

## 1. Atcoder ABC248 F-keep connect 标签：动态规划

AC记录：

2022-04-19 16:05:27	F - Keep Connect	selphine Q	C++ (GCC 9.2.1)	500	721 Byte	AC	811 ms	284852 KB	<a href="#">Detail</a>
---------------------	------------------	------------	-----------------	-----	----------	----	--------	-----------	------------------------

解题思路：

上下对应的两个点为一列， $F(i, k, c)$  表示第  $i$  列之前，去掉  $k$  条边，并且当前， $c == 1$  表示联通的状态的去边方法数， $c == 0$  表示不连通的状态的去边方法数。

显然得出结论

$$F(i, k, 1) :=$$

$$F(i-1, k, 1) + F(i-1, k, 0) + 3 \times F(i-1, k-1, 1)$$

$$F(i, k, 0) :=$$

$$F(i-1, k-1, 0) + 2 \times F(i-1, k-2, 1)$$

## 2. BZOJ/洛谷 [POI2011] MET-Meteors 标签：并行二分

AC记录：

 nintendo 04-19 19:46:43	Accepted 100	P3527 [POI2011]MET-Meteors	⌚ 6.89s / 📄 53.36MB / 🗃 2.82KB C++14 (GCC 9) 02
--	-----------------	----------------------------	--

解题思路：

假设我们只需要知道一个国家在第几波陨石来临，能达到目标的收集数量，那么只需要二分陨石的波数，然后线段树  $ck$  一下就好了。

题目中需要知道每一个国家在第几波陨石来临，能刚好达到收集目标。

那么需要考虑并行二分（分治）在每一次二分过后，把一些询问置于  $(mid+1, r)$  中，一些询问置于  $(l, mid)$  中。

时间复杂度  $O(n \log n)$  级别。

## 3. 代码源 每日一题 806 合法括号序列 标签：线段树、莫队算法

AC记录：

#192751	#813. 合法括号串	selphine	100	1024ms	8852kb	C++11	1.3kb	2022-04-21 13:31:28
---------	-------------	----------	-----	--------	--------	-------	-------	---------------------

解题思路：

这里只讨论线段树的做法，莫队算法在本题中的应用比较没有意义。

线段树的每一个节点表示当前节点维护的区间的最长合法括号序列的长度，以及没有被合法括号序列匹配的剩余左括号和右括号。

在线段树的 *pushup* 操作中，当前线段树节点的最长合法括号序列的长度为：左子节点的合法括号序列长度，右子节点的合法括号序列长度之和，加上左子节点没有被合法括号序列匹配的左括号去匹配右子节点没有被合法括号序列匹配的右括号的匹配出的合法括号序列长度。

本题的线段树其实是支持修改操作的。

## 4.代码源 每日一题 807 举证操作 标签：差分

AC记录：

#194801	#804. 矩阵操作	selphine	100	6743ms	34780kb	C++11	1.7kb	2022-04-22 16:23:42
---------	------------	----------	-----	--------	---------	-------	-------	---------------------

解题思路：

我们很容易可以发现，每一列 + 1 或者每一行 + 1 进行任意次数操作之后，会发现 以下 2 个现象：

- 对于确定的一些操作，操作顺序先后不会影响结果
- 对于同一行的每一个元素，于上一行的差分应该是一样的

至此，我们可以判断矩阵是否可以构造了，然后我们需要知道构造这个矩阵的最小操作数。

先考虑行和先考虑列都可以，枚举对于行列操作，把所有行（列）加到哪个标准最为合适（加完之后所有行的（假设先对行操作）数值大小都一样），然后再矩阵加到0。记录操作步数最小的操作方法。

## 5.ICPC昆明 E Easy String Problem 标签：莫队算法

AC记录：

51926041	Selphni	E	答案正确	174	9448	1427	C++	2022-04-22 19:53:02
----------	---------	---	------	-----	------	------	-----	---------------------

解题思路：

对于删去一个区间，如果不考虑重复情况，那么可能的生成串为  $(l) \times (n - r + 1)$

现在有可能存在重复的字符串，所以要去重。

由于产生的字符串的不连续性，可能产生的重复情况就是：（左边的某个字符代替了右边某个与该字符相同的字符。）

所以重复的情况个数就是： $(\sum (LEFT_i * RIGHT_i))$

其中  $LEFT$  和  $RIGHT$  表示字符  $i$  在给定区间左边，右边出现的次数。

用莫队很容易维护。