

## 有赞2019校招前端笔试

---

1、给定一个有序整数数组以及一个整数

*k*

，请问在

该数组中查找比

*k*

大的最小

元素的最优算法时间复杂度是多少？

选项

- A:  $O(\log \log n)$
- B:  $O(\log n)$
- C:  $O(n)$
- D:  $O(\log n \times \log n)$

正确答案: B

参考解析

- 二分法查找
- 找到 k 后, k 右边的就是要找的
- 二分法的时间复杂度是  $O(\log n)$
- 最后再加一, 但是这个会忽略掉

2、以下哪个排序算法对只有一两个元素乱序的数组排序性能最好?

- A. 快速排序
- B. 堆排序
- C. 归并排序
- D. 插入排序

正确答案: D

3、有 8 个完全相同的硬币, 其中只有一个硬币比其他 7 个重, 其余七个一样重, 给你一个没有刻度的天平, 请问最少需要称多少次才能找到重的那个硬币?

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

正确答案: A

一共8枚硬币, 先分成3, 3, 2. 两个3枚硬币的堆进行称重, 如果同等重量, 则再秤一次就可以找到; 如果重量不等, 那么找到重的一堆中的3个随机取两个去秤重, 3枚硬币那个重就能分辨出来了。

4、数组的 *leader* 元素是指比它右边所有元素都大的元素, 请问查找数组中所有 *leader* 元素的最优算法时间复杂度是多少?

- A.  $O(\log n)$
- B.  $O(n)$
- C.  $O(n \log n)$
- D.  $O(n^2)$

**正确答案: B**

只需遍历一边数组即可。从右向左遍历，每遍历出来一个leader，存下这个数，后续数只需与这个leader比较即可，直到有新的leader产生。

**5、给定以下哪些遍历结果可以重建出 BST?**

- A. 先序、中序、后序中的任意一个
- B. 先序或者后序中的一个
- C. 先序和中序两个
- D. 先序和后序两个

**正确答案: C D**

由于题目说的是重建出来二叉搜索树，所以前序遍历（中左右），后续遍历（左右中）都可唯一确定一个BST（可通过顺序判断左右子树），而中序遍历（左右中）排序出来是顺序排序，无法重建树。

关于D选项，如果是一般二叉树是不能的，因为左右子树此时是无法区分的。但在BST中，左子树都小于右子树，则可以区分出左右子树，故可建成排序二叉树

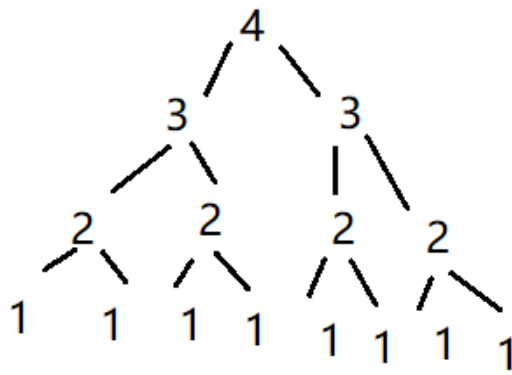
**6、请问下面这个函数的时间复杂度是多少（假设  $n > 0$ ）？**

```
function fn(n)
{
    if (n === 1)
        return 1;
    else
        return fn(n-1) + fn(n-1);
}
```

- A.  $O(n)$
- B.  $O(n \log n)$
- C.  $O(n^2)$
- D.  $O(2^n)$

**正确答案: D**

这里值得注意的是，里面有两个递归，是先执行第一个递归再执行后一个，故过程就是遍历这个树，复杂度为 $2^n$



牛客@疯狂踩坑人

7、关于下面这段 JavaScript 代码说法正确的是

```
var i = 0;
(function fn() {
  console.log(i++);
  setTimeout(fn, 1000);
})();
```

- A.fn是个递归函数
- B.i每隔一秒累加一次
- C.这段代码不会终止
- D.这段代码运行一定时间之后会导致栈溢出

正确答案: C

每个立即执行函数拥有单独的作用域

8、假设实现了一个归并排序（merge sort），这个实现在最坏情况下需要 30 秒的时间来处理一个长度为 64 的输入。请问该实现在 6 分钟内可以处理的最大输入长度最接近多少？

- A.256
- B.512
- C.1024
- D.2048

正确答案: B

设常数是x。已知时间复杂度为 $n \log(n)$ ，所以  $x n \log(n) = 0.5 \text{min}$ ， $n=64$ 。将求得的x带入 $x n \log(n) = 6$  得n大概为512。

## 9、在有 100 个元素的数组中找出其中最大的以及最小的元素所需的最少比较次数是多少次？

- A.145
- B.146
- C.148
- D.150

**正确答案: C**

第一次对100个数字两两比较,用了50次得到{较大的50数字},{较小的50个数字},在分别对着两个数组求最大值/最小值,所以共需 $50+49+49 = 148$ 次。

$1 + (100-2) + (100-2) / 2 = 148$ 。

先取出两个元素，比较1次，较大的为max，较小的为min:

之后的 $(100-2)$ 个元素，每个与max和min比较。

如果需要比较的元素比max大，或者比min小，则只需要比较一次，共比较 $(100-2)$ 次；

如果需要比较的元素比max小，则需再次与min比较。出现这种情况的概率为0.5，所以需比较 $(100-2) / 2$ 次；

第一次先一直对半比较得到最大值，记得存储第一次比较之后的50个较小值（里面一定有最小值）。所以第一次用了99次比较出来了最大值。第二次从事先存储的50个比较最小值，比较49次后可得最小值。故最优方法的比较次数为 $99+49=148$ 次。

## 10、关于 HTTP 的说法正确的是

- A.状态码301表示永久重定向，状态码302表示临时重定向
- B.因为HTTP协议是无状态协议，所以每个HTTP请求/应答客户端和服务端都要新建一个连接，完成之后立即断开连接
- C.检测请求的Referer可以用来抵御CSRF攻击
- D.Node.js发起的HTTP请求不存在跨域限制

**正确答案: A D**

后端不存在跨域问题，Referer可以检测到CSRF攻击，而不会去抵御CSRF攻击。

11、在一座荒岛上有三种变色龙，分别是 12 只棕色，15 只灰色以及 16 只黑色。当两只不同颜色的变色龙相遇时他们会同时变色为另外一种颜色，例如当 1 只棕色和 1 只灰色的变色龙相遇时他们会同时变成黑色的。请问这个荒岛上这些变色龙可能全部变成同一种颜色吗？请详细说明你的分析过程。

参考答案

1黑1灰变2棕，此时有12+2棕，15-1灰，16-1黑。14棕14灰变28黑，全黑。

充分条件：其中两种的数量恰好相等

1.

1只黑色与1只灰色相遇，变成2只棕色

棕色总数 =  $12 + 2 = 14$

2.

现在：

棕色总数 = 14

灰色总数= 14

棕和灰一一相遇，全部变成黑色

12棕色 + 2x

15灰色 -x

16黑色 -x

分别标记 a b c

abc三者轮流交换

while(1){

$a+2x == b-x \parallel a+2x == c-x ;$

OK!

}

or not //优化 升序排序