NOIP2023 模拟赛

粉丝问我啃臭键在哪里

Lynkcat

时间：2023 年 6 月 26 日 7:45 ~ 11:45

题目名称	啃臭键在哪里	原神，启动！	雨田天宇	安排 r
题目类型	传统型	传统型	传统型	传统型
目录	`ctrl`	`genshin`	`lty`	`plan`
可执行文件名	`ctrl`	`genshin`	`lty`	`plan`
输入文件名	`ctrl.in`	`genshin.in`	`lty.in`	`plan.in`
输出文件名	`ctrl.out`	`genshin.out`	`lty.out`	`plan.out`
每个测试点时限	1.0 秒	1.0 秒	1.0 秒	2.0 秒
内存限制	1.0 GiB	1.0 GiB	1.0 GiB	1.0 GiB
子任务数目	10	20	20	8
测试点是否等分	是	是	是	否
结果比较方式	全文比较	Special Judge	全文比较	Special Judge

提交源程序文件名

对于 C++ 语言	`ctrl.cpp`	`genshin.cpp`	`lty.cpp`	`plan.cpp`

编译选项

对于 C++ 语言	`-lm -O2`

注意事项（请仔细阅读）

1. 请选手仔细阅读本页内容和题目中的所有信息。
2. 在 Windows 下使用 LemonLime 测评，编译器版本为 G++ 9.3.0，栈空间限制与题目内存限制相同，评测机 CPU 配置和内存大小与选手用机相同。
3. 严禁选手使用任何方式与其他选手和场外人员进行交流。
4. 严禁选手查询互联网、书籍等资料。
5. 建议写对拍。
6. 难度被削弱了（大悲）（愤怒）。

啃臭键在哪里（ctrl）

【问题描述】

粉丝问我，啃臭键在哪里。

励志阿伟现在正处在一个冰火迷宫中，迷宫由 n 个格子组成，每个格子要么是冰之格，要么是火之格，励志阿伟刚开始可以选择从迷宫中任意一个开始走，走到第 i 个位置时会得到值为 a_i 的积分。

如果励志阿伟当前在冰之格，那么他可以选择一个编号大于当前格子的冰之格，跳到那里。如果励志阿伟当前在火之格，那么他可以选择一个编号大于当前格子的火之格，跳到那里。如果励志阿伟目前没有格子可以走，那么结束。

励志阿伟想最大化他的得分，于是他学会了一个超能力，他能在比赛开始的时候改变最多 m 个格子的状态，即将一个格子从冰之格变成火之格或从火之格变成冰之格，改变第 i 个格子的状态会让励志阿伟的得分减少 p_i 。

你能告诉励志阿伟他最多能得几分吗？（特别地，励志阿伟可以选择一个格子都不走积分为 0）。

【输入格式】

从文件 ***ctrl.in*** 中读入数据。

第一行两个正整数 n, m。

第二行 n 个正整数表示 a_i。

第三行 n 个正整数表示 p_i。

第四行 n 个正整数 b_i 为 0 或 1，0 表示这个格子是冰之格，1 表示这个格子是火之格。

【输出格式】

输出到文件 ***ctrl.out*** 中。

输出一个整数表示答案。

【样例 1 输入】

```
3 2
1 15 9
2 5 7
1 0 1
```

【样例 1 输出】

```
20
```

【样例 2 输入】

```
10 3
80 86 57 69 59 52 94 74 9 63
8 32 77 64 53 49 22 68 27 63
1 0 0 0 1 0 0 1 1 1
```

【样例 2 输出】

```
442
```

【数据规模与约定】

对于 20% 的数据，$b_i = 0, 0 \leq a_i, p_i \leq 100$。
对于 50% 的数据，$1 \leq n, m \leq 10, 0 \leq a_i, p_i \leq 100$。
对于 70% 的数据，$1 \leq n, m \leq 1000, -10^5 \leq a_i, p_i \leq 10^5$。
对于 100% 的数据，$1 \leq n, m \leq 10^5, -10^5 \leq a_i, p_i \leq 10^5$。

原神，启动！（genshin）

【问题描述】

你需要构造一个 $n \times n$ 的矩阵，每个点填黑或白，使得相邻格子不同色的对数恰好为 m，否则原始人起洞不了。

【输入格式】

从文件 ***genshin.in*** 中读入数据。

第一行一个数 T 表示数据组数。

每组数据输入第一行两个整数 n, m。

【输出格式】

输出到文件 ***genshin.out*** 中。

对于每组数据，若无解则输出一行 Impossible。否则输出 $n+1$ 行，第一行输出 Possible，接下来输出一个 $n \times n$ 的 01 矩阵，显然 01 分别代表什么颜色并不重要。

输出任一解即可。

【样例 1 输入】

```
2
3 6
3 1
```

【样例 1 输出】

```
Possible
010
100
000
Impossible
```

【数据规模与约定】

对于 100% 的数据，保证 $1 \leq T \leq 5$，$1 \leq n \leq 10^3, 1 \leq m \leq 2 \times n \times (n-1)$。

测试点编号	$n \leq$	特殊限制
$1 \sim 5$	10	
$6 \sim 10$	20	
$11 \sim 15$	10^3	$m <= n$
$16 \sim 20$	10^3	

雨田天宇（lty）

【问题描述】

曾像夜那么黑，每个清晨，曾阻挡每个梦，每一道门。终于也可能，无限可能，自由发生…

定义一个长度为 n 的序列 a 的权值为所有重排 a 之后的新序列 a' 的 $\sum_{i=1}^{n}|a'_i - a'_{n-i+1}|$ 的最大值。

求所有长度为 n 的每个数取值在 $[1,m]$ 的序列的权值的和，对 998244353 取模。

【输入格式】

从文件 ***lty.in*** 中读入数据。

第一行两个正整数 n,m。

【输出格式】

输出到文件 ***lty.out*** 中。

一行一个整数输出答案。

【样例 1 输入】

```
3 3
```

【样例 1 输出】

```
72
```

【样例 2 输入】

```
343 343
```

【样例 2 输出】

```
416089362
```

【数据规模与约定】

对于 100% 的数据，保证 $1 \leq n, m \leq 5000$。

测试点编号	$n \leq$	$m \leq$
$1 \sim 2$	5	5
$3 \sim 5$	100	100
$6 \sim 10$	500	500
$11 \sim 15$	1000	1000
$16 \sim 20$	5000	5000

安排 r (plan)

【题目描述】

安排 r

定义一个长度为 n 的序列 a 前缀和序列为另一个长度为 n 的序列 b 使得对于所有 i 都有 $b_i = \sum_{j=1}^{i} a_j$。

定义一个长度为 n 的序列 a 后缀和序列为另一个长度为 n 的序列 b 使得对于所有 i 都有 $b_i = \sum_{j=i}^{n} a_j$。

定义将一个序列一般化指把序列中的每个数都跟 10^{18} 取 min，然后跟 -10^{18} 取 max。

现在给你一个长度为 n 的序列，你可以执行以下三种操作：

1：将序列中所有数取反。

2：选取一个区间 $[l,r](1 \le l \le r \le n)$，将 $[l,r]$ 这个区间变为这个区间的前缀和序列，然后将 a 一般化。

3：选取一个区间 $[l,r](1 \le l \le r \le n)$，将 $[l,r]$ 这个区间变为这个区间的后缀和序列，然后将 a 一般化。

你需要花费最少的次数使得序列中每个数都是非负整数。（注意，在某些子任务中你不一定需要花费最少的次数，请阅读数据范围与提示部分。）

【输入格式】

从文件 plan.in 中读入数据。第一行两个整数 n, id，其中 id 表示子任务编号，$id = 0$ 表示样例。第二行 n 个整数表示序列 a_i。

【输出格式】

输出到文件 plan.out 中。

第一行输出一个数 m 表示操作次数。

接下来 m 行，若执行操作 1 则输出一行一个数 1，否则输出一行三个数表示第几种操作，l 和 r。

如果有多解输出任意一种即可。

【样例 1 输入】

```
7 0
0 0 1 -1 -1 -1 1
```

【样例 1 输出】

```
2
3 1 3
2 1 7
```

【数据范围及提示】

本题采用捆绑测试。

对于 100% 的数据，满足 $1 \le n \le 2 \times 10^5, -1 \le a_i \le 1$。

各子任务的附加限制如下表所示：

- 子任务 1 (10 pt)：保证最优解法操作次数不超过 1。
- 子任务 2 (10 pt)：你的解法获得满分的条件为次数小于等于 100。
- 子任务 3 (19 pt)：你的解法获得满分的条件为次数小于等于最优次数 +3。
- 子任务 4 (7 pt)：你的解法获得满分的条件为次数小于等于最优次数 +1。
- 子任务 5 (7 pt)：$n \le 3000$，保证存在一种最优解只需要用第二种操作。
- 子任务 6 (19 pt)：保证存在一种最优解只需要用第二种操作。
- 子任务 7 (18 pt)：$n \le 3000$。
- 子任务 8 (10 pt)：$n \le 2 \times 10^5$。