

M 爷的线段树

前缀和。设 $a[i]$ 表示以当前点为起点的增加段数量， $b[i]$ 表示以当前点为终点的增加段数量， $flag$ 为前一个点的改变量，初始为 0。

则每一个点最后改变量即为 $flag+a[i]-b[i-1]$ ，查询后输出即可

Std: <https://paste.ubuntu.com/p/nS5HKXpgFn/>

最后一个是谁？

约瑟夫环问题。 N 个人编号为 $1, 2, \dots, N$ ，依次报数，每报到 M 时，淘汰那个人，求最后胜利者的编号。

递推公式：

$$f(N, M) = (f(N-1, M) + M) \% N$$

$f(N, M)$ 表示， N 个人报数，每报到 M 时淘汰那个人，最终胜利者的编号

$f(N-1, M)$ 表示， $N-1$ 个人报数，每报到 M 时淘汰那个人，最终胜利者的编号

std: <https://paste.ubuntu.com/p/2p48Mpyy2D/>

R C

容斥原理。即先不考虑重叠的情况，把包含于某内容中的所有对象的数目先计算出来，然后再把计数时重复计算的数目排斥出去，使得计算的结果既无遗漏又无重复。

Std: <https://paste.ubuntu.com/p/qghHMcn5Xy/>

闰年判断

略

吃兔兔

可以发现第 i 秒的兔子排列等于第 $i-1$ 秒的兔子情况后面接上第 $i-2$ 秒的，由此规律建立两个数组，分别记录第 i 秒时的总兔子数以及大兔子数。考虑到数据范围，数组开到 91 就好。然后就可以想办法求对于无数秒的情况，前 x 只兔子中的大兔子数量。

有了上面的数组，我们可以找到总兔子数小于 x 的第 i 秒。去除第 i 秒的这些兔子， x 剩下的兔子排列其实与第 $i-1$ 秒的前几只兔子排列是一样的。因为 $i+1$ 秒相当于 i 秒加上 $i-1$ 秒。

也就是说对于无数秒情况下的前 x 只兔子，去除掉相当于第 i 秒排列的兔子后，剩下的兔子依旧可以看做是无数秒情况下的前几只兔子。

所以可以循环上述步骤算出，对于无数秒的情况，前 x 只兔子里有几只大兔子。
然后对于每一组输入，算一下左右端点，减一下就是区间大兔子数。

对于两个数组也可以稍微优化一下，第 i 秒的大兔子数等于第 $i-1$ 秒的兔子数，可以优化成一个数组。

<https://paste.ubuntu.com/p/Ycp5h5mG83/>

打乱数组

是一道 **dp** 题。数组 $f[i]$ 记录对于前 i 个数字，保留 i 这个位置的数字，能有的删除方案数。
想要更新 $f[i]$ ，只需要扫一遍 $1 \sim i-1$ 。如果第 k 个数字与第 i 个数字不相等，那么 $f[i]$ 就加上 $f[k]$ 。
因为 $f[i]$ 和 $f[k]$ 分别是以第 i 个数字以及第 k 个数字结尾的，不相同才能构成混乱数组。
还要注意只保留第 i 个数字的情况，所以 $f[i]$ 要加 1 。
最后答案是 $f[i]$ 这个数组的和。

<https://paste.ubuntu.com/p/gss7FM5S3W/>

作业

是一道二分答案的 **dp** 题，还有加上优先队列或者单调队列的优化。
是找来的题，具体解题可以看

<https://blog.csdn.net/elbadaernu/article/details/72649121>

复习

对于每次复习，肯定是复习成绩最低的那门课最赚。

<https://paste.ubuntu.com/p/rXdg5ZJrrD/>