

1. 不用递归就不能实现二叉树的前序遍历()错
2. 队列在程序调用时必不可少,因此递归离不开队列()错
3. 只有那种使用了局部变量的递归过程在转换成非递归过程时才必须使用栈()错
4. 一个递归算法必须包括()终止条件和递归部分
5. 对  $n$  个记录的线性表进行快速排序为减少算法的递归深度,以下叙述正确的是()每次分区后,先处理较短的部分
6. 递归函数最终会结束,那么这个函数一定? 有一个分支不调用自身
7. 下列方法中, \_\_\_\_ 不可以用来程序调优? B  
改善数据访问方式以提升缓存命中率  
使用多线程的方式提高 I/O 密集型操作的效率  
利用数据库连接池替代直接的数据库访问  
利用迭代替代递归  
合并多个远程调用批量发送  
共享冗余数据提高访问效率
8. 有一段楼梯台阶有 15 级台阶,以小明的脚力一步最多只能跨 3 级,请问小明登上这段楼梯有多少种不同的走法?() 5768
9. 4 个圆盘的 Hanoi 塔,总的移动次数为()15
10. 采用递归方式对顺序表进行快速排序,下列关于递归次数的叙述中,正确的是 ( ) D  
递归次数与初始数据的排列次序无关  
每次划分后,先处理较长的分区可以减少递归次数  
每次划分后,先处理较短的分区可以减少递归次数  
递归次数与每次划分后得到的分区处理顺序无关
11. Which of the following statements are true? ACD  
we can create a binary tree from given inorder and preorder traversal sequences.  
We can create a binary tree from given preorder and postorder traversal sequences.  
For an almost sorted array, insertion sort can be more effective than Quicksort.  
Suppose  $T(n)$  is the runtime of resolving a problem with  $n$  elements,  $T(n) = \Theta(1)$  if  $n = 1$ ;  $T(n) = 2T(n/2) + \Theta(n)$  if  $n > 1$ ; so  $T(n)$  is  $\Theta(n \log n)$ .  
None of the above.

```

6 1 class program
2 {
3     static void Main(string[] args)
4     {
5         int i;
6         i = x(x(8));
7     }
8     static int x(int n)
9     {
10        if (n <= 3)
11            return 1;
12        else
13            return x(n - 2) + x(n - 4) + 1;
14    }
15 }

```

递归算法x(x(8))需要调用几次函数x(int n)?

18

8 有函数int func(int i)的实现为

```

1 int func(int i)
2 {
3     if (i > 1)
4         return i * func(i - 1);
5     else
6         return 1;
7 }
8
9

```

请问函数调用func(5)的返回值是多少 ( )

120

10 设有递归算法如下，

```

1 int x(int n)
2 {
3     if(n<=3)
4         return 1;
5     else
6         return x(n-2)+x(n-4)+1;
7 }

```

试问计算x(x(8))时需要计算 ( ) 次x函数。

18

11 以下程序是用辗转相除法来计算两个非负数之间的最大公约数：

```

1 long long gcd(long long x, long long y) {
2     if (y == 0)
3         return x;
4     else
5         return gcd(y, x % y);
6 }

```

我们假设x,y中最大的那个数的长度为n, x>y, 基本运算时间复杂度为O(1), 那么该程序的时间复杂度为 ( ) O(logy)

13 执行完下列语句段后,i值为()

```

1 int f(int x){
2     return ((x>0)?x*f(x-1):2)
3 }
4 int i;
5 i=f(f(2));

```

48

14 给定下列程序，那么执行printf("%d\n", foo(20, 13));的输出结果是\_\_\_\_\_。

```

1 int foo(int x, int y){
2     if (x <= 0 || y <= 0)
3         return 1;
4     return 3 * foo( x-6, y/2 );
5 }

```

81

20 仔细阅读以下一段递归的函数定义：

```

1 int ack(int m,int n)
2 {
3     if(m==0)
4     {
5         return n+1;
6     }
7     else if(n==0)
8     {
9         return ack(m-1,1);
10    }
11    else
12    {
13        return ack(m-1,ack(m,n-1));
14    }
15 }

```

请问ack(3,3)的返回值是 1 。

61

```

19 1 int Recurse(int a, int b)
2 {
3     if(a >= b)
4     {
5         if(a == b)
6             return a;
7         else
8             return 0;
9     }
10    else
11    {
12        return Recurse(a + 1, b - 1) + a + b;
13    }
14 }

```

假设a=8, b=2012, Recurse函数的返回值是 1

2025050

16 当n=5时，下列函数的返回值是： ( )

```

1 int foo(int n) {
2     if (n < 2)
3         return n;
4     return foo(n - 1) + foo(n - 2);
5 }

```

5