

NOIP 2024 模拟赛

CDQZ

测试时间：2024.09.19

题目名称	线段	数据	飞机	红温的猴子
题目类型	传统型	传统型	传统型	传统型
目录	seg	data	plane	monkey
可执行文件名	seg	data	plane	monkey
输入文件名	seg.in	data.in	plane.in	monkey.in
输出文件名	seg.out	data.out	plane.out	monkey.out
提交文件名	seg.cpp	data.cpp	plane.cpp	monkey.cpp
时间限制	1.5 秒	1.0 秒	1.0 秒	1.0 秒
内存限制	512 MiB	512 MiB	512 MiB	512 MiB
子任务数目	7	9	5	7
测试点是否等分	否	否	否	否
编译选项	-O2 -std=c++14			

【注意事项（请仔细阅读）】

1. 选手提交的源程序请**直接放在个人目录下**，无需建立子文件夹；
2. 文件名（包括程序名和输入输出文件名）必须使用英文小写。
3. C++ 中函数 `main()` 的返回值类型必须是 `int`，值必须为 `0`。
4. **对于因未遵守以上规则对成绩造成的影响，相关申诉不予受理。**
5. 若无特殊说明，结果比较方式为**忽略行末空格、文末回车后的全文比较**。
6. 程序可使用的栈空间大小与该题内存空间限制一致。
7. 在终端中执行命令 `ulimit -s unlimited` 可将当前终端下的栈空间限制放大，但你使用的栈空间大小不应超过题目限制。
8. 若无特殊说明，每道题的**代码大小限制为 100KB**。
9. 若无特殊说明，输入与输出中同一行的相邻整数、字符串等均使用一个空格分隔。
10. 输入文件中可能存在行末空格，请选手使用更完善的读入方式（例如 `scanf` 函数）避免出错。
11. 直接复制 PDF 题面中的多行样例，数据将带有行号，建议选手直接使用对应目录下的样例文件进行测试。

12. 使用 `std::deque` 等 STL 容器时，请注意其内存空间消耗。
13. 请务必使用题面中规定的的编译参数，保证你的程序在本机能够通过编译。此外不允许在程序中手动开启其他编译选项，一经发现，本题成绩以 0 分处理。
14. 统一评测时采用的机器配置为：12th Gen Intel(R) Core(TM) i7-12700 2.10 GHz，内存 16GB。上述时限以此配置为准。
15. 评测在 Windows 11 下进行，使用 LemonLime 进行评测。如果对此条以及 14 条中的机器配置有疑问，请及时询问。
16. 不保证题目严格按照难度顺序排序。
17. 题目较简单，请已经 AK 的同学不要影响到其它同学做题。

线段 (seg)

【题目背景】

造题面的人憋了好久憋不出来一个比较优秀的题目背景，于是干脆直接放简要题意了。

【题目描述】

给定 n 条线段，两端点为 $(0, a_i)$ 和 (D, b_i) ，每条线段有一个权值 w_i 。保证这些线段不存在三线共点的情况。

设 $f(i, x)$ 表示 i 号线段在横坐标 x 处的纵坐标值。定义一个线段 p 在横坐标 x 处的耗能值为：

$$\sum_{f(i,x) > f(p,x)} w_i$$

现在小 G 提出了 q 次询问，每次询问给定 p, l, r ，表示对于线段 p ，横坐标在区间 $[l, r]$ 内的最大耗能值。

【输入格式】

从文件 `seg.in` 中读入数据。

本题输入强制在线。

第一行三个整数，表示 D, n, q 。

接下来 n 行，每行三个整数 a_i, b_i, w_i 。

接下来 q 行，每行三个整数 p', l', r' 。每次询问真实的 p, l, r 需要以下计算得到：

$$p = (p' + lstans) \bmod n + 1$$

$$l = (l' + lstans) \bmod D + 1$$

$$r = (r' + lstans) \bmod D + 1$$

其中 $lstans$ 表示上一次询问的答案。

特别地，第一次询问时 $lstans = 0$ 。

【输出格式】

输出到文件 `seg.out` 中。

输出一共 q 行，表示每次询问的答案。

【样例 1 输入】

```
1 10 8 10
2 4 8 1
3 1 6 9
4 3 5 6
5 1 7 10
6 2 7 2
7 4 10 6
8 2 8 2
9 2 3 9
10 1 3 9
11 7 4 7
12 7 9 6
13 4 3 7
14 7 0 7
15 3 5 7
16 1 1 5
17 5 5 3
18 0 6 8
19 4 7 0
```

【样例 1 输出】

```
1 27
2 11
3 30
4 30
5 0
6 11
7 15
8 15
9 36
10 6
```

【样例 2 ~ 8】

见选手目录下的 *seg/seg2 ~ 8.in* 与 *seg/seg2 ~ 8.ans*。
该部分样例按标号顺序分别满足子任务 1 至 7 的限制。

【测试点约束】

本题使用捆绑测试。

子任务编号	特殊限制	分值
1	$n \leq 8, q \leq 30$	5
2	$n \leq 50, q \leq 3000$	10
3	$n \leq 500, q \leq 3000$	20
4	$D \leq 1000$	18
5	$q \leq 3000$	12
6	$n \leq 1000$	15
7	无	20

对于所有数据，满足 $1 \leq n \leq 3000, 1 \leq q \leq 10^6, 1 \leq D, a_i, b_i, w_i \leq 10^9, l \leq r$ 。
本题输入量较大，请选手选用合适的输入输出方式。

数据 (data)

【题目背景】

暴龙送 TON 事件之后，其它的动物们常常拿这个事情乳暴龙。

【题目描述】

受此启发，幽灵出了一道输入是一个仅包含 **A,T,O,N** 的字符串的题准备让暴龙做。它很快就把这个题的数据造完了。

但考拉想要捣乱，它想要将每个输入数据中的字符打乱。

考拉的想法素质很差，因为幽灵并不擅长还原数据，它每次只能交换相邻的两个字符。

设数据 S 打乱后的数据为 S' ，定义数据 S 的打乱度为幽灵要将 S' 还原成 S 需要的最小交换次数。

考拉想要打乱度最大，于是他找到了你。你需要对于每一个给定的字符串 S ，输出它的最大打乱度，并且给出任意一个达到最大打乱度的 S' 。

请选手注意本题的评分方式与正常传统题不同，详见提示。

【输入格式】

从 *data.in* 中读入数据。

本题测试点中含有多组数据。

第一行两个整数 T ，表示数据组数。

接下来 T 行，每行一个仅包含 **A,T,O,N** 的字符串 S 。

【输出格式】

输出到文件 *data.out* 中。

一共 T 行，每行一个整数 w 和一个字符串 S' ，表示最大打乱度与其构造。

【样例 1 输入】

```
1 3
2 NNO
3 NT
4 OAANTTON
```

【样例 1 输出】

```
1 2 ONN
2 1 TN
3 17 TNNTA00A
```

【样例 2 ~ 10】

见选手目录下的 `data/data2 ~ 10.in` 与 `data/data2 ~ 10.ans`。
该部分样例按标号顺序分别满足子任务 1 至 9 的限制。

【测试点约束】

本题使用捆绑测试。令 $n = |S|$ 。

子任务编号	特殊限制	分值
1	$n \leq 8, \sum n \leq 30$	5
2	$n \leq 25, \sum n \leq 100$	10
3	$\sum n \leq 500$	20
4	$\sum n \leq 2000$	15
5	S 中仅包含一种字符, $\sum n \leq 2 \times 10^5$	5
6	S 中仅包含两种字符, $\sum n \leq 2 \times 10^5$	10
7	S 中仅包含三种字符, $\sum n \leq 2 \times 10^5$	10
8	$\sum n \leq 2 \times 10^5$	10
9	无	15

对于所有的数据, 满足 $1 \leq T \leq 10, 1 \leq n \leq 2 \times 10^5, 1 \leq \sum n \leq 10^6$, 保证 S 中仅含有字符 `N,A,T,O`。

【提示】

本题使用 **Special Judge**。

下发样例文件只有 T 行, 每行一个整数, 表示该组数据的最大打乱值。
本题不下发答案校验器, 请选手自行校验。

对于一个测试点, 若你在所有 T 组数据中的最大打乱值全部正确, 那么你可以得到该测试点 50% 的分数。若你在所有 T 组数据中 S' 的打乱值均等于最大打乱值, 那么你也可以得到该测试点 50% 的分数。若两者均正确, 那么你将获得该测试点 100% 的分数。

飞机 (plane)

【题目背景】

众所周知，猴子喜欢开飞机。

【题目描述】

我们称一个非空序列 A 是严格单峰的，那么存在一个下标 x ，满足：

$$A_1 < A_2 < \cdots < A_x, A_x > A_{x+1} > \cdots > A_n$$

同时我们定义这个序列的峰是 x 。

有一天，猴子挑战同时开两个飞机，考拉在猴子开飞机的时候，对于每个时刻都记录下了其中一个飞机目前的高度。考拉保证它提供的高度序列是按时间顺序排列的。

猴子自诩自己开飞机技术很牛，所以猜想考拉记录的两个飞机的高度子序列都是严格单峰的。

在这个猜想下，猴子想知道有多少可能的点对 (x, y) ，其中 $x < y$ ，满足 x 是记录的第一个飞机的高度子序列的峰在原序列中的位置， y 是记录的第二个飞机的高度子序列的峰在原序列中的位置。

【输入格式】

从文件 *plane.in* 中读取数据。

第一行输入包含一个整数 n ，表示高度序列的长度。

第二行输入包含 n 个整数 h_i ，表示考拉记录的高度序列。

【输出格式】

输出到文件 *plane.out* 中。

输出一行一个整数，表示可能的点对个数。

【样例 1 输入】

1	3
2	1 2 3

【样例 1 输出】

12

【样例 1 解释】

可能的点对有 (1,3) 和 (2,3)。

【样例 2 输入】

17
29 5 3 10 2 6 8

【样例 2 输出】

10

【样例 3 ~ 7】

见选手目录下的 *plane/plane3 ~ 7.in* 与 *plane/plane3 ~ 7.ans*。
该部分样例按标号顺序分别满足子任务 1 至 5 的限制。

【测试点约束】

本题使用捆绑测试。

子任务编号	特殊性质	分值
1	$n \leq 15$	10
2	$n \leq 500$	20
3	保证给定高度序列是严格单峰的	10
4	$n \leq 5000$	25
5	无	35

对于所有数据， $1 \leq n \leq 10^6, 1 \leq h_i \leq 10^9$ 。

红温的猴子 (monkey)

【题目背景】

考拉喜欢调戏猴子，猴子经常因此红温。

牛非常喜欢猴子红温的样子，所以某天也开始调戏猴子了。牛非常过分，对猴子干了一些不可理喻的事情，这使得猴子比以往更加红温。

这种状态下，猴子急切的想惩罚牛，他想到牛通常会去吃草，于是想破坏牛吃草的所有路线。

【题目描述】

牛吃草的路线可以看做是一个 n 个点 m 条边的无向图，猴子想要破坏所有 m 条边。

猴子可以从任意位置出发开始进行破坏，它有两种破坏边的方式。

- 第一种简称**红温模式**，猴子会破坏掉自己刚走过的边。
- 第二种简称**暴走模式**，猴子行走的时候会统计这是这个模式开始后走到的第几条边（记这个数是 k ），如果 k 是偶数，那么这条边将在其走完这条边后被其破坏，如果 k 是奇数，那么就不破坏这条边。

注意，猴子不能走在已经被它破坏的边上。

目前猴子处于红温模式，但是它可以在任意点切换到暴走模式。因为牛真的很过分，所以猴子没有办法从暴走模式切换到红温模式。

猴子想知道自己能不能将所有 m 条边全部破坏。但是猴子太懒了，所以把这个问题抛给了你，还要求你给出一个构造。

请选手注意本题的评分方式与正常传统题不同，详见提示。

【输入格式】

从文件 `monkey.in` 中读取数据。

第一行输入包含三个整数 c, n, m ，表示该数据属于子任务 c ，图的点数为 n ，图的边数为 m 。样例中 c 表示该数据满足子任务 c 的限制。

接下来 m 行，每行包含 2 个整数 u_i, v_i ，表示图上的一条无向边。

【输出格式】

输出到文件 `monkey.out` 中。

以下把切换模式和走一条边统称为**操作**。

输出第一行为一个整数 p ， $p = 1$ 表示猴子能破坏完所有边， $p = 0$ 表示猴子不能破坏完所有边。

若 $p = 1$ ，那么第二行输出两个整数 k, st ，表示猴子进行了多少次操作和猴子从哪个点出发。接下来 k 行，每行首先一个整数 op ，如果 $op = 1$ ，后面接一个数 id ，表示猴子走过了第 id 条边；如果 $op = -1$ ，表示猴子切换模式。

你需要保证 $0 \leq k \leq 2m + 2, 1 \leq st \leq n, 1 \leq id \leq m$ 。

【样例 1 输入】

```
1 1 4 6
2 1 2
3 1 3
4 1 4
5 2 3
6 2 4
7 3 4
```

【样例 1 输出】

```
1 1
2 9 3
3 1 4
4 1 5
5 1 6
6 1 2
7 -1
8 1 1
9 1 1
10 1 3
11 1 3
```

【样例 2 输入】

```
1 1 4 2
2 1 2
3 3 4
```

【样例 2 输出】

10

【样例 3 ~ 9】

见选手目录下的 `monkey/monkey3 ~ 9.in` 与 `monkey/monkey3 ~ 9.ans`。
该部分样例按标号顺序分别满足子任务 1 至 7 的限制。

【测试点约束】

本题使用捆绑测试。

子任务编号	特殊性质	分值
1	$n \leq 10$	12
2	保证图无解	4
3	保证图有解，且存在一种构造不需要切换模式	12
4	保证图有解， $n, m \leq 50$	16
5	$n, m \leq 500$	16
6	保证每个点度数小于等于 4， $n, m \leq 1000$	20
7	无	20

对于所有数据， $1 \leq n \leq 3 \times 10^3, 0 \leq m \leq 3 \times 10^3, 1 \leq u_i, v_i \leq n$ ，保证图不出现自环或重边。

【提示】

本题使用 Special Judge。

对于一个测试点：

- 如果答案无解并且选手输出无解，那么选手将获得该测试点 100% 的分数。
- 如果答案有解并且选手输出有解，但是构造错误，那么选手将获得该测试点 25% 的分数。
- 如果答案有解，选手输出有解并且构造正确，那么选手将获得该测试点 100% 的分数。

本题下发文件中的 `checker.cpp` 可以用来检验你的输出的正确性。使用时，请先编译 `checker.cpp` 得到可执行文件，然后运行如下命令：

- Windows: `checker.exe <in> <out> <ans>`
- Linux: `./checker <in> <out> <ans>`

其中 `<in>` 表示输入文件名, `<out>` 表示你的输出文件名, `<ans>` 表示答案文件名。

下发样例答案文件中, 对于每组数据只有一行一个整数 p , 其中 $p = 0$ 时代表无解, $p = 1$ 时代表有解。

由于搬题人很菜, 如果使用下发的 `checker.cpp` 遇到问题, 请及时提出。