

IOI 2019 中国国家队选拔

CTS 2019

第二试

时间：2019 年 5 月 14 日 08:30 ~ 13:30

题目名称	田野	重复	氮金手游
题目类型	传统型	传统型	传统型
目录	fields	repeat	fgo
可执行文件名	fields	repeat	fgo
输入文件名	fields.in	repeat.in	fgo.in
输出文件名	fields.out	repeat.out	fgo.out
每个测试点时限	4.0 秒	1.0 秒	1.0 秒
内存限制	512 MB	512 MB	512 MB
子任务数目	20	10	20
测试点是否等分	是	是	是

提交源程序文件名

对于 C++ 语言	fields.cpp	repeat.cpp	fgo.cpp
对于 C 语言	fields.c	repeat.c	fgo.c
对于 Pascal 语言	fields.pas	repeat.pas	fgo.pas

编译选项

对于 C++ 语言	-O2 -std=c++11
对于 C 语言	-O2 -std=c11
对于 Pascal 语言	-O2

田野 (fields)

【题目描述】

Last night I saw you running

In the open fields of grace

No longer were you broken or in pain¹

你找到了一片一望无际的大田野，在这片田野中你忘记了曾经破碎、痛苦的过去。你像小孩一样在上帝的恩赐中奔跑。

然而你发现了一个问题，在这片田野中有若干条峡谷。你随时都有坠入峡谷中的危险。为了继续自由自在的奔跑，你决定用若干围栏将这些峡谷围起来。

我们可以忽视峡谷的宽度，将每一条峡谷看做一条线段。这些线段可以相交，而你的围栏必须是一条或多条闭合不自交且两两不相交的曲线，使得任何一个峡谷都完全在某一条闭合曲线围成的闭合区域之内。

当然，围栏需要消耗资源，消耗的资源 and 围栏的长度成正比，你希望最小化消耗的资源总量，所以希望你求出围栏总长度的下确界，换句话说，希望你找到一个最大的实数 x ，使得不存在一个方案使得围栏总长度小于 x 。

【输入格式】

从文件 *fields.in* 中读入数据。

输入文件的第一行为一个整数 n ，表示峡谷的个数。

接下来 n 行，第 i 行四个整数 a_i, b_i, c_i, d_i ，表示第 i 条峡谷为一条连接点 (a_i, b_i) 和点 (c_i, d_i) 的线段。保证两个端点不重合，不同的线段不会涉及到相同的点。保证任意三点不共线。

【输出格式】

输出到文件 *fields.out* 中。

输出一行一个实数，表示围栏总长度的下确界。你的答案和标准答案的绝对误差和相对误差的最小值不能超过 10^{-6} 。

【样例 1 输入】

1

0 0 0 1

¹题目中的歌词来自 Jackie Evancho 的 Open Fields of Grace，作曲者为 Lisa Venkatrathnam 和 Paul Sumares。

【样例 1 输出】

2.00000000

【样例 1 解释】

一个四个端点分别为 $(-0.01, -0.01)$, $(-0.01, 1.01)$, $(0.01, 1.01)$, $(0.01, -0.01)$ 的长方形完全包含输入的线段, 且总长度为 2.08, 略大于下确界。

我们可以证明, 不存在长度恰好为 2 的方案。我们可以通过将正方形无限向输入线段“缩紧”来构造一个长度为任意大于 2 的方案。

【样例 2 输入】

4
-1 7 0 7
0 0 0 1
2 -3 5 5
2 2 6 -1

【样例 2 输出】

23.563573998194637061425470524757

【样例 2 解释】

下图为输入的线段, 注意线段可以相交:

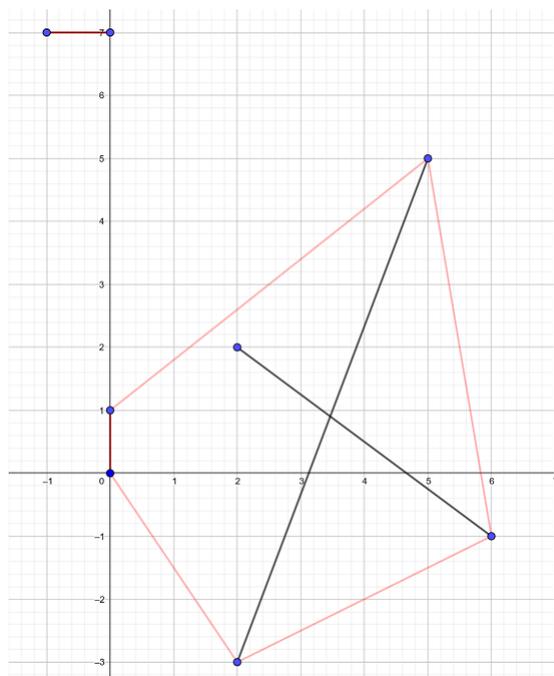


图 1: 样例 2 解释

我们可以通过无限“逼近”这些红色的曲线来构造任意总长度大于答案的方案。注意通过样例 1，我们很容易知道左上角的红色线段被算了两遍。

【样例 3 输入】

```
4
-1 1 -1 3
0 4 2 4
3 1 3 3
0 0 2 0
```

【样例 3 输出】

```
13.656854249492380195206754896839
```

【样例 3 解释】

答案为 $8 + 4\sqrt{2}$ 。

解释如图：

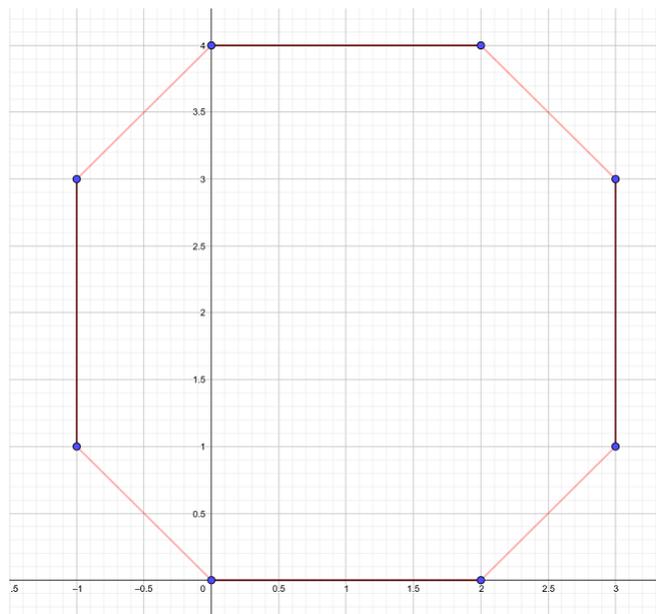


图 2: 样例 3 解释

我们可以通过无限“逼近”这些红色的曲线来构造任意总长度大于 $8 + 4\sqrt{2}$ 的方案。

【样例 4】

见选手目录下的 *fields/fields4.in* 与 *fields/fields4.ans*。

【测试数据约定】

对于 5% 的数据，保证 $1 \leq n \leq 1$ 。

对于 10% 的数据，保证 $1 \leq n \leq 2$ 。

对于 15% 的数据，保证 $1 \leq n \leq 10$ 。

对于 30% 的数据，保证 $1 \leq n \leq 15$ 。

对于 45% 的数据，保证 $1 \leq n \leq 30$ 。

对于 55% 的数据，保证 $1 \leq n \leq 60$ 。

对于 65% 的数据，保证 $1 \leq n \leq 120$ 。

对于 75% 的数据，保证 $1 \leq n \leq 200$ 。

对于另外 10% 的数据，保证答案最多包含两条曲线。

对于 100% 的数据，保证 $1 \leq n \leq 250$, $0 \leq |a_i|, |b_i|, |c_i|, |d_i| \leq 10^9$ 。保证两个端点不重合，不同的线段不会涉及到相同的点。保证任意三点不共线。

重复 (repeat)

【题目背景】

贾樟柯在《山河故人》里说，“生活就是重复”。在生活中，人们总是喜欢重复自己做过的事情。语言就是一个很经典的例子。

比如，我们在表示疑问时，总不满足于使用一个问号“?”；使用一连串的问号“????”总是显得比较有力。

在表示抱歉时，一句“对不起”总显得不够情愿；连着表示“对不起对不起”才足够表达自己的真诚。

【题目描述】

A 国就是一个喜欢重复的国家。在这个国家中，一个基本句子可以用一个长度恰好为 m 的小写字母字符串表示。为了表达自己对重复的喜爱，A 国的人们总喜欢把自己想要表达的句子重复无限多次。

有时，这样的重复是充满意义的。A 国的人们把一个字典序小于给定的字符串 s ，且长度和 s 相同的小写字母字符串称为一个有意义的语义片段。他们想知道，有多少个不同的基本句子（即长度恰好为 m 的小写字母字符串）在经过无限重复后，可以从中找出至少一个有意义的语义片段？

【输入格式】

从文件 `repeat.in` 中读入数据。

输入文件的第一行为一个正整数 m ，表示基本句子的长度；第二行为一个小写字母字符串 s ，其含义详见题目描述。

【输出格式】

输出到文件 `repeat.out` 中。

输出一行一个整数，为满足条件的基本句子的数量。为了避免答案过大，你只需要输出将答案对 998244353 取模后的结果。

【样例 1 输入】

```
3
abc
```

【样例 1 输出】

```
79
```

【样例 2 输入】

5
zxcvb

【样例 2 输出】

11881375

【测试数据约定】

设字符串 s 的长度为 n , 则 n 、 m 的范围遵循如下表格:

测试点编号	m	n	其他限制
1	$m \leq 5$	$n \leq 10$	$m < n$
2	$m \leq 30$	$n \leq 30$	$m = n$
3	$m \leq 50$	$n \leq 30$	$m > n$
4	$m \leq 100$	$n \leq 100$	$m < n$
5	$m \leq 200$	$n \leq 200$	$m = n$
6	$m \leq 300$	$n \leq 200$	$m > n$
7	$m \leq 300$	$n \leq 2000$	$m < n$
8	$m \leq 1000$	$n \leq 1000$	$m = n$
9	$m \leq 2000$	$n \leq 200$	$m > n$
10	$m \leq 2000$	$n \leq 2000$	$m > n$

对于 100% 的数据, 保证 $1 \leq n, m \leq 2000$ 。

氪金手游 (fgo)

【题目描述】

小刘同学是一个喜欢氪金手游的男孩子。

他最近迷上了一个新游戏，游戏的内容就是不断地抽卡。现在已知：

- 卡池里总共有 N 种卡，第 i 种卡有一个权值 W_i ，小刘同学不知道 W_i 具体的值是什么。但是他通过和网友交流，他了解到 W_i 服从一个分布。
- 具体地，对每个 i ，小刘了解到三个参数 $p_{i,1}, p_{i,2}, p_{i,3}$ ， W_i 将会以 $p_{i,j}$ 的概率取值为 j ，保证 $p_{i,1} + p_{i,2} + p_{i,3} = 1$ 。

小刘开始玩游戏了，他每次会氪一元钱来抽一张卡，其中抽到卡 i 的概率为：

$$\frac{W_i}{\sum_j W_j}.$$

小刘会不停地抽卡，直到他手里集齐了全部 N 种卡。

抽卡结束之后，服务器记录下来了小刘第一次得到每张卡的时间 $\{T_i\}$ 。游戏公司在这里设置了一个彩蛋：公司准备了 $N - 1$ 个二元组 (u_i, v_i) ，如果对任意的 i ，成立 $T_{u_i} < T_{v_i}$ ，那么游戏公司就会认为小刘是极其幸运的，从而送给他一个橱柜的手办作为幸运大奖。

游戏公司为了降低获奖概率，它准备的这些 (u_i, v_i) 满足这样一个性质：对于任意的 $\emptyset \neq S \subseteq \{1, 2, \dots, N\}$ ，总能找到 (u_i, v_i) 满足： $u_i \in S, v_i \notin S$ 或者 $u_i \notin S, v_i \in S$ 。

请你求出小刘同学能够得到幸运大奖的概率，可以保证结果是一个有理数，请输出它对 998244353 取模的结果。

【输入格式】

从文件 `fgo.in` 中读入数据。

第一行一个整数 N ，表示卡的种类数。

接下来 N 行，每行三个整数 $a_{i,1}, a_{i,2}, a_{i,3}$ ，而题目给出的 $p_{i,j} = \frac{a_{i,j}}{a_{i,1} + a_{i,2} + a_{i,3}}$ 。

接下来 $N - 1$ 行，每行两个整数 u_i, v_i ，描述一个二元组（意义见题目描述）。

【输出格式】

输出到文件 `fgo.out` 中。

输出一行一个整数，表示所求概率对 998244353 取模的结果。

【样例 1 输入】

```
2
0 0 1
```

1 1 0

1 2

【样例 1 输出】

524078286

【样例 1 解释】

W_2 以 $\frac{1}{2}$ 的概率取 1 或者 2:

- 如果 $W_2 = 1$, 那么 $T_1 < T_2$ 的概率为 $\frac{3}{4}$ 。
- 否则 $W_2 = 2$, $T_1 < T_2$ 的概率为 $\frac{3}{5}$ 。

综合所有情况答案为 $\frac{1}{2} \left(\frac{3}{4} + \frac{3}{5} \right) = \frac{27}{40}$, 你可以验证它对 998244353 取模的结果确实是所给答案。

【样例 2】

见选手目录下的 *fgo/fgo2.in* 与 *fgo/fgo2.ans*。

【测试数据约定】

对于全部的测试数据, 保证 $N \leq 1000$, $a_{i,j} \leq 10^6$.

- 20 分的数据, $N \leq 15$.
- 15 分的数据, $N \leq 200$, 且每个限制保证 $|u_i - v_i| = 1$ 。
- 20 分的数据, $N \leq 1000$, 且每个限制保证 $|u_i - v_i| = 1$ 。
- 15 分的数据, $N \leq 200$ 。
- 30 分的数据, 无特殊限制。