

算法作业1

分治法

1. 给你一个整数数组 `nums`，其中元素已经按升序排列，请你将其转换为一棵高度平衡二叉搜索树。

高度平衡二叉树是一棵满足「每个节点的左右两个子树的高度差的绝对值不超过 1」的二叉树。

2. 输入一个长度为 `n` 的整型数组 `array`，数组中的一个或连续多个整数组成一个子数组，子数组最小长度为 1。求所有子数组的和的最大值，当所有整数均为负数时，定义最大字段和为 0，要求用分治法求解

3. 编写一个高效的算法来搜索 `m x n` 矩阵 `matrix` 中的一个目标值 `target`。该矩阵具有以下特性：
每行的元素从左到右升序排列。每列的元素从上到下升序排列。

实例：

1	4	7	11	15
2	5	8	12	19
3	6	9	16	22
10	13	14	17	24
18	21	23	26	30

4. 在数组中的两个数字，如果前面一个数字大于后面的数字，则这两个数字组成一个逆序对。输入一个数组，求出这个数组中的逆序对的总数 `P`，题目保证输入的数组中没有的相同的数字

要求：空间复杂度 $O(n)$ ，时间复杂度 $O(n\log n)$

示例：

输入：[1,2,3,4,5,6,7,0]
返回值：7

动态规划

1. 给定一个整数数组 cost ，其中 $\text{cost}[i]$ 是从楼梯第 i 个台阶向上爬需要支付的费用，下标从0开始。一旦你支付此费用，即可选择向上爬一个或者两个台阶。你可以选择从下标为 0 或下标为 1 的台阶开始爬楼梯。

请你计算并返回达到楼梯顶部的最低花费。

示例：

输入：[2,5,20]
返回值：5
说明：你将从下标为1的台阶开始，支付5，向上爬两个台阶，到达楼梯顶部。总花费为5

2. 给定两个字符串 str1 和 str2 ，输出两个字符串的最长公共子串，题目保证 str1 和 str2 的最长公共子串存在且唯一。

要求：空间复杂度 $O(n^2)$ ，时间复杂度 $O(n^2)$

3. 给定一个 $n * m$ 的矩阵 a ，从左上角开始每次只能向右或者向下走，最后到达右下角的位置，路径上所有的数字累加起来就是路径和，输出所有的路径中最小的路径和。

要求：时间复杂度 $O(nm)$

例如：当输入[[1,3,5,9],[8,1,3,4],[5,0,6,1],[8,8,4,0]]时，对应的返回值为12，

所选择的最小累加和路径如下图所示：

1	3	5	9
8	1	3	4
5	0	6	1
8	8	4	0

4. 在下面的数字三角形中寻找一条从顶部到底部的路径，使得路径上所经过的数字之和最小。路径上的每一步都只能向下或者右下走，求出最大和即可

输入：

4 //三角形行数

2

3 4

6 5 7

4 1 8 3

返回值：11

说明：最小路径是 2 , 3 , 5 , 1

5. 输入一个长度为n的整型数组array，数组中的一个或连续多个整数组成一个子数组，子数组最小长度为1。求所有子数组的和的最大值，当所有整数均为负数时，定义最大字段和为0，要求用动态规划求解

6. 给定数组arr，arr中所有的值都为正整数且不重复。每个值代表一种面值的货币，每种面值的货币可以使用任意张，再给定一个aim，代表要找的钱数，求组成aim的最少货币数。如果无解，请返回-1。

要求：时间复杂度 $O(n \times aim)$ ，空间复杂度 $O(aim)$

QQ : 1715516004

7. 给定一个长度为 n 的数组 arr ，求它的最长严格上升子序列的长度。

要求：时间复杂度 $O(n^2)$ ，空间复杂度 $O(n)$

8. 给你一个字符串 s ，找到 s 中最长的回文子串

