

定义局部最小的概念。arr长度为1时，arr[0]是局部最小。arr的长度为N(N>1)时，如果arr[0]<arr[1]，那么arr[0]是局部最小；如果arr[N-1]<arr[N-2]，那么arr[N-1]是局部最小；如果0<i<N-1，既有arr[i]<arr[i-1]又有arr[i]<arr[i+1]，那么arr[i]是局部最小。给定无序数组arr，已知arr中任意两个相邻的数都不相等，写一个函数，只需返回arr中任意一个局部最小出现的位置即可。

给定一个double类型的数组arr，其中的元素可正可负可0，返回子数组累乘的最大乘积。例如arr=[-2.5, 4, 0, 3, 0.5, 8, -1]，子数组[3, 0.5, 8]累乘可以获得最大的乘积12，所以返回12。

给定一棵完全二叉树的头节点head，返回这棵树的节点个数。如果完全二叉树的节点数为N，请实现时间复杂度低于O(N)的解法。

给定两个有序数组arr1和arr2，两个数组长度都为N，求两个数组中所有数的上中位数。

例如：

arr1 = {1, 2, 3, 4};

arr2 = {3, 4, 5, 6};

一共8个数则上中位数是第4个数，所以返回3。

arr1 = {0, 1, 2};

arr2 = {3, 4, 5};

一共6个数则上中位数是第3个数，所以返回2。

要求：时间复杂度O(logN)

给定两个有序数组arr1和arr2，在给定一个整数k，返回两个数组的所有数中第K小的数。

例如：

arr1 = {1, 2, 3, 4, 5};

arr2 = {3, 4, 5};

K = 1;

因为1为所有数中最小的，所以返回1；

arr1 = {1, 2, 3};

`arr2 = {3, 4, 5, 6};`

`K = 4;`

因为3为所有数中第4小的数，所以返回3；

要求：如果arr1的长度为N，arr2的长度为M，时间复杂度请达到 $O(\log(\min\{M, N\}))$ 。