

NOIP 2018

dy0607

October 28, 2018

题目名称	Odyssey	Necklace	Polygon
源文件名	odyssey	necklace	polygon
输入文件名	odyssey.in	necklace.in	polygon.in
输出文件名	odyssey.out	necklace.out	polygon.out
题目类型	传统型	传统型	传统型
每个测试点时限	1.0s	1.0s	1.5s
空间限制	512MB	512MB	512MB
编译命令	-lm -O2 -std=c++11		

Notes:

1. 评测在Ubuntu16.04(64bit)上进行，评测时开启无限栈；
2. 评测机配置为Intel® Pentium(R) CPU G2030 @ 3.00GHz × 2，内存4GB；
3. 题目难度与顺序无关。

1 Odyssey

1.1 Description

Santiago有 n 个梦想中的地点，于是他决定进行一次长途旅行。

整个世界可以视为一个二维平面，而Santiago的家乡Andalusia就在原点上，而另外 n 个点都在第一象限内，其中第 i 个点的坐标为 (X_i, Y_i) 。如果对于两个点 i, j （将原点视为0号点），满足 $\max(X_i, Y_i) \leq \max(X_j, Y_j)$ ，那么 i 到 j 有一条有向道路，道路长度为 $|X_i - X_j| + |Y_i - Y_j|$ 。

Santiago希望找一条从Andalusia开始的路径，经过所有的点至少一次，并最小化路径总长度。

1.2 Input

从文件`odyssey.in`中读入数据。

第一行一个整数 n 。

接下来 n 行，每行两个整数 X_i, Y_i 。

1.3 Output

输出到文件`odyssey.out`中。

一行一个整数表示最小的路径长度，如果路径不存在，输出 $\binom{2333^{2333}}{233^{233}}$ 的值。

1.4 Sample1

1.4.1 Input

```
8
2 2
1 4
2 3
3 1
3 4
1 1
4 3
1 2
```

1.4.2 Output

```
15
```

1.5 Sample2

见选手目录下的 *odyssey/odyssey2.in* 与 *odyssey/odyssey2.ans*。

1.6 Subtasks

对于所有数据，有 $1 \leq n \leq 2 \times 10^5$, $1 \leq X_i, Y_i \leq 10^9$ ，可能有重复的点。

- Subtask1(24%), $n \leq 8$.
- Subtask2(18%), $n \leq 18$.
- Subtask3(16%), $n \leq 5000$.
- Subtask4(16%), X_i, Y_i 在 $[1, n]$ 内随机生成。
- Subtask5(26%), 没有特殊的约束。

2 Necklace

2.1 Description

Scout在Boo的盒子中找到了一条项链，这条项链由 n 个珍珠构成，第 i 颗的重量为 w_i 。项链是环形的，在项链上第 i 颗珍珠和 $(i \bmod n) + 1$ 相邻。

Scout希望将项链从若干个位置切开，将切成的每一份分别藏在不同的地方，并保证每一份的重量之和均不超过 m 。

现在Scout会给出 q 次询问，每次给出一个 m ，你需要回答对于每个 m ，至少要将项链切成多少份，才能使每一份的重量之和均不超过 m 。

2.2 Input

从文件`necklace.in`中读入数据。

第一行两个整数 n, q 。

第二行 n 个整数表示 w_i 。

接下来 q 行，每行一个整数 m 表示询问。

2.3 Output

输出到文件`necklace.out`中。

输出 q 行，按顺序给出每一个询问的答案。

2.4 Sample1

2.4.1 Input

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

2.4.2 Output

```
2
4
3
```

2.4.3 Explanation

对于第一个询问，一种划分方案为 $2)(421)(32$.

对于第二个询问，一种划分方案为 $2)(4)(21)(3)(2$.

对于第三个询问，一种划分方案为 $(24)(213)(2)$.

2.5 Sample2

见选手目录下的`necklace/necklace2.in`与`necklace/necklace2.ans`.

2.6 Subtasks

对于所有数据，有 $1 \leq n \leq 10^6, 1 \leq q \leq 50, 0 \leq w_i \leq 10^9, \max\{w_i\} \leq m \leq 10^{15}$.

- Subtask1(22%), $n \leq 15$.
- Subtask2(21%), $n \leq 10^3$.
- Subtask3(34%), $n \leq 5 \times 10^4$.
- Subtask4(23%), 没有特殊的约束.

3 Polygon

3.1 Description

Amir有一个 $n \times n$ 的矩阵 A ，初始时每个位置都是0；他还有一个 k 条边的简单多边形（即除了相邻的边之外，不会有两条边相交），满足所有的边都与坐标轴平行，且所有顶点的坐标都是非负整数。

对于任意 $x, y \geq 0$ ，如果以 $(x, y), (x+1, y), (x, y+1), (x+1, y+1)$ 为顶点的正方形在多边形内部，那么设 $f(x, y) = 1$ ，否则 $f(x, y) = 0$ ；而 $g(x, y) = \sum_{i=0}^x \sum_{j=0}^y f(i, j)$ 。

现在Amir会进行 q 次操作，有两种类型：

- 1 a b 对所有 $a \leq x \leq n, b \leq y \leq n$ ，令 $A[x][y] = A[x][y] + (a+b) \times f(x-a, y-b)$ 。
- 2 a b 对所有 $a \leq x \leq n, b \leq y \leq n$ ，令 $A[x][y] = A[x][y] + (a+b) \times g(x-a, y-b)$ 。

方便起见，所有操作在模 2^{32} 意义下进行。

Amir想考考Hassan所有操作结束后矩阵中所有元素的异或和是多少，但他自己也算不出结果，于是这个问题就交给你了。

3.2 Input

从文件`polygon.in`中读入数据。

第一行三个整数 n, k, q 。

接下来 k 行，每行两个整数 X_i, Y_i ，逆时针给出多边形的顶点坐标。

接下来 q 行，每行三个整数描述一次操作。

3.3 Output

输出到文件`polygon.out`中。

输出一个整数表示答案。

3.4 Sample1

3.4.1 Input

```
3 4 2
0 0
2 0
2 2
0 2
```

```
1 1 1
2 2 2
```

3.4.2 Output

20

3.4.3 Explanation

$f(i, j)$ 构成的矩阵:

```
: : :
0 0 0 ...
1 1 0 ...
1 1 0 ...
```

$g(i, j)$ 构成的矩阵:

```
: : :
2 4 4 ...
2 4 4 ...
1 2 2 ...
```

第一次操作后, A 矩阵为: (左下角为 $A[1][1]$)

```
0 0 0
2 2 0
2 2 0
```

第二次操作后:

```
0 8 16
2 6 8
2 2 0
```

求异或和, 得到答案为20。

3.5 Sample2

见选手目录下的`polygon/polygon2.in`与`polygon/polygon2.ans`。

3.6 Sample3

见选手目录下的 *polygon/polygon3.in* 与 *polygon/polygon3.ans*。

3.7 Subtasks

对于所有数据，满足 $1 \leq n \leq 5 \times 10^3, 0 \leq q \leq 2 \times 10^5, 4 \leq k \leq 200, 0 \leq X_i, Y_i \leq n, 1 \leq a, b \leq n$ 。

数据保证多边形不自交，且所有内角均不为 π (即内角均不为平角)。

本题共20个测试点，每个测试点5分。各个测试点还满足如下约束：

测试点编号	k	n	q	操作种类
1	= 4	≤ 100	= 0	1
2			≤ 100	1, 2
3				1
4		$\leq 5 \times 10^3$	$\leq 5 \times 10^3$	1, 2
5			$\leq 2 \times 10^5$	1
6				1, 2
7				1
8				1, 2
9	≤ 40	≤ 100	$\leq 10^3$	1
10				1, 2
11		$\leq 5 \times 10^3$	$\leq 2 \times 10^5$	1
12				1
13				1, 2
14				1
15				1, 2
16				1
17				1, 2
18	≤ 200	$\leq 5 \times 10^3$	$\leq 2 \times 10^5$	1
19				1
20				1, 2