMP1:

MOV R6,A

MOV A,R7

INC A

MOVC A, @A+DPTR

MOV DPL,A

MOV DPH,R6

POP ACC

POP ACC

CLR A

JMP @A+DPTR

*Explain what is the function ?*

**JAVA SECTION:**

Q 1. Construct a Java program to do the following

**(if you do this question, you do not have to answer Q2, Q3 and Q4):**

byte b =

byte pos = ;

bu.IsBitOn(b, pos);

b = 4;

bu.IsBitOn(b, pos);

Result:

> Byte = 5, Bit at position 0 is on.

> Byte = 4, Bit at position 0 is off

Eg.

5 = 00000101

7 6 5 4 3 2 1 0

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 |

Bit at position 0 is on

Q 2. Explain why Java program can be easily run on different platforms/operating systems.

Q 3. Describe the structure of a Class in Java

Q 4. Explain *Inheritance* and *Overloaded Function* in Java

面试考题集绵（技术类）

1、server与oracle区别？

2、存储过程优缺点

3、存储过程中怎样调用存储过程？

4、JSP上怎样调用存储过程？

5、MVC中的问题?

6、Servlet通用方法？

7、JSP中的WEB.XML是干什么的？会不会配置？

8、Tomcat运行机制？

9、Tomcat中的servlet.xml是干什么的？会不会配置？

10、JSP运行机制？

11、XML和HTML区别？

12、你是否能谈谈数据结构中的哈稀表？

13、堆和栈的区别？

14、指令元素中的include和动作元素中的include有什么区别？

15、动作元素中的forword和response.sendRedirect的区别？

16、Model View Control(MVC)中的M、V、C在JSP中分别代表什么？你用MVC模式做过些什么项目？

17、你会weblogic 吗？

18、在MVC模试中，JSP 将数据交到Servlet ，Servlet怎样将Bean传入到另一个JSP页面。

19、JSP 将数据交到Servlet时，Servlet出错，怎样保存JSP页面上的信息

20、你用JSP做过项目没有。

21、你用JSP做过项目没有（如果做过，对方会问做过什么样的项目）

22、会UINX,LINUX操作命令吗？

23、用JSP做过项目没有

24、能用MVC写一个登陆和查看自已所购的商品（我就在纸上写了一个MVC的程序）

25、你怎样测试一个类的方法运行时间快慢（提示：我写了一个时间类专门测试另一类的方法运行时间）

1. 你会Linux吗？

27、Linux相关的一些操作命令

28、程序题目基本以C语言为主（我想公司题库主要针对大学生）

29、简述中间件

1. 有一个BEAN，需要用Senlet将其发送到JSP页，写出两种方法？
2. 写出显示模态窗口的方法？
3. 在Servlet中如何实现定时访问数据库？
4. DB2中从表中选取数据的SQL语句。
5. 如何实现在线人数统计，说明其原理。
6. JSP页中有哪些默认对象，说明其功能。
7. 在struts中，模拟写出success后转发到JSP1的配置
8. 控制台应用程序example.java如下：  
   public class example  
   {  
   public static void main(String[] args)  
   {  
   int i=0, j=9;  
   do  
   {  
   if(i++>--j) break;  
   } while(i<4);  
   System.out.println("i="+i+" and j="+j);  
   }  
   }

2、程序片断如下：  
int i = 4;  
int a[] = new int[5];  
while(i>=0)  
{  
a[i] = i;  
}  
for(int j=0; j<=a.length; j++)  
{  
System.out.println(a[j]);  
}  
请问当该程序片断被运行时，是否会出现错误？如果没有，输出结果是什么？如果有错误，请指出错误并改正。

题目：   
已知strcpy函数的原型是：   
char \* strcpy(char \* strDest,const char \* strSrc);   
1.不调用库函数，实现strcpy函数。   
2.解释为什么要返回char \*。

第一题：ASP中，VBScript的唯一的数据类型是什么？  
  
第二题：在ASP中，VBScript有多种控制程序流程语句，如If…Then, Select… Case,  
For … Next, Do … Loop, Exit等语句。请为这五个语句分别写一段使用的代码。  
  
第三题：请看如下代码  
<%  
TestString="Test"  
TestA  
TestB  
Response.write TestString  
  
Sub TestA()  
TestString="TestA"  
End Sub  
  
Sub TestB()  
Dim TestString  
TestString="TestB"  
End Sub  
%>  
这段代码执行后，运行结果是什么？并解释一下为什么？  
  
第四题：在ASP中，Server中有一个方法是URLEncode(string)  
如: response.write Server.URLEncode("Test.ASP?TestNum=100&TestStr=你好")  
结果输出: Test%2EASP%3FTestNum%3D100%26TestStr%3D%C4%E3%BA%C3  
在ASP中,有ASC(String),Hex(Number),Mid(String,start,[,length])这三个可能用  
到的函数，如果是三个函数的用法  
如:  
ASC("A")=65,ASC("你")= -15133  
Hex(65)="41",Hex(-15133)="C4E3"  
Mid("hello",2,1)="e", mid("this is test!",9,2)="te"  
  
现在要求编写编码函数Function TestEncode(SourceString),及一个解码函数  
Function TestDecode(CodeString)。TestEncode(SourceString)是将SourceString  
串中非字母且非汉字且非数字的字符转换为对应Ansi编码的十六进制编码！  
如:  
TestEncode("Test.ASP?TestNum=100&TestStr=你好")=  
"Test%2EASP%3FTestNum%3D100%26TestStr%3D你好"  
而TestDecode(CodeString)是将编码的串还原，是TestEncode的逆函数。  
  
第五题：  
编写一个星期的函数GetWeek(aDate)  
返回"星期一、星期二、星期三..."  
  
第六题：  
用ASP输出九九乘法口决表的网页  
输出如下：  
1\*1=1  
1\*2=2 2\*2=4  
1\*3=3 2\*3=6 3\*3=9  
...  
要求编写一个完整的ASP文件  
  
第七题到第九题  
已知SQL Server数据库的有一个数据库TestDB，学生表结构如下：  
表名:Student  
字段名 类型 说明  
id int 自增1  
name varchar(16)  
sex char(1) 'F'表示女性,'M'表示男性  
... ...  
  
已知已经定义了ADODB.Connection对象ConnTestDB已连接了上述的TestDB数据库  
可以在以后的测试题中直接引用该对象.  
  
第七题：  
编写ASP代码,将Student中的人的姓名及性别列出来,并给统计学生人数如下:  
姓名 性别  
张三 男  
李四 男  
王五 女  
... ...  
总共有100个学生  
  
第八题:  
在上述数据库中,有一个表存放学生的得分的，结构如下：  
表名：Score  
字段名 类型 说明  
StuID int 学生的ID值，关系是：Score.StuID=Student.ID  
Chinese int  
math int  
要求输出内容：  
姓名 语文 数学 总成绩  
张三 60 100 160  
...  
请编写实现上述功的ASP代码  
  
第九题：  
已知：  
某一学生：陈六，男，语文80分，数学60分，现要求编写ASP代码  
将该学的数据插入数据库中，分别插入到上述的两个表Student,Score表中。

请用你熟悉的技术（不必局限asp 和 jsp）独立完成下列测试题目，以  
便评估能力：  
现有如下数据库结构  
product 表  
----------------------------------------------------------  
-------  
productid 商品编号 int  
productname 商品名称 varchar 50  
sale 表  
----------------------------------------------------------  
----------  
saleid 报价记录编号 int  
productid 报价商品编号（与product表关联）int  
seller 报价商店名称 varchar 50  
price 价格 float  
要求用asp 和 jsp 各编写一个web程序，选出 商品编号，名称 ，最  
高和最低报价，最高和最低报价商店名称，进行列表，并进行分页（每  
页20条 记录）。

最高和最低，这是两个结果，还分页？

类myArray定义如下：  
public class myArray  
{  
static int[] a = {21,42,16,31,23,47,35};  
public void sortArray() //对数组从小到大进行排序  
{ …… }  
public int getMax() //返回最大的数组元素  
{ …… }  
public int getMin() //返回最小的数组元素  
{ …… }  
public float getAverage() //返回所有数组元素的平均值  
{ …… }  
public static void main(String[] args)  
{ myArray myarray = new myArray();  
myarray.sortArray();  
System.out.println(myarray.getMax());  
System.out.println(myarray.getMin());  
System.out.println(myarray.getAverage());  
}  
在main方法中实现了数组元素最大值、最小值和平均值的输出。请编程实现sortArray()、getMax()、getMin()、getAverage()四个方法。  
提示：排序完成以后，最大值即为最后一个数组元素，最小值为第一个数组元素。

请在jsp网页中利用脚本计算1\*2\*3\*……\*10的值并输出。

5、在Web服务器端有一个表单文件如下：  
<html>  
<head><title>check</title></head>  
<body>  
<form action=”check.jsp” method=”post”>  
Please input your name and password:<br>  
username: <input type=”text” name=”username”><br>  
password: <input type=”password” name=”usrpwd”><br>  
<input type=”submit” name=”confirm” value=”confirm”>  
</form>  
</body>  
</html>  
现在需要在check.jsp文件中实现如下功能：对用户输入的用户名和密码进行检查，如果合法，则在网页中直接输出字符串”login success”;如果不合法，则直接在网页中输出字符串”login failure”；假定该网站的用户信息被保存在数组s[]={”jack”,”jack2006”}中，s[0]为用户名，s[1]为用户密码。请问check.jsp应该如何利用脚本实现上述功能，写出完整的程序代码。不考虑用户输入中文字符的情况。  
提示：可调用字符串自带的equals()方法进行合法的用户名和密码与用户输入参数的比较。

1、写出以下函数的功能简述

Len                           Trim

Now                          Replace

2、写出ASP中常用的5大对象并作简单的解释

3、Response对象的\_\_\_\_\_\_\_\_\_\_\_\_\_方法可以将虚拟路径转化为物理路径。

4、<%Session.abandon%>意思为:\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

5、 在ASP中，VBScript有多种控制程序流程语句，如If…Then, Select… Case,  
    For … Next, Do … Loop, Exit等语句。请为这五个语句分别写一段使用的代码。

二、程序分析

6、用ASP输出九九乘法口决表的网页  
输出如下：  
1\*1=1  
1\*2=2 2\*2=4  
1\*3=3 2\*3=6 3\*3=9  
请写出完整的ASP代码程序。

7、编写一个通用的错误信息提示模块或者数据操作模块（连接数据库和关闭数据库），并注明其调用方法及相关参数说明。  
[数据库操作模块相关提示：server.createobject("ADODB.Connection")  
server.createobject("ADODB.Recordset")   
Driver={SQL Server}; server=(local)  database=fjw  uid=sa   pwd=sa    
以上仅以MSSQL为例提示]

三、数据库相关

News表  (新闻内容表)  
Id   ClassID  AddName           title                datatime  
1      3       admin      湖南有了新的变化         2006-9-15  
2      4       李先       长沙也有了新的变化       2005-6-9  
…………  
.  
Newsclass表 (新闻栏目表)  
ClassID        ClassName  
2            国外要闻  
3            国内要闻  
4            本地要闻

8、写一个SQL语句查询显示出新闻的栏目与标题及添加者按时间排序

9、设计一个简单的存储过程从上两表中查询，返回ID、添加者、栏目名、新闻标题、添加时间。该存储过程不使用任何参数。

题目：上列需求中需要多少数据表能够完成基本设计？并简单画出设计。提示：表与表的关系。  
  
Asp&ADO  
1．写出您最常用的ASP内置对象（个数不限）和最常用的ADO对象。提示：接受参数等。  
  
2．<%=rs.fields.item(“Name”)%> 是什么意思？  
  
1．面向对象的思想包括？在ASP中如何简单实现面对对象？

2．如何看待分层？ASP如何实现简单分层？  
  
  
1．利用DreamWeaver建立简单的动态站点。实现功能自定。主要是对数据库的操作。  
2．利用作图软件（FireWorks）或CSS制作圆角表格（表格四角为圆润）。  
(网上资料的查找能力)

1.       下面哪种系统不属于网络操作系统: (D)

A. UNIX     B.  Linux    C windows NT或sever     D OS X

2.中国的地域性顶级域名是: (C)

A. an     B.  bn    C cn     D dn

3.HTML中超连接的一对标记是: A

A. <a></a>     B. <b></b>    C <i></i>   D <p></p>

4.下面哪种不属于VBscript程序的流程 (C)

A. 顺序     B. 选择    C 交叉  D 循环

5.Response.Write方法的作用是(A)

A 向浏览器输出动态数据

B 将用户信息提交到服务器

C 查询服务器中的数据

D 更新服务器数据

6.下面对在网站建设中Cookies的解释正确的是(B)

A 一种意大利小甜饼

B 保存用户信息的一个小型文本文件

C 保存在服务器上的用户信息文档

D 保存在用户上的小程序

7. 请问下面程序段执行完毕，页面上显示内容是什么：(C)

<%

Response.Write “<a href=''http://www.sina.com.cn''>新浪</a>”

%>

A. 新浪              B. <a href=''http://www.sina.com.cn''>新浪</a>

C. 新浪（超链接）    D. 该句有错，无法正常输出

8 请问下面程序段执行完毕，页面上显示的内容是什么？(A)

<%

Response.Write “春秋”

Response.End

Response.Write “战国”

%>

A. 春秋          B. 战国

C. 春秋战国      D. 春秋（换行）战国

9.Application对象的默认有效期为多少分钟？(D)

A. 10         B. 15        C. 20          D. 应用程序从启动到结束

10. 下列Response对象的用法错误的是（A）

A  <% Response.Write 输出到浏览器信息 %>

B  <% = "输出到浏览器信息" %>

C  <% Response.End %>

D  以上全都正确

\* 面向对象的语言具有\_\_\_\_\_性，\_\_\_\_\_\_性，\_\_\_\_\_\_性。

\* ADO.NET中的对象主要有\_\_\_\_\_\_\_\_、\_\_\_\_\_\_\_\_、\_\_\_\_\_\_\_\_、\_\_\_\_\_\_\_\_、\_\_\_\_\_\_\_\_

\* 填充以下方法，完成对“name长度必须大于5”的检测

private bool CheckName(string name)

{

if (\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_)

{

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_;

}

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_;

}

也可以写做：

private bool CheckName(string name)

{

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_;

}

\* .NET中的可访问修饰符有\_\_\_\_\_\_\_\_\_\_、\_\_\_\_\_\_\_\_\_\_\_、\_\_\_\_\_\_\_\_\_\_\_、\_\_\_\_\_\_\_\_\_\_\_等。

\* 当整数a赋值给一个object对象后，整数a将会被\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_。

\* 委托声明的关键字是\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_。

\* 异常捕获的语句是\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_。

\* 没有显式指定基础类型的枚举声明意味着所对应的基础类型是\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_。

\* 数组的默认下标最小值为\_\_\_\_\_\_\_。

\* 接口的类型不可以是枚举类型。

\* 接口的类型可以是结构类型。

\* 一个类可以是继承自另外一个类，一个接口也可以继承自另外一个接口。

\* C#中有一元运算符，二元运算符，三元运算符。

回答题：

\* public static const int A=1;这段代码有错误么？

\* & 和 && 有何区别？

\* 写出程序的输出结果

class Class1 {

private string str = "Class1.str";

private int i = 0;

static void StringConvert(string str) {

str = "string being converted.";

}

static void StringConvert(Class1 c) {

c.str = "string being converted.";

}

static void Add(int i) {

i++;

}

static void AddWithRef(ref int i) {

i++;

}

\* 接口的类型不可以是枚举类型。

\* 接口的类型可以是结构类型。

\* 一个类可以是继承自另外一个类，一个接口也可以继承自另外一个接口。

\* C#中有一元运算符，二元运算符，三元运算符。

回答题：

\* public static const int A=1;这段代码有错误么？

\* & 和 && 有何区别？

\* 写出程序的输出结果

class Class1 {

private string str = "Class1.str";

private int i = 0;

static void StringConvert(string str) {

str = "string being converted.";

}

static void StringConvert(Class1 c) {

c.str = "string being converted.";

}

static void Add(int i) {

i++;

}

static void AddWithRef(ref int i) {

i++;

}

1、 分数统计（15）

要求：（1）输入某班级学生的姓名、分数；

（2）对（1）的分数进行降幂排列并输出；

（3）具有输入输出界面。

2、 打字程序（15）

要求：（1）随即产生一字符串，每次产生的字符串内容、长度都不同；

（2）根据（1）的结果，输入字符串，判断输入是否正确，输出正确率；

（3）具有输入输出界面。

3、 文本编辑器（15）

要求：（1）编辑文本；

（2）保存、打开指定位置的文本文件；

（3）具有输入输出界面。

4、 加密（15）

要求：（1）输入任意一段明文M，以及密钥K;

（2）根据一下公式将其转换为密文C。

Ci = mi + K ,其中i = 0,1,……n-1 , K 为密钥；

（3）具有输入输出界面。

5、进制转换器（15）

要求：（1）可输入二进制、八进制、十进制、十六进制数；

（2）将已输入的数转换成其余进制的数；

1. 具有输入输出界面

**（经典）C语言测试：想成为嵌入式程序员应知道的0x10个基本问题- -**

C语言测试：想C语言测试：想成为嵌入式程序员应知道的0x10个基本问题  
  
  
C语言测试是招聘嵌入式系统程序员过程中必须而且有效的方法。这些年，我既参加也组织了许多这种测试，在这过程中我意识到这些测试能为带面试者和被面试者提供许多有用信息，此外，撇开面试的压力不谈，这种测试也是相当有趣的。  
从被面试者的角度来讲，你能了解许多关于出题者或监考者的情况。这个测试只是出题者为显示其对ANSI标准细节的知识而不是技术技巧而设计吗？这个愚蠢的问题吗？如要你答出某个字符的ASCII值。这些问题着重考察你的系统调用和内存分配策略方面的能力吗？这标志着出题者也许花时间在微机上而不上在嵌入式系统上。如果上述任何问题的答案是"是"的话，那么我知道我得认真考虑我是否应该去做这份工作。  
从面试者的角度来讲，一个测试也许能从多方面揭示应试者的素质：最基本的，你能了解应试者C语言的水平。不管怎么样，看一下这人如何回答他不会的问题也是满有趣。应试者是以好的直觉做出明智的选择，还是只是瞎蒙呢？当应试者在某个问题上卡住时是找借口呢，还是表现出对问题的真正的好奇心，把这看成学习的机会呢？我发现这些信息与他们的测试成绩一样有用。  
有了这些想法，我决定出一些真正针对嵌入式系统的考题，希望这些令人头痛的考题能给正在找工作的人一点帮住。这些问题都是我这些年实际碰到的。其中有些题很难，但它们应该都能给你一点启迪。  
这个测试适于不同水平的应试者，大多数初级水平的应试者的成绩会很差，经验丰富的程序员应该有很好的成绩。为了让你能自己决定某些问题的偏好，每个问题没有分配分数，如果选择这些考题为你所用，请自行按你的意思分配分数。  
预处理器（Preprocessor）  
  
1 . 用预处理指令#define 声明一个常数，用以表明1年中有多少秒（忽略闰年问题）  
  
  
#define SECONDS\_PER\_YEAR (60 \* 60 \* 24 \* 365)UL  
我在这想看到几件事情：  
&#8226;; #define 语法的基本知识（例如：不能以分号结束，括号的使用，等等）  
  
  
&#8226;; 懂得预处理器将为你计算常数表达式的值，因此，直接写出你是如何计算一年中有多少秒而不是计算出实际的值，是更清晰而没有代价的。  
&#8226;; 意识到这个表达式将使一个16位机的整型数溢出-因此要用到长整型符号L,告诉编译器这个常数是的长整型数。  
&#8226;; 如果你在你的表达式中用到UL（表示无符号长整型），那么你有了一个好的起点。记住，第一印象很重要。  
2 . 写一个"标准"宏MIN ，这个宏输入两个参数并返回较小的一个。  
  
  
#define MIN(A,B) （（A） <= (B) ? (A) : (B))  
  
这个测试是为下面的目的而设的：  
&#8226;; 标识#define在宏中应用的基本知识。这是很重要的，因为直到嵌入(inline)操作符变为标准C的一部分，宏是方便产生嵌入代码的唯一方法，对于嵌入式系统来说，为了能达到要求的性能，嵌入代码经常是必须的方法。  
&#8226;; 三重条件操作符的知识。这个操作符存在C语言中的原因是它使得编译器能产生比if-then-else更优化的代码，了解这个用法是很重要的。  
&#8226;; 懂得在宏中小心地把参数用括号括起来  
&#8226;; 我也用这个问题开始讨论宏的副作用，例如：当你写下面的代码时会发生什么事？  
  
least = MIN(\*p++, b);  
  
  
3. 预处理器标识#error的目的是什么？  
如果你不知道答案，请看参考文献1。这问题对区分一个正常的伙计和一个书呆子是很有用的。只有书呆子才会读C语言课本的附录去找出象这种问题的答案。当然如果你不是在找一个书呆子，那么应试者最好希望自己不要知道答案。  
死循环（Infinite loops）  
  
  
4. 嵌入式系统中经常要用到无限循环，你怎么样用C编写死循环呢？  
这个问题用几个解决方案。我首选的方案是：  
  
while(1)  
{  
?}  
  
  
  
一些程序员更喜欢如下方案：  
  
for(;;)  
{  
?}  
  
  
  
这个实现方式让我为难，因为这个语法没有确切表达到底怎么回事。如果一个应试者给出这个作为方案，我将用这个作为一个机会去探究他们这样做的基本原理。如果他们的基本答案是："我被教着这样做，但从没有想到过为什么。"这会给我留下一个坏印象。  
第三个方案是用 goto  
  
Loop:  
...  
goto Loop;  
  
  
应试者如给出上面的方案，这说明或者他是一个汇编语言程序员（这也许是好事）或者他是一个想进入新领域的BASIC/FORTRAN程序员。  
  
数据声明（Data declarations）  
  
5. 用变量a给出下面的定义  
a) 一个整型数（An integer）  
b)一个指向整型数的指针（ A pointer to an integer）  
c)一个指向指针的的指针，它指向的指针是指向一个整型数（ A pointer to a pointer to an intege）r  
d)一个有10个整型数的数组（ An array of 10 integers）  
e) 一个有10个指针的数组，该指针是指向一个整型数的。（An array of 10 pointers to integers）  
f) 一个指向有10个整型数数组的指针（ A pointer to an array of 10 integers）  
g) 一个指向函数的指针，该函数有一个整型参数并返回一个整型数（A pointer to a function that takes an integer as an argument and returns an integer）  
h)一个有10个指针的数组，该指针指向一个函数，该函数有一个整型参数并返回一个整型数（ An array of ten pointers to functions that take an integer argument and return an integer ）  
  
答案是：  
a) int a; // An integer  
b) int \*a; // A pointer to an integer  
c) int \*\*a; // A pointer to a pointer to an integer  
d) int a[10]; // An array of 10 integers  
e) int \*a[10]; // An array of 10 pointers to integers  
f) int (\*a)[10]; // A pointer to an array of 10 integers  
g) int (\*a)(int); // A pointer to a function a that takes an integer argument and returns an integer  
h) int (\*a[10])(int); // An array of 10 pointers to functions that take an integer argument and return an integer  
人们经常声称这里有几个问题是那种要翻一下书才能回答的问题，我同意这种说法。当我写这篇文章时，为了确定语法的正确性，我的确查了一下书。但是当我被面试的时候，我期望被问到这个问题（或者相近的问题）。因为在被面试的这段时间里，我确定我知道这个问题的答案。应试者如果不知道所有的答案（或至少大部分答案），那么也就没有为这次面试做准备，如果该面试者没有为这次面试做准备，那么他又能为什么出准备呢？  
Static  
6. 关键字static的作用是什么？  
这个简单的问题很少有人能回答完全。在C语言中，关键字static有三个明显的作用：  
&#8226;; 在函数体，一个被声明为静态的变量在这一函数被调用过程中维持其值不变。  
&#8226;; 在模块内（但在函数体外），一个被声明为静态的变量可以被模块内所用函数访问，但不能被模块外其它函数访问。它是一个本地的全局变量。  
&#8226;; 在模块内，一个被声明为静态的函数只可被这一模块内的其它函数调用。那就是，这个函数被限制在声明它的模块的本地范围内使用。  
大多数应试者能正确回答第一部分，一部分能正确回答第二部分，同是很少的人能懂得第三部分。这是一个应试者的严重的缺点，因为他显然不懂得本地化数据和代码范围的好处和重要性。  
  
  
Const  
  
7．关键字const有什么含意？  
我只要一听到被面试者说："const意味着常数"，我就知道我正在和一个业余者打交道。去年Dan Saks已经在他的文章里完全概括了const的所有用法，因此ESP(译者：Embedded Systems Programming)的每一位读者应该非常熟悉const能做什么和不能做什么.如果你从没有读到那篇文章，只要能说出const意味着"只读"就可以了。尽管这个答案不是完全的答案，但我接受它作为一个正确的答案。（如果你想知道更详细的答案，仔细读一下Saks的文章吧。）  
如果应试者能正确回答这个问题，我将问他一个附加的问题：  
下面的声明都是什么意思？  
  
const int a;  
int const a;  
const int \*a;  
int \* const a;  
int const \* a const;  
  
/\*\*\*\*\*\*/  
前两个的作用是一样，a是一个常整型数。第三个意味着a是一个指向常整型数的指针（也就是，整型数是不可修改的，但指针可以）。第四个意思a是一个指向整型数的常指针（也就是说，指针指向的整型数是可以修改的，但指针是不可修改的）。最后一个意味着a是一个指向常整型数的常指针（也就是说，指针指向的整型数是不可修改的，同时指针也是不可修改的）。如果应试者能正确回答这些问题，那么他就给我留下了一个好印象。顺带提一句，也许你可能会问，即使不用关键字const，也还是能很容易写出功能正确的程序，那么我为什么还要如此看重关键字const呢？我也如下的几下理由：  
&#8226;; 关键字const的作用是为给读你代码的人传达非常有用的信息，实际上，声明一个参数为常量是为了告诉了用户这个参数的应用目的。如果你曾花很多时间清理其它人留下的垃圾，你就会很快学会感谢这点多余的信息。（当然，懂得用const的程序员很少会留下的垃圾让别人来清理的。）  
&#8226;; 通过给优化器一些附加的信息，使用关键字const也许能产生更紧凑的代码。  
&#8226;; 合理地使用关键字const可以使编译器很自然地保护那些不希望被改变的参数，防止其被无意的代码修改。简而言之，这样可以减少bug的出现。  
Volatile  
  
8. 关键字volatile有什么含意?并给出三个不同的例子。  
一个定义为volatile的变量是说这变量可能会被意想不到地改变，这样，编译器就不会去假设这个变量的值了。精确地说就是，优化器在用到这个变量时必须每次都小心地重新读取这个变量的值，而不是使用保存在寄存器里的备份。下面是volatile变量的几个例子：  
&#8226;; 并行设备的硬件寄存器（如：状态寄存器）  
&#8226;; 一个中断服务子程序中会访问到的非自动变量(Non-automatic variables)  
&#8226;; 多线程应用中被几个任务共享的变量  
  
回答不出这个问题的人是不会被雇佣的。我认为这是区分C程序员和嵌入式系统程序员的最基本的问题。搞嵌入式的家伙们经常同硬件、中断、RTOS等等打交道，所有这些都要求用到volatile变量。不懂得volatile的内容将会带来灾难。  
假设被面试者正确地回答了这是问题（嗯，怀疑是否会是这样），我将稍微深究一下，看一下这家伙是不是直正懂得volatile完全的重要性。  
&#8226;; 一个参数既可以是const还可以是volatile吗？解释为什么。  
&#8226;; 一个指针可以是volatile 吗？解释为什么。  
&#8226;; 下面的函数有什么错误：  
int square(volatile int \*ptr)  
{  
return \*ptr \* \*ptr;  
}  
  
下面是答案：  
&#8226;; 是的。一个例子是只读的状态寄存器。它是volatile因为它可能被意想不到地改变。它是const因为程序不应该试图去修改它。  
&#8226;; 是的。尽管这并不很常见。一个例子是当一个中服务子程序修该一个指向一个buffer的指针时。  
&#8226;; 这段代码有点变态。这段代码的目的是用来返指针\*ptr指向值的平方，但是，由于\*ptr指向一个volatile型参数，编译器将产生类似下面的代码：  
  
  
int square(volatile int \*ptr)  
{  
int a,b;  
a = \*ptr;  
b = \*ptr;  
return a \* b;  
}  
  
  
由于\*ptr的值可能被意想不到地该变，因此a和b可能是不同的。结果，这段代码可能返不是你所期望的平方值！正确的代码如下：  
  
long square(volatile int \*ptr)  
{  
int a;  
a = \*ptr;  
return a \* a;  
}  
  
位操作（Bit manipulation）  
  
9. 嵌入式系统总是要用户对变量或寄存器进行位操作。给定一个整型变量a，写两段代码，第一个设置a的bit 3，第二个清除a 的bit 3。在以上两个操作中，要保持其它位不变。  
对这个问题有三种基本的反应  
&#8226;; 不知道如何下手。该被面者从没做过任何嵌入式系统的工作。  
&#8226;; 用bit fields。Bit fields是被扔到C语言死角的东西，它保证你的代码在不同编译器之间是不可移植的，同时也保证了的你的代码是不可重用的。我最近不幸看到Infineon为其较复杂的通信芯片写的驱动程序，它用到了bit fields因此完全对我无用，因为我的编译器用其它的方式来实现bit fields的。从道德讲：永远不要让一个非嵌入式的家伙粘实际硬件的边。  
&#8226;; 用 #defines 和 bit masks 操作。这是一个有极高可移植性的方法，是应该被用到的方法。最佳的解决方案如下：  
  
  
#define BIT3 (0x1 << 3)  
static int a;  
  
void set\_bit3(void) {  
a |= BIT3;  
}  
void clear\_bit3(void) {  
a &= ~BIT3;  
}  
  
一些人喜欢为设置和清除值而定义一个掩码同时定义一些说明常数，这也是可以接受的。我希望看到几个要点：说明常数、|=和&=~操作。  
访问固定的内存位置（Accessing fixed memory locations）  
  
10. 嵌入式系统经常具有要求程序员去访问某特定的内存位置的特点。在某工程中，要求设置一绝对地址为0x67a9的整型变量的值为0xaa66。编译器是一个纯粹的ANSI编译器。写代码去完成这一任务。  
这一问题测试你是否知道为了访问一绝对地址把一个整型数强制转换（typecast）为一指针是合法的。这一问题的实现方式随着个人风格不同而不同。典型的类似代码如下：  
  
int \*ptr;  
ptr = (int \*)0x67a9;  
\*ptr = 0xaa55;  
  
A more obscure approach is:  
一个较晦涩的方法是：  
  
\*(int \* const)(0x67a9) = 0xaa55;  
  
即使你的品味更接近第二种方案，但我建议你在面试时使用第一种方案。  
  
中断（Interrupts）  
  
11. 中断是嵌入式系统中重要的组成部分，这导致了很多编译开发商提供一种扩展—让标准C支持中断。具代表事实是，产生了一个新的关键字\_\_interrupt。下面的代码就使用了\_\_interrupt关键字去定义了一个中断服务子程序(ISR)，请评论一下这段代码的。  
  
\_\_interrupt double compute\_area (double radius)  
{  
double area = PI \* radius \* radius;  
printf("\nArea = %f", area);  
return area;  
}  
  
这个函数有太多的错误了，以至让人不知从何说起了：  
&#8226;; ISR 不能返回一个值。如果你不懂这个，那么你不会被雇用的。  
&#8226;; ISR 不能传递参数。如果你没有看到这一点，你被雇用的机会等同第一项。  
&#8226;; 在许多的处理器/编译器中，浮点一般都是不可重入的。有些处理器/编译器需要让额处的寄存器入栈，有些处理器/编译器就是不允许在ISR中做浮点运算。此外，ISR应该是短而有效率的，在ISR中做浮点运算是不明智的。  
&#8226;; 与第三点一脉相承，printf()经常有重入和性能上的问题。如果你丢掉了第三和第四点，我不会太为难你的。不用说，如果你能得到后两点，那么你的被雇用前景越来越光明了。  
  
\*\*\*\*\*  
代码例子（Code examples）  
  
12 . 下面的代码输出是什么，为什么？  
  
void foo(void)  
{  
unsigned int a = 6;  
int b = -20;  
(a+b > 6) ? puts("> 6") : puts("<= 6");  
}  
这个问题测试你是否懂得C语言中的整数自动转换原则，我发现有些开发者懂得极少这些东西。不管如何，这无符号整型问题的答案是输出是 ">6"。原因是当表达式中存在有符号类型和无符号类型时所有的操作数都自动转换为无符号类型。 因此-20变成了一个非常大的正整数，所以该表达式计算出的结果大于6。这一点对于应当频繁用到无符号数据类型的嵌入式系统来说是丰常重要的。如果你答错了这个问题，你也就到了得不到这份工作的边缘。  
13. 评价下面的代码片断：  
  
unsigned int zero = 0;  
unsigned int compzero = 0xFFFF;  
/\*1's complement of zero \*/  
  
对于一个int型不是16位的处理器为说，上面的代码是不正确的。应编写如下：  
  
unsigned int compzero = ~0;  
  
这一问题真正能揭露出应试者是否懂得处理器字长的重要性。在我的经验里，好的嵌入式程序员非常准确地明白硬件的细节和它的局限，然而PC机程序往往把硬件作为一个无法避免的烦恼。  
到了这个阶段，应试者或者完全垂头丧气了或者信心满满志在必得。如果显然应试者不是很好，那么这个测试就在这里结束了。但如果显然应试者做得不错，那么我就扔出下面的追加问题，这些问题是比较难的，我想仅仅非常优秀的应试者能做得不错。提出这些问题，我希望更多看到应试者应付问题的方法，而不是答案。不管如何，你就当是这个娱乐吧...  
  
动态内存分配（Dynamic memory allocation）  
14. 尽管不像非嵌入式计算机那么常见，嵌入式系统还是有从堆（heap）中动态分配内存的过程的。那么嵌入式系统中，动态分配内存可能发生的问题是什么？  
这里，我期望应试者能提到内存碎片，碎片收集的问题，变量的持行时间等等。这个主题已经在ESP杂志中被广泛地讨论过了（主要是 P.J. Plauger, 他的解释远远超过我这里能提到的任何解释），所有回过头看一下这些杂志吧！让应试者进入一种虚假的安全感觉后，我拿出这么一个小节目：  
下面的代码片段的输出是什么，为什么？  
  
char \*ptr;  
if ((ptr = (char \*)malloc(0)) ==  
NULL)  
else  
puts("Got a null pointer");  
puts("Got a valid pointer");  
  
这是一个有趣的问题。最近在我的一个同事不经意把0值传给了函数malloc，得到了一个合法的指针之后，我才想到这个问题。这就是上面的代码，该代码的输出是"Got a valid pointer"。我用这个来开始讨论这样的一问题，看看被面试者是否想到库例程这样做是正确。得到正确的答案固然重要，但解决问题的方法和你做决定的基本原理更重要些。  
Typedef  
:  
15 Typedef 在C语言中频繁用以声明一个已经存在的数据类型的同义字。也可以用预处理器做类似的事。例如，思考一下下面的例子：  
  
#define dPS struct s \*  
typedef struct s \* tPS;  
  
以上两种情况的意图都是要定义dPS 和 tPS 作为一个指向结构s指针。哪种方法更好呢？（如果有的话）为什么？  
这是一个非常微妙的问题，任何人答对这个问题（正当的原因）是应当被恭喜的。答案是：typedef更好。思考下面的例子：  
  
dPS p1,p2;  
tPS p3,p4;  
  
第一个扩展为  
  
struct s \* p1, p2;  
  
.  
上面的代码定义p1为一个指向结构的指，p2为一个实际的结构，这也许不是你想要的。第二个例子正确地定义了p3 和p4 两个指针。  
  
晦涩的语法  
  
16 . C语言同意一些令人震惊的结构,下面的结构是合法的吗，如果是它做些什么？  
  
int a = 5, b = 7, c;  
c = a+++b;  
  
这个问题将做为这个测验的一个愉快的结尾。不管你相不相信，上面的例子是完全合乎语法的。问题是编译器如何处理它？水平不高的编译作者实际上会争论这个问题，根据最处理原则，编译器应当能处理尽可能所有合法的用法。因此，上面的代码被处理成：  
  
c = a++ + b;  
  
因此, 这段代码持行后a = 6, b = 7, c = 12。  
如果你知道答案，或猜出正确答案，做得好。如果你不知道答案，我也不把这个当作问题。我发现这个问题的最大好处是这是一个关于代码编写风格，代码的可读性，代码的可修改性的好的话题。  
好了，伙计们，你现在已经做完所有的测试了。这就是我出的C语言测试题，我怀着愉快的心情写完它，希望你以同样的心情读完它。如果是认为这是一个好的测试，那么尽量都用到你的找工作的过程中去吧。天知道也许过个一两年，我就不做现在的工作，也需要找一个。  
Nigel Jones 是一个顾问，现在住在Maryland，当他不在水下时，你能在多个范围的嵌入项目中找到他。 他很高兴能收到读者的来信，他的email地址是: NAJones@compuserve.com 。  
References  
&#8226;; Jones, Nigel, "In Praise of the #error directive," Embedded Systems Programming, September 1999, p. 114.  
&#8226;; Jones, Nigel, " Efficient C Code for Eight-bit MCUs ," Embedded Systems Programming, November 1998, p. 66.  
c语言常见笔试题总结

【1 使用宏】

1.1

#ifdef NDEBUG

#define TRACE(S) S

#else

#define TRACE(S) printf("%s;\n", #S); S

#endif

问：以上TRACE()宏的作用是什么？

1.2 #error的作用？

1.3 定义一个宏，求出给定数组中的元素的个数

#define NELEMENTS(array) ??

1.4 定义一个宏，求出给定结构中给定成员的偏移量

#define OFFSET(structure, member) ??

【2 数据声明和定义】

给定以下类型的变量a的定义式：

a) An integer

b) A pointer to an integer

c) A pointer to a pointer to an integer

d) An array of 10 integers

e) An array of 10 pointers to integers

f) A pointer to an array of 10 integers

g) A pointer to a <I>function</I> that takes an integer as an argument and returns an integer

h) An array of ten pointers to <I>function</I>s that take an integer argument and return an integer

【3 复杂类型（1）】

有如下表达式：

char (\*(\*x())[])();

请用文字描述x是什么。

【4 复杂类型（2）】

jmp\_buf的定义：

typedef struct \_jmp\_buf

{

REG\_SET reg;

int extra[3];

} jmp\_buf[1];

setjmp函数的原型：

extern int setjmp (jmp\_buf \_\_env);

问：调用setjmp时传递\_\_env的内容，还是传递指针？

【5 头文件】

问：为什么标准头文件都有类似以下的结构？

#ifndef \_\_INCvxWorksh

#define \_\_INCvxWorksh

#ifdef \_\_cplusplus

extern "C" {

#endif

/\*...\*/

#ifdef \_\_cplusplus

}

#endif

#endif /\* \_\_INCvxWorksh \*/

【6 static关键字】

请说出static关键字的3种用处：

（1）用于全局变量；

（2）用于局部变量；

（3）用于函数。

/\* file.c \*/

static int a;

int b;

static int fn()

{

static int x;

int y;

}

【7 const关键字】

7.1 const关键字的意义是什么？

7.2 解释以下的变量定义：

const int a1;

int const a2;

const int \*a3;

int \* const a4;

int const \* const a5;

【8 volatile关键字】

8.1 volatile意义？例如

volatile int \*p;

8.2 volatile能和const一起使用吗？例如

volatile const int \*p;

【9 sizeof()】

有以下定义：

char \*pmsg = "A";

char msg[] = "A";

char ch = 'A';

问：

sizeof(pmsg) = ?

sizeof(msg) = ?

sizeof(“A”) = ?

sizeof(ch) = ?

sizeof(‘A’) = ? （在C++中等于多少？）

void f(char param[100])

{

// sizeof(param) = ?

}

【10 字符串】

有以下代码

char \*pmsg = "hello, world!";

strcpy(pmsg, "hi, there.");

试评论该代码。

【11 混合运算】

有以下代码：

void foo()

{

unsigned int a = 6;

int b = -20;

(a+b > 6) ? puts("> 6") : puts(" < = 6");

}

请问调用foo()的输出？

【12 内存访问】

有以下代码：

void fn()

{

int a[100];

int \*p;

p = (int \*)((unsigned int)a + 1);

printf(“p=0x%x\n”, \*p);

}

试评论以上代码。

【13 C库函数】

请说明以下函数的意义：

void perror(const char \*\_\_s);

fdprintf(int, const char \*, ...);

isspace(), isxdigit(), strerr(), sprintf()

coon @ 23:44:01 | 阅读全文 | 评论 0 | 引用 0 | 编辑

c语言笔试题(九)

2006-09-06

Tag: C语言

1．

#include "stdio.h"

int main()

{

int a;

int \*p;

p = &a;

\*p = 0x500;

a = (int )(\*(&p));

a = (int )(&(\*p));

if(a == (int)p)

printf("equal !\n");

else

printf("not equal !\n");

}

请问本程序的输出显示是什么？

答案：输出显示为”equal!”

2．

struct {

signed int bit0:1;

signed int bit1:1;

signed int bit2:1;

signed int bit3:1;

signed int bit4:1;

signed int bit5:1;

signed int bit6:1;

signed int bit7:1;

}bits;

请问sizeof(bits)是否是正确的表达式？

请问语句bits mybits; 的定义是否正确？如果不正确，要如何修改上述的结构定义才能使该语句正确？修改后的结构定义是否会影响sizeof(bits)的正确性？如果正确则该表达式的值为多少？如果将上述的结构中int类型改为char类型，此时sizeof(bits)的大小为多少？

答案：1）是正确的表达式，因为sizeof后面的内容可以是类型，也可以是变量。

2）该语句的定义不正确，因为此时的bits为一个变量；应该这样修改结构的定义

typedef struct {

signed int bit0:1;

signed int bit1:1;

signed int bit2:1;

signed int bit3:1;

signed int bit4:1;

signed int bit5:1;

signed int bit6:1;

signed int bit7:1;

}bits;

修改后sizeof(bits)表达式依然正确，其值为4；类型改为char后其值为1，注意该值是在VC环境中的32位程序中得到的值，在不同的编译器其值有可能不同，因此在编程时不能自己假定类似结构的大小。

3．

struct bit{

unsigned int a[0]:1,a[1]:1,a[2]:1….a[7]:1;

}

请问这种写法是否正确？为什么？

答案：不正确，位域中的变量不能是数组。

4．

struct a {

int x;

char y;

struct a z;

struct a \*p;

}

请问这种定义结构正确否？ 如果有问题，问题在哪里？

答案：结构中不能对定义结构本身的非指针变量，如果编译器支持则会导致无限嵌套，因此一般编译器都会认为struct a是未定义的类型，即使提前声明也不会有任何用处。

5． 什么是可重入函数？C语言中写可重入函数，应注意的事项？

答案：可重入函数是指能够被多个线程“同时”调用的函数，并且能保证函数结果的正确性的函数。在编写可重入函数时通常要注意如下的一些问题：

尽量不要使用全局变量，静态变量，如果使用了应该注意对变量访问的互斥。通常可以根据具体的情况采用：信号量机制，关调度机制，关中断机制等方式来保证函数的可重入性。

不要调用不可重入的函数，调用了不可重入的函数会使该函数也变为不可重入的函数。

注意对系统中的临界资源，互斥资源的访问方式，防止使函数成为不可重入的函数。

一般驱动程序都是不可重入的函数，因此在编写驱动程序时一定要注意重入的问题。

6． 简述stack frame 的含义。

答案：stack frame的中文译名为：栈框架，表示函数在栈空间的调用层次，以x86平台的函数调用为例，通常一个函数编译成汇编程序，都有如下的结构：

其中的leave指令相当于：mov ebp,esp ；pop ebp

各个函数在栈空间的映象为：

test1函数 test2函数 test3函数

因此在函数test3中，就可以根据这种栈框架的形式得到函数调用层次上的每个函数的基址指针，当前栈指针，以及函数调用点等信息。

7． printf (“%d%d\n”,++n, power(2,n)); 其中power(2,n)为实现一定功能的函数 如 2^n 。

请问这种表示方法有什么潜在的问题？

答案：编译器的不同，对++n 和power(2,n)处理的先后顺序不一样，形成二义性，造成程

序的移植性差，因此最好把++n 写在printf函数外面，以消除二义性。

printf (s);

请问这样的语句有没有问题？（s为一指向有效字符串的指针）

答案：没有%的话，可以这样表达，如果有%在s中的话，有意想不到的输出结果。

9． 两段代码共存于一个文件，编译时有选择的编译其中的一部分，请问如何实现？

答案：有两种简单的办法可以实现：

在源码中使用条件编译语句，然后在程序文件中定义宏的形式来选择需

要的编译代码。

在源码中使用条件编译语句，然后在编译命令的命令中加入宏定义命令

来实现选择编译。

10．数据结构指针传给函数，函数能访问数据单元，但不能修改实际的内容，如何实现？

答案：定义为指向常量的指针，这样指针所指的数据结构中的内容就不会被改变。如：

const 类型 \*p 或 类型 const \*p

11． 在头文件中定义静态变量，可能产生什么问题？

答案：在使用了该头文件的每个c程序文件中都单独存在一个该静态变量，这样造成空间的浪费并且很容易引起错误。因此建议不要在头文件中定义任何变量。

12．malloc()与 calloc()的区别？

答案：

1）参数上的区别

malloc (size\_t size);

calloc (size\_t n , size\_t size);

malloc分配一块size大小的内存块，而calloc分配一个n\*size大小的内存块

2）返回内存块的状态不同

malloc分配的内存块没有被清零，而calloc分配的内存块是清了零的。但是建议在使用内存时，如果需要初始化，则最好自己按照需要来进行初试化，不要依赖函数的实现说明。

13．寄存器变量可不可以访问其地址？可否是全局变量？在什么场合使用寄存器变量？

答案：这些问题都与编译器的实现有关，建议不要声明全局变量为寄存器变量，即使是局部变量都最好不要声明其为寄存器变量，现在的编译器在优化时都会较为合理的安排寄存器变量的使用，而人为的安排有时会造成优化的低效。

14．"\n" '\n' 的区别？

答案：前者是一个字符串并且以’/0’结束，而后者只是一个简单的字符。

15．包含预定义头文件< > 和" "的区别？

答案：< >只在指定的目录里寻找被包含文件；" "先在当前目录下查找 ，再在指定目录下查找；通常<>方式用于系统的头文件，而一般用户的头文件用" "的方式。

16．strunt S\_A{

int a[10];

};

void f()

{

int i;

strunt S\_A \*s\_ptr;

for (i=0,i<10,i++)

s\_ptr -> a[i] = i;

}

请问这段代码正确否？

答案：这段代码不正确，没有对s\_ptr指针进行初始化，在编程中要注意此类低级错误的发生。

coon @ 23:37:37 | 阅读全文 | 评论 0 | 引用 0 | 编辑

c语言笔试题(八)

2006-09-06

Tag: C语言

　　#pragma pack(8)

　　struct s1{

　　short a;

　　long b;

　　};

　　struct s2{

　　char c;

　　s1 d;

　　long long e;

　　};

　　#pragma pack()

　　问

　　1.sizeof(s2) = ?

　　2.s2的s1中的a后面空了几个字节接着是b?

　　如果您知道答案请在讨论中写出，以下是部份网友的答案，供参考：

　　网友rwxybh(行云)的答案：

　　内存布局是

　　1\*\*\* 11\*\*

　　1111 \*\*\*\*

　　1111 1111

　　所以答案就是24和3

　　下面是一个测试的程序，试一试就知道了，我用的是VC2005

　　#pragma pack(8)

　　struct s1{

　　short a; // 2 BYtes

　　long b; // 4 Bytes

　　};

　　struct s2{

　　char c; // 1 Byte

　　s1 d; // 8 Bytes

　　long long e; // 8 Bytes

　　};

　　// 1\*\*\* 11\*\*

　　// 1111 \*\*\*\*

　　// 1111 1111

　　//

　　// 00 01 02 03 04 05 06 07

　　// 00 01 02 03 04 05 06 07

　　// 00 01 02 03 04 05 06 07

　　//

　　#pragma pack()

　　int main(int argc, char\* argv[])

　　{

　　s2 a;

　　char \*p = (char \*)&a;

　　for(int i=0;i<24;++i)

　　p[i] = (char)(i%8);

　　printf("%d\n",sizeof(a));

　　printf("c=0x%lx\n",a.c);

　　printf("d.a=0x%x\n",a.d.a);

　　printf("d.b=0x%x\n",a.d.b);

　　printf("e=0x%llx\n",a.e);

　　return 0;

　　}

　　结果：

　　24

　　c=0x0

　　d.a=0x504

　　d.b=0x3020100

　　e=0x706050403020100

　　网友 redleaves (ID最吊的网友)的答案和分析：

　　如果代码:

　　#pragma pack(8)

　　struct S1{

　　char a;

　　long b;

　　};

　　struct S2 {

　　char c;

　　struct S1 d;

　　long long e;

　　};

　　#pragma pack()

　　sizeof(S2)结果为24.

　　成员对齐有一个重要的条件,即每个成员分别对齐.即每个成员按自己的方式对齐.

　　也就是说上面虽然指定了按8字节对齐,但并不是所有的成员都是以8字节对齐.其对齐的规则是,每个成员按其类型的对齐参数(通常是这个类型的大小)和指定对齐参数(这里是8字节)中较小的一个对齐.并且结构的长度必须为所用过的所有对齐参数的整数倍,不够就补空字节.

　　S1中,成员a是1字节默认按1字节对齐,指定对齐参数为8,这两个值中取1,a按1字节对齐;成员b是4个字节,默认是按4字节对齐,这时就按4字节对齐,所以sizeof(S1)应该为8;

　　S2中,c和S1中的a一样,按1字节对齐,而d 是个结构,它是8个字节,它按什么对齐呢?对于结构来说,它的默认对齐方式就是它的所有成员使用的对齐参数中最大的一个,S1的就是4.所以,成员d就是按4字节对齐.成员e是8个字节,它是默认按8字节对齐,和指定的一样,所以它对到8字节的边界上,这时,已经使用了12个字节了,所以又添加了4个字节的空,从第16个字节开始放置成员e.这时,长度为24,已经可以被8(成员e按8字节对齐)整除.这样,一共使用了24个字节.

　　a b

　　S1的内存布局：11\*\*,1111,

　　c S1.a S1.b d

　　S2的内存布局：1\*\*\*,11\*\*,1111,\*\*\*\*11111111

　　这里有三点很重要:

　　1.每个成员分别按自己的方式对齐,并能最小化长度

　　2.复杂类型(如结构)的默认对齐方式是它最长的成员的对齐方式,这样在成员是复杂类型时,可以最小化长度

　　3.对齐后的长度必须是成员中最大的对齐参数的整数倍,这样在处理数组时可以保证每一项都边界对齐

　　网友xue23(xue23) 的答案和分析：

　　有程序查一下各个变量的内存地址得知：

　　各个变量在内存中的位置为

　　c\*\*\*aa\*\*

　　bbbb\*\*\*\*

　　dddddddd

　　测试代码为：

　　s2 ss;

　　cout << "ss.c = " << &ss << endl ;

　　cout << "ss.d.a = " <<&ss.d.a << endl;

　　cout << "ss.d.b = " <<&(ss.d.b) <　　cout << "ss.d = " <<&ss.e << endl;

　　print out 各个变量的内存地址不就可以看出来了吗。

　　所以答案是24,2.

　　但是我的想像中应该是这样的分布情况：

　　c\*\*\*\*\*\*\*

　　aa\*\*bbbb

　　dddddddd

　　不知为什么会c和a放在一起，组成8位长度。

coon @ 23:35:17 | 阅读全文 | 评论 0 | 引用 0 | 编辑

c语言笔试题(七)

2006-09-06

Tag: C语言

找错

Void test1()

{

char string[10];

char\* str1="0123456789";

strcpy(string, str1);

}

Void test2()

{

char string[10], str1[10];

for(I=0; I<10;I++)

{

str1[i] ='a';

}

strcpy(string, str1);

}

Void test3(char\* str1)

{

char string[10];

if(strlen(str1)<=10)

{

strcpy(string, str1);

}

}

2. 找错

#define MAX\_SRM 256

DSN get\_SRM\_no()

{

static int SRM\_no;

int I;

for(I=0;I{

SRM\_no %= MAX\_SRM;

if(MY\_SRM.state==IDLE)

{

break;

}

}

if(I>=MAX\_SRM)

return (NULL\_SRM);

else

return SRM\_no;

}

3. 写出程序运行结果

int sum(int a)

{

auto int c=0;

static int b=3;

c+=1;

b+=2;

return(a+b+C);

}

void main()

{

int I;

int a=2;

for(I=0;I<5;I++)

{

printf("%d,", sum(a));

}

}

4.

int func(int a)

{

int b;

switch(a)

{

case 1: 30;

case 2: 20;

case 3: 16;

default: 0

}

return b;

}

则func(1)=?

5:

int a[3];

a[0]=0; a[1]=1; a[2]=2;

int \*p, \*q;

p=a;

q=&a[2];

则a[q-p]=?

6.

定义 int \*\*a[3][4], 则变量占有的内存空间为：\_\_\_\_\_

7.

编写一个函数，要求输入年月日时分秒，输出该年月日时分秒的下一秒。如输入2004年12月31日23时59分59秒，则输出2005年1月1日0时0分0秒。

coon @ 23:34:38 | 阅读全文 | 评论 0 | 引用 0 | 编辑

c语言笔试题(六)

2006-09-06

Tag: C语言

4. static有什么用途？（请至少说明两种）

1.限制变量的作用域

2.设置变量的存储域

7. 引用与指针有什么区别？

1) 引用必须被初始化，指针不必。

2) 引用初始化以后不能被改变，指针可以改变所指的对象。

2) 不存在指向空值的引用，但是存在指向空值的指针。

8. 描述实时系统的基本特性

在特定时间内完成特定的任务，实时性与可靠性

9. 全局变量和局部变量在内存中是否有区别？如果有，是什么区别？

全局变量储存在静态数据库，局部变量在堆栈

10. 什么是平衡二叉树？

左右子树都是平衡二叉树 且左右子树的深度差值的绝对值不大于1

11. 堆栈溢出一般是由什么原因导致的？

没有回收垃圾资源

12. 什么函数不能声明为虚函数？

constructor

13. 冒泡排序算法的时间复杂度是什么？

O(n^2)

14. 写出float x 与“零值”比较的if语句。

if(x>0.000001&&x<-0.000001)

16. Internet采用哪种网络协议？该协议的主要层次结构？

tcp/ip 应用层/传输层/网络层/数据链路层/物理层

17. Internet物理地址和IP地址转换采用什么协议？

ARP (Address Resolution Protocol)（地址解析協議）

18.IP地址的编码分为哪俩部分？

IP地址由两部分组成，网络号和主机号。不过是要和“子网掩码”按位与上之后才能区分哪些是网络位哪些是主机位。

2.用户输入M,N值，从1至N开始顺序循环数数，每数到M输出该数值，直至全部输出。写出C程序。

循环链表，用取余操作做

3.不能做switch()的参数类型是：

switch的参数不能为实型。

華為

1、局部变量能否和全局变量重名？

答：能，局部会屏蔽全局。要用全局变量，需要使用"::"

局部变量可以与全局变量同名，在函数内引用这个变量时，会用到同名的局部变量，而不会用到全局变量。对于有些编译器而言，在同一个函数内可以定义多个同名的局部变量，比如在两个循环体内都定义一个同名的局部变量，而那个局部变量的作用域就在那个循环体内

2、如何引用一个已经定义过的全局变量？

答：extern

可以用引用头文件的方式，也可以用extern关键字，如果用引用头文件方式来引用某个在头文件中声明的全局变理，假定你将那个变写错了，那么在编译期间会报错，如果你用extern方式引用时，假定你犯了同样的错误，那么在编译期间不会报错，而在连接期间报错

3、全局变量可不可以定义在可被多个.C文件包含的头文件中？为什么？

答：可以，在不同的C文件中以static形式来声明同名全局变量。

可以在不同的C文件中声明同名的全局变量，前提是其中只能有一个C文件中对此变量赋初值，此时连接不会出错

4、语句for( ；1 ；)有什么问题？它是什么意思？

答：和while(1)相同。

5、do……while和while……do有什么区别？

答：前一个循环一遍再判断，后一个判断以后再循环

6、请写出下列代码的输出内容

#include

main()

{

int a,b,c,d;

a=10;

b=a++;

c=++a;

d=10\*a++;

printf("b，c，d：%d，%d，%d"，b，c，d）;

return 0;

}

答：10，12，120

1、static全局变量与普通的全局变量有什么区别？static局部变量和普通局部变量有什么区别？static函数与普通函数有什么区别？

全局变量(外部变量)的说明之前再冠以static 就构成了静态的全局变量。全局变量本身就是静态存储方式，静态全局变量当然也是静态存储方式。这两者在存储方式上并无不同。这两者的区别虽在于非静态全局变量的作用域是整个源程序，当一个源程序由多个源文件组成时，非静态的全局变量在各个源文件中都是有效的。 而静态全局变量则限制了其作用域，即只在定义该变量的源文件内有效，在同一源程序的其它源文件中不能使用它。由于静态全局变量的作用域局限于一个源文件内，只能为该源文件内的函数公用，因此可以避免在其它源文件中引起错误。

从以上分析可以看出， 把局部变量改变为静态变量后是改变了它的存储方式即改变了它的生存期。把全局变量改变为静态变量后是改变了它的作用域， 限制了它的使用范围。

static函数与普通函数作用域不同。仅在本文件。只在当前源文件中使用的函数应该说明为内部函数(static)，内部函数应该在当前源文件中说明和定义。对于可在当前源文件以外使用的函数，应该在一个头文件中说明，要使用这些函数的源文件要包含这个头文件

static全局变量与普通的全局变量有什么区别：static全局变量只初使化一次，防止在其他文件单元中被引用;

static局部变量和普通局部变量有什么区别：static局部变量只被初始化一次，下一次依据上一次结果值；

static函数与普通函数有什么区别：static函数在内存中只有一份，普通函数在每个被调用中维持一份拷贝

2、程序的局部变量存在于（堆栈）中，全局变量存在于（静态区 ）中，动态申请数据存在于（ 堆）中。

3、设有以下说明和定义：

typedef union {long i; int k[5]; char c;} DATE;

struct data { int cat; DATE cow; double dog;} too;

DATE max;

则语句 printf("%d",sizeof(struct date)+sizeof(max));的执行结果是：\_\_\_52\_\_\_\_

答：DATE是一个union, 变量公用空间. 里面最大的变量类型是int[5], 占用20个字节. 所以它的大小是20

data是一个struct, 每个变量分开占用空间. 依次为int4 + DATE20 + double8 = 32.

所以结果是 20 + 32 = 52.

当然...在某些16位编辑器下, int可能是2字节,那么结果是 int2 + DATE10 + double8 = 20

4、队列和栈有什么区别？

队列先进先出，栈后进先出

5、写出下列代码的输出内容

#include

int inc(int a)

{

return(++a);

}

int multi(int\*a,int\*b,int\*c)

{

return(\*c=\*a\*\*b);

}

typedef int(FUNC1)(int in);

typedef int(FUNC2) (int\*,int\*,int\*);

void show(FUNC2 fun,int arg1, int\*arg2)

{

INCp=&inc;

int temp =p(arg1);

fun(&temp,&arg1, arg2);

printf("%d\n",\*arg2);

}

main()

{

int a;

show(multi,10,&a);

return 0;

}

答：110

7、请找出下面代码中的所以错误

说明：以下代码是把一个字符串倒序，如“abcd”倒序后变为“dcba”

1、#include"string.h"

2、main()

3、{

4、 char\*src="hello,world";

5、 char\* dest=NULL;

6、 int len=strlen(src);

7、 dest=(char\*)malloc(len);

8、 char\* d=dest;

9、 char\* s=src[len];

10、 while(len--!=0)

11、 d++=s--;

12、 printf("%s",dest);

13、 return 0;

14、}

答：

方法1：

int main(){

char\* src = "hello,world";

int len = strlen(src);

char\* dest = (char\*)malloc(len+1);//要为分配一个空间

char\* d = dest;

char\* s = &src[len-1];//指向最后一个字符

while( len-- != 0 )

\*d++=\*s--;

\*d = 0;//尾部要加

printf("%s\n",dest);

free(dest);// 使用完，应当释放空间，以免造成内存汇泄露

return 0;

}

方法2：

#include

#include

main()

{

char str[]="hello,world";

int len=strlen(str);

char t;

for(int i=0; i

{

t=str[i];

str[i]=str[len-i-1]; str[len-i-1]=t;

}

printf("%s",str);

return 0;

}

1.-1,2,7,28,,126请问28和126中间那个数是什么？为什么？

第一题的答案应该是4^3-1=63

规律是n^3-1(当n为偶数0，2，4)

n^3+1(当n为奇数1，3，5)

答案：63

2.用两个栈实现一个队列的功能？要求给出算法和思路！

设2个栈为A,B, 一开始均为空.

入队:

将新元素push入栈A;

出队:

(1)判断栈B是否为空；

(2)如果不为空，则将栈A中所有元素依次pop出并push到栈B；

(3)将栈B的栈顶元素pop出；

这样实现的队列入队和出队的平摊复杂度都还是O(1), 比上面的几种方法要好。3.在c语言库函数中将一个字符转换成整型的函数是atool()吗，这个函数的原型是什么？

函数名: atol

功 能: 把字符串转换成长整型数

用 法: long atol(const char \*nptr);

程序例:

#include

#include

int main(void)

{

long l;

char \*str = "98765432";

l = atol(lstr);

printf("string = %s integer = %ld\n", str, l);

return(0);

}

2.对于一个频繁使用的短小函数,在C语言中应用什么实现,在C++中应用什么实现?

c用宏定义，c++用inline

3.直接链接两个信令点的一组链路称作什么?

PPP点到点连接

4.接入网用的是什么接口?

5.voip都用了那些协议?

6.软件测试都有那些种类?

黑盒：针对系统功能的测试 白合：测试函数功能，各函数接口

7.确定模块的功能和模块的接口是在软件设计的那个队段完成的?

概要设计阶段

8.enum string

{

x1,

x2,

x3=10,

x4,

x5,

}x;

问x= 0x801005，0x8010f4 ;

9.unsigned char \*p1;

unsigned long \*p2;

p1=(unsigned char \*)0x801000;

p2=(unsigned long \*)0x810000;

请问p1+5= ;

p2+5= ;

三.选择题:

1.Ethternet链接到Internet用到以下那个协议?

A.HDLC;B.ARP;C.UDP;D.TCP;E.ID

2.属于网络层协议的是:

A.TCP;B.IP;C.ICMP;D.X.25

3.Windows消息调度机制是:

A.指令队列;B.指令堆栈;C.消息队列;D.消息堆栈;

4.unsigned short hash(unsigned short key)

{

return (key>>)%256

}

请问hash(16),hash(256)的值分别是:

A.1.16;B.8.32;C.4.16;D.1.32

四.找错题:

1.请问下面程序有什么错误?

int a[60][250][1000],i,j,k;

for(k=0;k<=1000;k++)

for(j=0;j<250;j++)

for(i=0;i<60;i++)

a[i][j][k]=0;

把循环语句内外换一下

2.#define Max\_CB 500

void LmiQueryCSmd(Struct MSgCB \* pmsg)

{

unsigned char ucCmdNum;

......

for(ucCmdNum=0;ucCmdNum

{

......;

}

死循环

3.以下是求一个数的平方的程序,请找出错误:

#define SQUARE(a)((a)\*(a))

int a=5;

int b;

b=SQUARE(a++);

4.typedef unsigned char BYTE

int examply\_fun(BYTE gt\_len; BYTE \*gt\_code)

{

BYTE \*gt\_buf;

gt\_buf=(BYTE \*)MALLOC(Max\_GT\_Length);

......

if(gt\_len>Max\_GT\_Length)

{

return GT\_Length\_ERROR;

}

.......

}

五.问答题:

1.IP Phone的原理是什么?

IPV6

2.TCP/IP通信建立的过程怎样，端口有什么作用？

三次握手，确定是哪个应用程序使用该协议

3.1号信令和7号信令有什么区别，我国某前广泛使用的是那一种？

4.列举5种以上的电话新业务？

微软亚洲技术中心的面试题！！！

1．进程和线程的差别。

线程是指进程内的一个执行单元,也是进程内的可调度实体.

与进程的区别:

(1)调度：线程作为调度和分配的基本单位，进程作为拥有资源的基本单位

(2)并发性：不仅进程之间可以并发执行，同一个进程的多个线程之间也可并发执行

(3)拥有资源：进程是拥有资源的一个独立单位，线程不拥有系统资源，但可以访问隶属于进程的资源.

(4)系统开销：在创建或撤消进程时，由于系统都要为之分配和回收资源，导致系统的开销明显大于创建或撤消线程时的开销。

2.测试方法

人工测试：个人复查、抽查和会审

机器测试：黑盒测试和白盒测试

2．Heap与stack的差别。

Heap是堆，stack是栈。

Stack的空间由操作系统自动分配/释放，Heap上的空间手动分配/释放。

Stack空间有限，Heap是很大的自由存储区

C中的malloc函数分配的内存空间即在堆上,C++中对应的是new操作符。

程序在编译期对变量和函数分配内存都在栈上进行,且程序运行过程中函数调用时参数的传递也在栈上进行

3．Windows下的内存是如何管理的？

4．介绍.Net和.Net的安全性。

5．客户端如何访问.Net组件实现Web Service？

6．C/C++编译器中虚表是如何完成的？

7．谈谈COM的线程模型。然后讨论进程内/外组件的差别。

8．谈谈IA32下的分页机制

小页(4K)两级分页模式，大页(4M)一级

9．给两个变量，如何找出一个带环单链表中是什么地方出现环的？

一个递增一，一个递增二，他们指向同一个接点时就是环出现的地方

10．在IA32中一共有多少种办法从用户态跳到内核态？

通过调用门，从ring3到ring0，中断从ring3到ring0，进入vm86等等

11．如果只想让程序有一个实例运行，不能运行两个。像winamp一样，只能开一个窗口，怎样实现？

用内存映射或全局原子（互斥变量）、查找窗口句柄..

FindWindow，互斥，写标志到文件或注册表,共享内存。.

12．如何截取键盘的响应，让所有的‘a’变成‘b’？

键盘钩子SetWindowsHookEx

　13．Apartment在COM中有什么用？为什么要引入？

　14．存储过程是什么？有什么用？有什么优点？

我的理解就是一堆sql的集合，可以建立非常复杂的查询，编译运行，所以运行一次后，以后再运行速度比单独执行SQL快很多

　15．Template有什么特点？什么时候用？

16．谈谈Windows DNA结构的特点和优点。

网络编程中设计并发服务器，使用多进程 与 多线程 ，请问有什么区别？

1，进程：子进程是父进程的复制品。子进程获得父进程数据空间、堆和栈的复制品。

2，线程：相对与进程而言，线程是一个更加接近与执行体的概念，它可以与同进程的其他线程共享数据，但拥有自己的栈空间，拥有独立的执行序列。

两者都可以提高程序的并发度，提高程序运行效率和响应时间。

线程和进程在使用上各有优缺点：线程执行开销小，但不利于资源管理和保护；而进程正相反。同时，线程适合于在SMP机器上运行，而进程则可以跨机器迁移。

思科

1. 用宏定义写出swap（x，y）

#define swap(x, y)\

x = x + y;\

y = x - y;\

x = x - y;

2.数组a[N]，存放了1至N-1个数，其中某个数重复一次。写一个函数，找出被重复的数字.时间复杂度必须为o（N）函数原型：

int do\_dup(int a[],int N)

3 一语句实现x是否为2的若干次幂的判断

int i = 512;

cout << boolalpha << ((i & (i - 1)) ? false : true) << endl;

4.unsigned int intvert(unsigned int x,int p,int n)实现对x的进行转换,p为起始转化位,n为需要转换的长度,假设起始点在右边.如x=0b0001 0001,p=4,n=3转换后x=0b0110 0001

unsigned int intvert(unsigned int x,int p,int n){

unsigned int \_t = 0;

unsigned int \_a = 1;

for(int i = 0; i < n; ++i){

\_t |= \_a;

\_a = \_a << 1;

}

\_t = \_t << p;

x ^= \_t;

return x;

}

慧通：

什么是预编译

何时需要预编译：

１、总是使用不经常改动的大型代码体。

２、程序由多个模块组成，所有模块都使用一组标准的包含文件和相同的编译选项。在这种情况下，可以将所有包含文件预编译为一个预编译头。

char \* const p;

char const \* p

const char \*p

上述三个有什么区别？

char \* const p; //常量指针，p的值不可以修改

char const \* p；//指向常量的指针，指向的常量值不可以改

const char \*p； //和char const \*p

char str1[] = "abc";

char str2[] = "abc";

const char str3[] = "abc";

const char str4[] = "abc";

const char \*str5 = "abc";

const char \*str6 = "abc";

char \*str7 = "abc";

char \*str8 = "abc";

cout << ( str1 == str2 ) << endl;

cout << ( str3 == str4 ) << endl;

cout << ( str5 == str6 ) << endl;

cout << ( str7 == str8 ) << endl;

结果是：0 0 1 1

解答：str1,str2,str3,str4是数组变量，它们有各自的内存空间；

而str5,str6,str7,str8是指针，它们指向相同的常量区域。

12. 以下代码中的两个sizeof用法有问题吗？[C易]

void UpperCase( char str[] ) // 将 str 中的小写字母转换成大写字母

{

for( size\_t i=0; i

if( 'a'<=str[i] && str[i]<='z' )

str[i] -= ('a'-'A' );

}

char str[] = "aBcDe";

cout << "str字符长度为: " << sizeof(str)/sizeof(str[0]) << endl;

UpperCase( str );

cout << str << endl;

答：函数内的sizeof有问题。根据语法，sizeof如用于数组，只能测出静态数组的大小，无法检测动态分配的或外部数组大小。函数外的str是一个静态定义的数组，因此其大小为6，函数内的str实际只是一个指向字符串的指针，没有任何额外的与数组相关的信息，因此sizeof作用于上只将其当指针看，一个指针为4个字节，因此返回4。

一个32位的机器,该机器的指针是多少位

指针是多少位只要看地址总线的位数就行了。80386以后的机子都是32的数据总线。所以指针的位数就是4个字节了。

main()

{

int a[5]={1,2,3,4,5};

int \*ptr=(int \*)(&a+1);

printf("%d,%d",\*(a+1),\*(ptr-1));

}

输出：2,5

\*(a+1）就是a[1]，\*(ptr-1)就是a[4],执行结果是2，5

&a+1不是首地址+1，系统会认为加一个a数组的偏移，是偏移了一个数组的大小（本例是5个int）

int \*ptr=(int \*)(&a+1);

则ptr实际是&(a[5]),也就是a+5

原因如下：

&a是数组指针，其类型为 int (\*)[5];

而指针加1要根据指针类型加上一定的值，

不同类型的指针+1之后增加的大小不同

a是长度为5的int数组指针，所以要加 5\*sizeof(int)

所以ptr实际是a[5]

但是prt与(&a+1)类型是不一样的(这点很重要)

所以prt-1只会减去sizeof(int\*)

a,&a的地址是一样的，但意思不一样，a是数组首地址，也就是a[0]的地址，&a是对象（数组）首地址，a+1是数组下一元素的地址，即a[1],&a+1是下一个对象的地址，即a[5].

1.请问以下代码有什么问题：

int main()

{

char a;

char \*str=&a;

strcpy(str,"hello");

printf(str);

return 0;

}

没有为str分配内存空间，将会发生异常

问题出在将一个字符串复制进一个字符变量指针所指地址。虽然可以正确输出结果，但因为越界进行内在读写而导致程序崩溃。

char\* s="AAA";

printf("%s",s);

s[0]='B';

printf("%s",s);

有什么错？

"AAA"是字符串常量。s是指针，指向这个字符串常量，所以声明s的时候就有问题。

cosnt char\* s="AAA";

然后又因为是常量，所以对是s[0]的赋值操作是不合法的。

1、写一个“标准”宏，这个宏输入两个参数并返回较小的一个。

.#define Min(X, Y) ((X)>(Y)?(Y):(X))//结尾没有;

2、嵌入式系统中经常要用到无限循环，你怎么用C编写死循环。

while(1){}或者for(;;)

3、关键字static的作用是什么？

定义静态变量

4、关键字const有什么含意？

表示常量不可以修改的变量。

5、关键字volatile有什么含意？并举出三个不同的例子？

提示编译器对象的值可能在编译器未监测到的情况下改变。

int (\*s[10])(int) 表示的是什么啊

int (\*s[10])(int) 函数指针数组，每个指针指向一个int func(int param)的函数。

1.有以下表达式：

int a=248; b=4;int const c=21;const int \*d=&a;

int \*const e=&b;int const \*f const =&a;

请问下列表达式哪些会被编译器禁止？为什么？

\*c=32;d=&b;\*d=43;e=34;e=&a;f=0x321f;

\*c 这是个什么东东，禁止

\*d 说了是const， 禁止

e = &a 说了是const 禁止

const \*f const =&a; 禁止

2.交换两个变量的值，不使用第三个变量。即a=3,b=5,交换之后a=5,b=3;

有两种解法, 一种用算术算法, 一种用^(异或)

a = a + b;

b = a - b;

a = a - b;

or

a = a^b;// 只能对int,char..

b = a^b;

a = a^b;

or

a ^= b ^= a;

3.c和c++中的struct有什么不同？

c和c++中struct的主要区别是c中的struct不可以含有成员函数，而c++中的struct可以。c++中struct和class的主要区别在于默认的存取权限不同，struct默认为public，而class默认为private

4.#include

#include

void getmemory(char \*p)

{

p=(char \*) malloc(100);

strcpy(p,"hello world");

}

int main( )

{

char \*str=NULL;

getmemory(str);

printf("%s/n",str);

free(str);

return 0;

}

程序崩溃，getmemory中的malloc 不能返回动态内存， free（）对str操作很危险

5.char szstr[10];

strcpy(szstr,"0123456789");

产生什么结果？为什么？

长度不一样，会造成非法的OS

6.列举几种进程的同步机制，并比较其优缺点。

原子操作

信号量机制

自旋锁

管程，会合，分布式系统

7.进程之间通信的途径

共享存储系统

消息传递系统

管道：以文件系统为基础

11.进程死锁的原因

资源竞争及进程推进顺序非法

12.死锁的4个必要条件

互斥、请求保持、不可剥夺、环路

13.死锁的处理

鸵鸟策略、预防策略、避免策略、检测与解除死锁

15. 操作系统中进程调度策略有哪几种？

FCFS(先来先服务)，优先级，时间片轮转，多级反馈

8.类的静态成员和非静态成员有何区别？

类的静态成员每个类只有一个，非静态成员每个对象一个

9.纯虚函数如何定义？使用时应注意什么？

virtual void f()=0;

是接口，子类必须要实现

10.数组和链表的区别

数组：数据顺序存储，固定大小

连表：数据可以随机存储，大小可动态改变

12.ISO的七层模型是什么？tcp/udp是属于哪一层？tcp/udp有何优缺点？

应用层

表示层

会话层

运输层

网络层

物理链路层

物理层

tcp /udp属于运输层

TCP 服务提供了数据流传输、可靠性、有效流控制、全双工操作和多路复用技术等。

与 TCP 不同， UDP 并不提供对 IP 协议的可靠机制、流控制以及错误恢复功能等。由于 UDP 比较简单， UDP 头包含很少的字节，比 TCP 负载消耗少。

tcp: 提供稳定的传输服务，有流量控制，缺点是包头大，冗余性不好

udp: 不提供稳定的服务，包头小，开销小

1：(void \*)ptr 和 (\*(void\*\*))ptr的结果是否相同？其中ptr为同一个指针

.(void \*)ptr 和 (\*(void\*\*))ptr值是相同的

2：int main()

{

int x=3;

printf("%d",x);

return 1;

}

问函数既然不会被其它函数调用，为什么要返回1？

mian中，c标准认为0表示成功，非0表示错误。具体的值是某中具体出错信息

1，要对绝对地址0x100000赋值，我们可以用

(unsigned int\*)0x100000 = 1234;

那么要是想让程序跳转到绝对地址是0x100000去执行，应该怎么做？

\*((void (\*)( ))0x100000 ) ( );

首先要将0x100000强制转换成函数指针,即:

(void (\*)())0x100000

然后再调用它:

\*((void (\*)())0x100000)();

用typedef可以看得更直观些:

typedef void(\*)() voidFuncPtr;

\*((voidFuncPtr)0x100000)();

2，已知一个数组table，用一个宏定义，求出数据的元素个数

#define NTBL

#define NTBL (sizeof(table)/sizeof(table[0]))

面试题: 线程与进程的区别和联系? 线程是否具有相同的堆栈? dll是否有独立的堆栈?

进程是死的，只是一些资源的集合，真正的程序执行都是线程来完成的，程序启动的时候操作系统就帮你创建了一个主线程。

每个线程有自己的堆栈。

DLL 中有没有独立的堆栈，这个问题不好回答，或者说这个问题本身是否有问题。因为DLL中的代码是被某些线程所执行，只有线程拥有堆栈，如果DLL中的代码是 EXE中的线程所调用，那么这个时候是不是说这个DLL没有自己独立的堆栈？如果DLL中的代码是由DLL自己创建的线程所执行，那么是不是说DLL有独立的堆栈？

以上讲的是堆栈，如果对于堆来说，每个DLL有自己的堆，所以如果是从DLL中动态分配的内存，最好是从DLL中删除，如果你从DLL中分配内存，然后在EXE中，或者另外一个DLL中删除，很有可能导致程序崩溃

unsigned short A = 10;

printf("~A = %u\n", ~A);

char c=128;

printf("c=%d\n",c);

输出多少？并分析过程

第一题，～A ＝0xfffffff5,int值 为－11，但输出的是uint。所以输出4294967285

第二题，c＝0x10,输出的是int，最高位为1，是负数，所以它的值就是0x00的补码就是128，所以输出－128。

这两道题都是在考察二进制向int或uint转换时的最高位处理。

分析下面的程序：

void GetMemory(char \*\*p,int num)

{

\*p=(char \*)malloc(num);

}

int main()

{

char \*str=NULL;

GetMemory(&str,100);

strcpy(str,"hello");

free(str);

if(str!=NULL)

{

strcpy(str,"world");

}

printf("\n str is %s",str);

getchar();

}

问输出结果是什么？希望大家能说说原因，先谢谢了

输出str is world。

free 只是释放的str指向的内存空间,它本身的值还是存在的.

所以free之后，有一个好的习惯就是将str=NULL.

此时str指向空间的内存已被回收,如果输出语句之前还存在分配空间的操作的话,这段存储空间是可能被重新分配给其他变量的,

尽管这段程序确实是存在大大的问题（上面各位已经说得很清楚了），但是通常会打印出world来。

这是因为，进程中的内存管理一般不是由操作系统完成的，而是由库函数自己完成的。

当你malloc一块内存的时候，管理库向操作系统申请一块空间（可能会比你申请的大一些），然后在这块空间中记录一些管理信息（一般是在你申请的内存前面一点），并将可用内存的地址返回。但是释放内存的时候，管理库通常都不会将内存还给操作系统，因此你是可以继续访问这块地址的，只不过。。。。。。。。楼上都说过了，最好别这么干。

char a[10],strlen(a)为什么等于15？运行的结果

#include "stdio.h"

#include "string.h"

void main()

{

char aa[10];

printf("%d",strlen(aa));

}

sizeof()和初不初始化，没有关系；

strlen()和初始化有关。

char (\*str)[20];/\*str是一个数组指针，即指向数组的指针．\*/

char \*str[20];/\*str是一个指针数组，其元素为指针型数据．\*/

long a=0x801010;

a+5=?

0x801010用二进制表示为：“1000 0000 0001 0000 0001 0000”，十进制的值为8392720，再加上5就是8392725罗

1)给定结构struct A

{

char t:4;

char k:4;

unsigned short i:8;

unsigned long m;

};问sizeof(A) = ?

给定结构struct A

{

char t:4; 4位

char k:4; 4位

unsigned short i:8; 8位

unsigned long m; // 偏移2字节保证4字节对齐

}; // 共8字节

2)下面的函数实现在一个数上加一个数，有什么错误？请改正。

int add\_n ( int n )

{

static int i = 100;

i += n;

return i;

}

当你第二次调用时得不到正确的结果，难道你写个函数就是为了调用一次？问题就出在 static上？

// 帮忙分析一下

#include

#include

#include

#include

#include

#include

typedef struct AA

{

int b1:5;

int b2:2;

}AA;

void main()

{

AA aa;

char cc[100];

strcpy(cc,"0123456789abcdefghijklmnopqrstuvwxyz");

memcpy(&aa,cc,sizeof(AA));

cout << aa.b1 <

cout << aa.b2 <

}

答案是 -16和１

首先sizeof(AA)的大小为4,b1和b2分别占5bit和2bit.

经过strcpy和memcpy后,aa的4个字节所存放的值是:

0,1,2,3的ASC码，即00110000,00110001,00110010,00110011

所以，最后一步：显示的是这４个字节的前５位，和之后的２位

分别为：10000,和01

因为int是有正负之分　　所以：答案是-16和１

求函数返回值，输入x=9999;

int func （ x ）

{

int countx = 0;

while ( x )

{

countx ++;

x = x&(x-1);

}

return countx;

}

结果呢？

知道了这是统计9999的二进制数值中有多少个1的函数，且有

9999＝9×1024＋512＋256＋15

9×1024中含有1的个数为2；

512中含有1的个数为1；

256中含有1的个数为1；

15中含有1的个数为4；

故共有1的个数为8，结果为8。

1000 - 1 = 0111，正好是原数取反。这就是原理。

用这种方法来求1的个数是很效率很高的。

不必去一个一个地移位。循环次数最少。

int a,b,c 请写函数实现C=a+b ,不可以改变数据类型,如将c改为long int,关键是如何处理溢出问题

bool add (int a, int b,int \*c)

{

\*c=a+b;

return (a>0 && b>0 &&(\*c<0 && b<0 &&(\*c>a || \*c>b)));

}

分析：

struct bit

{ int a:3;

int b:2;

int c:3;

};

int main()

{

bit s;

char \*c=(char\*)&s;

cout<<

\*c=0x99;

cout << s.a <<<<<

int a=-1;

printf("%x",a);

return 0;

}

输出为什么是

4

1

-1

-4

ffffffff

因为0x99在内存中表示为 100 11 001 , a = 001, b = 11, c = 100

当c为有符合数时, c = 100, 最高1为表示c为负数，负数在计算机用补码表示，所以c = -4;同理

b = -1;

当c为有符合数时, c = 100,即 c = 4,同理 b = 3

位域 ：

有些信息在存储时，并不需要占用一个完整的字节， 而只需占几个或一个二进制位。例如在存放一个开关量时，只有0和1 两种状态，用一位二进位即可。为了节省存储空间，并使处理简便，Ｃ语言又提供了一种数据结构，称为“位域”或“位段”。所谓“位域”是把一个字节中的二进位划分为几个不同的区域，并说明每个区域的位数。每个域有一个域名，允许在程序中按域名进行操作。这样就可以把几个不同的对象用一个字节的二进制位域来表示。一、位域的定义和位域变量的说明位域定义与结构定义相仿，其形式为：

struct 位域结构名

{ 位域列表 };

其中位域列表的形式为： 类型说明符 位域名：位域长度

例如：

struct bs

{

int a:8;

int b:2;

int c:6;

};

位域变量的说明与结构变量说明的方式相同。 可采用先定义后说明，同时定义说明或者直接说明这三种方式。例如：

struct bs

{

int a:8;

int b:2;

int c:6;

}data;

说明data为bs变量，共占两个字节。其中位域a占8位，位域b占2位，位域c占6位。对于位域的定义尚有以下几点说明：

1. 一个位域必须存储在同一个字节中，不能跨两个字节。如一个字节所剩空间不够存放另一位域时，应从下一单元起存放该位域。也可以有意使某位域从下一单元开始。例如：

struct bs

{

unsigned a:4

unsigned :0 /\*空域\*/

unsigned b:4 /\*从下一单元开始存放\*/

unsigned c:4

}

在这个位域定义中，a占第一字节的4位，后4位填0表示不使用，b从第二字节开始，占用4位，c占用4位。

2. 由于位域不允许跨两个字节，因此位域的长度不能大于一个字节的长度，也就是说不能超过8位二进位。

3. 位域可以无位域名，这时它只用来作填充或调整位置。无名的位域是不能使用的。例如：

struct k

{

int a:1

int :2 /\*该2位不能使用\*/

int b:3

int c:2

};

从以上分析可以看出，位域在本质上就是一种结构类型， 不过其成员是按二进位分配的。

二、位域的使用位域的使用和结构成员的使用相同，其一般形式为： 位域变量名?位域名 位域允许用各种格式输出。

main(){

struct bs

{

unsigned a:1;

unsigned b:3;

unsigned c:4;

} bit,\*pbit;

bit.a=1;

bit.b=7;

bit.c=15;

pri

改错：

#include

int main(void) {

int \*\*p;

int arr[100];

p = &arr;

return 0;

}

解答：

搞错了,是指针类型不同,

int \*\*p; //二级指针

&arr; //得到的是指向第一维为100的数组的指针

#include

int main(void) {

int \*\*p, \*q;

int arr[100];

q = arr;

p = &q;

return 0;

}

下面这个程序执行后会有什么错误或者效果:

#define MAX 255

int main()

{

unsigned char A[MAX],i;//i被定义为unsigned char

for (i=0;i<=MAX;i++)

A[i]=i;

}

解答：死循环加数组越界访问（C/C++不进行数组越界检查）

MAX=255

数组A的下标范围为:0..MAX-1,这是其一..

其二.当i循环到255时,循环内执行:

A[255]=255;

这句本身没有问题..但是返回for (i=0;i<=MAX;i++)语句时,

由于unsigned char的取值范围在(0..255),i++以后i又为0了..无限循环下去.

struct name1{

char str;

short x;

int num;

}

struct name2{

char str;

int num;

short x;

}

sizeof(struct name1)=8,sizeof(struct name2)=12

在第二个结构中，为保证num按四个字节对齐，char后必须留出3字节的空间；同时为保证整个结构的自然对齐（这里是4字节对齐），在x后还要补齐2个字节，这样就是12字节。

intel：

A.c 和B.c两个c文件中使用了两个相同名字的static变量,编译的时候会不会有问题?这两个static变量会保存到哪里（栈还是堆或者其他的）?

static的全局变量，表明这个变量仅在本模块中有意义，不会影响其他模块。

他们都放在数据区，但是编译器对他们的命名是不同的。

如果要使变量在其他模块也有意义的话，需要使用extern关键字。

struct s1

{

int i: 8;

int j: 4;

int a: 3;

double b;

};

struct s2

{

int i: 8;

int j: 4;

double b;

int a:3;

};

printf("sizeof(s1)= %d\n", sizeof(s1));

printf("sizeof(s2)= %d\n", sizeof(s2));

result: 16, 24

第一个struct s1

{

int i: 8;

int j: 4;

int a: 3;

double b;

};

理论上是这样的，首先是i在相对0的位置，占8位一个字节，然后，j就在相对一个字节的位置，由于一个位置的字节数是4位的倍数，因此不用对齐，就放在那里了，然后是a，要在3位的倍数关系的位置上，因此要移一位，在15位的位置上放下，目前总共是18位，折算过来是2字节2位的样子，由于double是8 字节的，因此要在相对0要是8个字节的位置上放下，因此从18位开始到8个字节之间的位置被忽略，直接放在8字节的位置了，因此，总共是16字节。

第二个最后会对照是不是结构体内最大数据的倍数，不是的话，会补成是最大数据的倍数

1）读文件file1.txt的内容（例如）：

12

34

56

输出到file2.txt：

56

34

12

（逆序）

2）输出和为一个给定整数的所有组合

例如n=5

5=1+4；5=2+3（相加的数不能重复）

则输出

1，4；2，3。

望高手赐教！！

第一题,注意可增长数组的应用.

#include

#include

int main(void)

{

int MAX = 10;

int \*a = (int \*)malloc(MAX \* sizeof(int));

int \*b;

FILE \*fp1;

FILE \*fp2;

fp1 = fopen("a.txt","r");

if(fp1 == NULL)

{printf("error1");

exit(-1);

}

fp2 = fopen("b.txt","w");

if(fp2 == NULL)

{printf("error2");

exit(-1);

}

int i = 0;

int j = 0;

while(fscanf(fp1,"%d",&a[i]) != EOF)

{

i++;

j++;

if(i >= MAX)

{

MAX = 2 \* MAX;

b = (int\*)realloc(a,MAX \* sizeof(int));

if(b == NULL)

{

printf("error3");

exit(-1);

}

a = b;

}

}

for(;--j >= 0;)

fprintf(fp2,"%d\n",a[j]);

fclose(fp1);

fclose(fp2);

return 0;

}

第二题.

#include

int main(void)

{

unsigned long int i,j,k;

printf("please input the number\n");

scanf("%d",&i);

if( i % 2 == 0)

j = i / 2;

else

j = i / 2 + 1;

printf("The result is \n");

for(k = 0; k < j; k++)

printf("%d = %d + %d\n",i,k,i - k);

return 0;

}

#include

void main()

{

unsigned long int a,i=1;

scanf("%d",&a);

if(a%2==0)

{

for(i=1;i

printf("%d",a,a-i);

}

else

for(i=1;i<=a/2;i++)

printf(" %d, %d",i,a-i);

}

兄弟,这样的题目若是做不出来实在是有些不应该, 给你一个递规反向输出字符串的例子,可谓是反序的经典例程.

void inverse(char \*p)

{

if( \*p = = '' )

return;

inverse( p+1 );

printf( "%c", \*p );

}

int main(int argc, char \*argv[])

{

inverse("abc");

return 0;

}

借签了楼上的“递规反向输出”

#include

void test(FILE \*fread, FILE \*fwrite)

{

char buf[1024] = {0};

if (!fgets(buf, sizeof(buf), fread))

return;

test( fread, fwrite );

fputs(buf, fwrite);

}

int main(int argc, char \*argv[])

{

FILE \*fr = NULL;

FILE \*fw = NULL;

fr = fopen("data", "rb");

fw = fopen("dataout", "wb");

test(fr, fw);

fclose(fr);

fclose(fw);

return 0;

}

在对齐为4的情况下

struct BBB

{

long num；

char \*name;

short int data;

char ha;

short ba[5];

}\*p;

p=0x1000000;

p+0x200=\_\_\_\_;

(Ulong)p+0x200=\_\_\_\_;

(char\*)p+0x200=\_\_\_\_;

希望各位达人给出答案和原因，谢谢拉

解答：假设在32位CPU上，

sizeof(long) = 4 bytes

sizeof(char \*) = 4 bytes

sizeof(short int) = sizeof(short) = 2 bytes

sizeof(char) = 1 bytes

由于是4字节对齐，

sizeof(struct BBB) = sizeof(\*p)

= 4 + 4 + 2 + 1 + 1/\*补齐\*/ + 2\*5 + 2/\*补齐\*/ = 24 bytes (经Dev-C++验证)

p=0x1000000;

p+0x200=\_\_\_\_;

= 0x1000000 + 0x200\*24

(Ulong)p+0x200=\_\_\_\_;

= 0x1000000 + 0x200

(char\*)p+0x200=\_\_\_\_;

= 0x10000

c/c++中内存的分配问题

2008/03/28 04:08 P.M.

|  |
| --- |
| 注意：本文转自网络，版权归原作者所有。  在C++中，内存分成5个区，他们分别是堆、栈、自由存储区、全局/静态存储区和常量存储区。  　　栈，就是那些由编译器在需要的时候分配，在不需要的时候自动清楚的变量的存储区。里面的变量通常是局部变量、函数参数等。  　　堆，就是那些由new分配的内存块，他们的释放编译器不去管，由我们的应用程序去控制，一般一个new就要对应一个delete。如果程序员没有释放掉，那么在程序结束后，操作系统会自动回收。  　　自由存储区，就是那些由malloc等分配的内存块，他和堆是十分相似的，不过它是用free来结束自己的生命的。  　　全局/静态存储区，全局变量和静态变量被分配到同一块内存中，在以前的C语言中，全局变量又分为初始化的和未初始化的，在C++里面没有这个区分了，他们共同占用同一块内存区。  　　常量存储区，这是一块比较特殊的存储区，他们里面存放的是常量，不允许修改（当然，你要通过非正当手段也可以修改，而且方法很多，在《const的思考》一文中，我给出了6种方法）  　　明确区分堆与栈  　　在bbs上，堆与栈的区分问题，似乎是一个永恒的话题，由此可见，初学者对此往往是混淆不清的，所以我决定拿他第一个开刀。  　　首先，我们举一个例子：  void f() { int\* p=new int[5]; }  　　这条短短的一句话就包含了堆与栈，看到new，我们首先就应该想到，我们分配了一块堆内存，那么指针p呢？他分配的是一块栈内存，所以这句话的意思就是：在栈内存中存放了一个指向一块堆内存的指针p。在程序会先确定在堆中分配内存的大小，然后调用operator new分配内存，然后返回这块内存的首地址，放入栈中，他在VC6下的汇编代码如下：  00401028 push 14h 0040102A call operator new (00401060) 0040102F add esp,4 00401032 mov dword ptr [ebp-8],eax 00401035 mov eax,dword ptr [ebp-8] 00401038 mov dword ptr [ebp-4],eax  　　这里，我们为了简单并没有释放内存，那么该怎么去释放呢？是delete p么？澳，错了，应该是delete []p，这是为了告诉编译器：我删除的是一个数组，VC6就会根据相应的Cookie信息去进行释放内存的工作。  　　好了，我们回到我们的主题：堆和栈究竟有什么区别？  　　主要的区别由以下几点：  　　1、管理方式不同；  　　2、空间大小不同；  　　3、能否产生碎片不同；  　　4、生长方向不同；  　　5、分配方式不同；  　　6、分配效率不同；  　　管理方式：对于栈来讲，是由编译器自动管理，无需我们手工控制；对于堆来说，释放工作由程序员控制，容易产生memory leak。  　　空间大小：一般来讲在32位系统下，堆内存可以达到4G的空间，从这个角度来看堆内存几乎是没有什么限制的。但是对于栈来讲，一般都是有一定的空间大小的，例如，在VC6下面，默认的栈空间大小是1M（好像是，记不清楚了）。当然，我们可以修改：  　　打开工程，依次操作菜单如下：Project->Setting->Link，在Category 中选中Output，然后在Reserve中设定堆栈的最大值和commit。  　　注意：reserve最小值为4Byte；commit是保留在虚拟内存的页文件里面，它设置的较大会使栈开辟较大的值，可能增加内存的开销和启动时间。  　　碎片问题：对于堆来讲，频繁的new/delete势必会造成内存空间的不连续，从而造成大量的碎片，使程序效率降低。对于栈来讲，则不会存在这个问题，因为栈是先进后出的队列，他们是如此的一一对应，以至于永远都不可能有一个内存块从栈中间弹出，在他弹出之前，在他上面的后进的栈内容已经被弹出，详细的可以参考数据结构，这里我们就不再一一讨论了。  　　生长方向：对于堆来讲，生长方向是向上的，也就是向着内存地址增加的方向；对于栈来讲，它的生长方向是向下的，是向着内存地址减小的方向增长。  　　分配方式：堆都是动态分配的，没有静态分配的堆。栈有2种分配方式：静态分配和动态分配。静态分配是编译器完成的，比如局部变量的分配。动态分配由alloca函数进行分配，但是栈的动态分配和堆是不同的，他的动态分配是由编译器进行释放，无需我们手工实现。  　　分配效率：栈是机器系统提供的数据结构，计算机会在底层对栈提供支持：分配专门的寄存器存放栈的地址，压栈出栈都有专门的指令执行，这就决定了栈的效率比较高。堆则是C/C++函数库提供的，它的机制是很复杂的，例如为了分配一块内存，库函数会按照一定的算法（具体的算法可以参考数据结构/操作系统）在堆内存中搜索可用的足够大小的空间，如果没有足够大小的空间（可能是由于内存碎片太多），就有可能调用系统功能去增加程序数据段的内存空间，这样就有机会分到足够大小的内存，然后进行返回。显然，堆的效率比栈要低得多。  　　从这里我们可以看到，堆和栈相比，由于大量new/delete的使用，容易造成大量的内存碎片；由于没有专门的系统支持，效率很低；由于可能引发用户态和核心态的切换，内存的申请，代价变得更加昂贵。所以栈在程序中是应用最广泛的，就算是函数的调用也利用栈去完成，函数调用过程中的参数，返回地址，EBP和局部变量都采用栈的方式存放。所以，我们推荐大家尽量用栈，而不是用堆。  　　虽然栈有如此众多的好处，但是由于和堆相比不是那么灵活，有时候分配大量的内存空间，还是用堆好一些。  　　无论是堆还是栈，都要防止越界现象的发生（除非你是故意使其越界），因为越界的结果要么是程序崩溃，要么是摧毁程序的堆、栈结构，产生以想不到的结果,就算是在你的程序运行过程中，没有发生上面的问题，你还是要小心，说不定什么时候就崩掉，那时候debug可是相当困难的：）  可以参见：<http://www.cndw.com/tech/program/2006051066011.asp> |