



关于重载和多态正确的是

正确答案: B 你的答案: 空 (错误)

如果父类和子类都有相同的方法, 参数个数不同, 将子类对象赋给父类后, 由于子类继承于父类, 所以使用父类指针

调用父类方法时, 实际调用的是子类的方法

选项全部都不正确

重载和多态在 C++ 面向对象编程中经常用到的方法, 都只在实现子类的方法时才会使用

```
class A{
    void test(float a){cout<<"1";}
};
class B:public A{
    void test(int b){cout<<"2";}
};
A *a=new A;
B *b=new B;
a=b;
a.test(1.1);
结果是 1
```

现有两堆硬币, 小明和小亮玩游戏, 每次每人只能从其中一堆中取走 1 个或 2 个硬币, 最后将硬币取完者算作胜利, 当两堆硬币的个数分别是 12, 13 时, 小明应该如何安排策略才能必定获得胜利?

正确答案: C 你的答案: 空 (错误)

安排小亮先取硬币

小明或小亮均一定必胜

安排自己先取硬币

不可能必胜

不考虑任何编译器优化(如: NRVO), 下述代码的第 10 行会发生

```
1  #include <stdio.h>//1
2  class B{//2
3      //{3
4      };//4
5      B func(const B& rhs){//5
6          return rhs;//6
7      };//7
8      int main(int argc, char **argv){//8
9          B b1, b2;//9
10         b2=func(b1);//10
11     };//11
```

正确答案: D 你的答案: 空 (错误)



一次默认构造函数, 一次拷贝构造函数, 一次析构函数, 一次 (拷贝赋值运算符) operator=  
二次拷贝构造函数, 一次析构函数  
一次 (拷贝赋值运算符) operator=, 一次析构函数  
一次拷贝构造函数, 一次析构函数, 一次 (拷贝赋值运算符) operator=

下面代码的输出结果是()

```
1  int main() {
2      int pid;
3      int num=1;
4      pid=fork();
5      if(pid>0) {
6          num++;
7          printf("in parent:num:%d addr:%x\n", num, &num);
8      }
9      else if(pid==0) {
10         printf("in child:num:%d addr:%x\n", num, &num);
11     }
12 }
```

正确答案: B 你的答案: 空 (错误)

父子进程中输出的 num 相同, num 地址不相同  
父子进程中输出的 num 不同, num 地址相同  
父子进程中输出的 num 相同, num 地址也相同  
父子进程中输出的 num 不同, num 地址不相同

```
1  int a=0;
2  class someClass{
3      int b;
4      static int c;
5  };
6  int main() {
7      int d=0;
8      someClass *p=new someClass();
9      return 0;
10 }
```

关于以上代码中的变量在内存中的存储位置描述不正确的是()

正确答案: B 你的答案: 空 (错误)

- b 存在堆区
- c 存在堆区
- d 存在栈区
- a 存在全局变量区

以下代码段有问题的是()





正确答案: A B C 你的答案: 空 (错误)

```
void func1(char *e){
    char *p1;
    p1=malloc(100);
    sprintf(p1,error:"%s'." ,e);
    local_log(p1);
}

int func2(char *filename)
{
    FILE *fp;
    int key;
    fp=fopen(filename,"r");
    fscanf(fp,"%d",&key);
    return key;
}

void func3(char *info){
    char *p,*pp;
    p=malloc(100);
    pp=p;
    free(p);
    sprintf(pp,*info:"%s'." ,info);
    free(pp);
}
```

选项全部都正确

```
1  #include<stdio.h>
2  int main()
3  {
4      unsigned char i=7;
5      int j=0;
6      for(;i>0;i-=3)
7          {
8              ++j;
9          }
10         printf("%d\n",j);
11         return 0;
12 }
```

请问该程序的输出是多少?

正确答案: C 你的答案: 空 (错误)

2  
死循环  
173



172

快速排序在下面哪种情况下优势最明显()

正确答案: C 你的答案: 空 (错误)

数据有多个相同数值

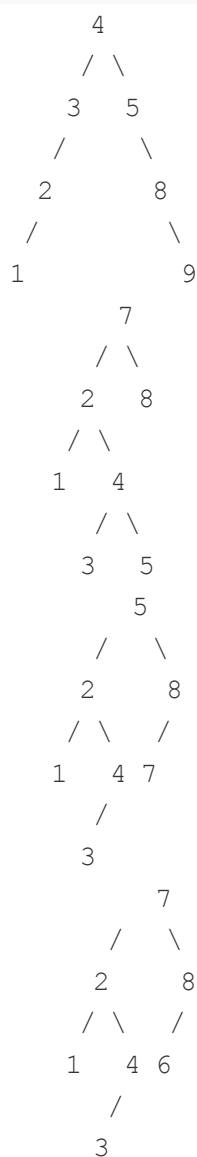
数据基本有序

数据基本无序

数据无任何相同数值

下列各树形结构中,哪些是平衡二叉查找树:

正确答案: C 你的答案: 空 (错误)



下面哪种数据结构最适合创建一个优先级队列()。

正确答案: A 你的答案: 空 (错误)





堆  
双向链表  
单向链表  
栈

某个大型的网络游戏网站,现有几亿用户,为了实时获取前十名游戏分数最高的玩家,使用以下哪个排序算法比较合理()

正确答案: D 你的答案: 空 (错误)

基数排序  
快速排序  
二叉排序  
堆排序

有 A,B,C,D,E 五个字符,出现的频率分别为 2,5,3,3,4,由 A,B,C,D,E 生成的最优二叉树中,该树的带权路径长是多少()

正确答案: C 你的答案: 空 (错误)

35  
49  
39  
45

从根开始按层次(第 0 层->第 1 层->第 2 层)遍历一颗二叉树,需要使用什么辅助数据结构?()

正确答案: B 你的答案: 空 (错误)

heap  
queue  
binary tree  
stack

假设 MySQL 数据库表:

```
create table T{
k int unsigned not null auto_increment,
a date,
b varchar(24),
c int,d varchar(24),
primary key(k),unique key a_index (a DESC,b DESC),
key k1(b),key k2(c),key k3(d));
```

如下哪些 sql 语句查询能较好的利用索引?()

正确答案: A D 你的答案: 空 (错误)

```
select b from T WHERE b like 'aaa%';
select a,b from T WHERE a='2015-10-25' ORDER BY b ASC,c ASC;
select a,b,c from T WHERE a='2015-10-25' ORDER BY b ASC;
```





```
select a,b,c from T WHERE a='2015-10-25' ORDER BY a,b;
```

在一个游戏的任务中,玩家需要进入 1 个山洞,取得宝石,之后回到入口.

山洞的地图如下:

S-----T

S 是入口

T 处有宝箱,打开宝箱之后可能得到的物品有:

- 1)宝石,出现概率为 5%.
- 2)魔法券.出现概率为 50%.玩家每消耗一个魔法券,可以直接传送到入口 S.
- 3)什么也没有,概率为 45%.

S 到 T 的距离为 1.

每次玩家回到 S 之后,宝箱 T 的状态会重置,再次进入山洞可以重新打开宝箱获得物品.

玩家的任务是到达 T 获取宝石之后回到入口 S.如果到达 T 之后没有获得宝石,可以走出山洞之后

再进入反复刷.

问题:玩家完成任务所走路程的数学期望是()

正确答案: D 你的答案: 空 (错误)

40

25

35

30

下面关于 TCP 的描述,错误的是()

正确答案: C 你的答案: 空 (错误)

TCP 是一种面向连接的协议,给用户进程提供可靠的全双工的字节流

TCP 客户端和服务端之间建立连接需要经过 3 次握手

只要有一方主动关闭连接后,这个 TCP 连接就结束了

TCP 在传输数据过程中必须保持着连接,这个连接会给通信过程增加开销

关于进程和线程描述正确的是()

正确答案: A B D 你的答案: 空 (错误)

线程不拥有系统资源,但可以访问隶属于进程的资源

在创建或销毁进程时,系统开销明显大于创建或销毁线程时开销

进程是调度和拥有资源的基本单位

不仅进程可以并发执行,同一个进程的多个线程之间也可以并发执行

下面()哪个函数返回的有效句柄用完后不需要 CloseHandle

正确答案: B 你的答案: 空 (错误)

CreateThread

GetCurrentProcess

OpenProcess

获取更多资料礼包!

微信关注: 白熊求职



CreateFile

invalidateRect 的作用是()

正确答案: D 你的答案: 空 (错误)

擦除一个矩形区域

删除一个矩形

使一个矩形区域变为无效, 从而释放部分 GDI 资源

使一个矩形区域变为无效, 从而可以重绘

典型的创建 Windows 窗口过程的流程为:()

正确答案: C 你的答案: 空 (错误)

注册窗口类->创建窗口->更新窗口->显示窗口->消息循环

创建窗口->注册窗口类->显示窗口->更新窗口->消息循环

注册窗口类->创建窗口->显示窗口->更新窗口->消息循环

创建窗口->注册窗口类->更新窗口->显示窗口->消息循环

