

NOIP2019 模拟赛

Tea

2019 年 8 月 5 日

| | | | |
|---------|--------------|----------|------------|
| 题目名称 | 序列 | 箱子 | 旅行 |
| 题目类型 | 传统型 | 传统型 | 传统型 |
| 目录 | sequence | case | travel |
| 源文件名 | sequence.cpp | case.cpp | travel.cpp |
| 输入文件名 | sequence.in | case.in | travel.in |
| 输出文件名 | sequence.out | case.out | travel.out |
| 每个测试点时限 | 1s | 1s | 1s |
| 内存限制 | 128MB | 128MB | 128MB |
| 是否有下发样例 | 是 | 是 | 是 |
| 是否开启 O2 | 是 | 否 | 否 |

注意事项:

1. 评测时开启无限栈与 c++11。
2. 评测将在 linux 环境下用 LemonPlus 进行。
3. 如发现原题，请不要大叫”这不是 XXX 上的原题吗!”。
4. AK 后可以离场，不要大声喧哗。
5. Best wishes!

序列 (sequence)

【题目描述】

小 x 是一个喜欢摆弄序列的孩子。

小 x 有一个由二元组构成的长度为 n 的序列 A (n 为偶数),

其中 $A_i = (x_i, y_i)$, 其中 $x_i, y_i \in \{0, 1\}$

他现在想通过若干次交换序列中的两个元素, 使得

$$\sum_{i=1}^{\frac{n}{2}} x_i = \sum_{i=\frac{n}{2}+1}^n y_i$$

他想知道任意一种构造方法, 但是由于序列太长所以他希望你能帮他解决这个问题。

【输入格式】

从文件 *sequence.in* 中读入数据。

第一行一个正整数 n , 表示序列长度。

接下来 n 行, 每行两个数 x, y , 表示 A 序列。

【输出格式】

输出到文件 *sequence.out* 中。

输出 n 个正整数, 每两个数字用空格隔开, 第 i 个数表示构造的序列中第 i 项对应的是 A 的哪一项。

如果有多组解, 输出任意一组。

无解输出-1。

【输入输出样例 1】

| sequence.in | sequence.out |
|-------------|--------------|
| 4 | 1 4 2 3 |
| 0 0 | |
| 0 1 | |
| 1 0 | |
| 1 1 | |

【样例 1 解释】

原序列为 (0, 0), (0, 1), (1, 0), (1, 1),

新序列为 (0, 0), (1, 1), (0, 1), (1, 0)。

等式两边都是 1。

【样例 2】

见下发文件。

【数据范围与提示】

| 子任务 | 分值 | 限制 |
|-----|----|------------------------|
| 1 | 10 | $n \leq 16$ |
| 2 | 30 | $n \leq 5000$ |
| 3 | 60 | $n \leq 3 \times 10^6$ |

对于 100% 的数据， $1 \leq n \leq 3 \times 10^6$

注意：本题读入输出规模较大，请使用较快的读入与输出方法。

箱子 (case)

【题目描述】

小 h 是个有收集癖的孩子。

一天他回到家，发现各种物品排成了一列，每个物品都有一个重量 w_i 和高度 h_i 。

他觉得这一列太长了，于是打算用若干个箱子把它们按从左到右的顺序装起来。

每个箱子的最大装载量为 W ，即只能装重量总和不超过 W 的物品。

箱子的高度就是其中物品高度的最大值。

箱子的重量就是其中物品重量的总和。

用箱子当然是要有费用的，第 i 个箱子的费用为当前用的箱子的个数 (包括 i) 乘以这个箱子的重量再加上这个箱子的高度。

它希望最小化费用，你能帮帮他吗？

【输入格式】

从文件 *case.in* 中读入数据。

第一行为两个整数 n, W ，表示物品数量和箱子的最大载重。

接下来 n 行，每行两个整数 h_i 与 w_i ，表示物品的高度与重量。

【输出格式】

输出到文件 *case.out* 中。

第一行，一个整数，表示最小花费。

【输入输出样例 1】

| case.in | case.out |
|---------|----------|
| 5 10 | 70 |
| 5 7 | |
| 9 2 | |
| 8 5 | |
| 13 2 | |
| 3 8 | |

【样例 1 解释】

分 $[1]$, $[2, 3, 4]$, $[5]$ 装进 3 个箱子。

费用分别为 $7 \times 1 + 5$, $9 \times 2 + 13$, $8 \times 3 + 3$ 。

和为 70。

【样例 2】

见下发文件。

【数据范围与提示】

| 子任务 | 分值 | 限制 |
|-----|----|---------------|
| 1 | 10 | $n \leq 10$ |
| 2 | 20 | $n \leq 100$ |
| 3 | 30 | $n \leq 5000$ |
| 4 | 40 | $n \leq 10^5$ |

对于 100% 的数据, $1 \leq n \leq 10^5, 0 \leq w_i \leq W \leq 10^9, 0 \leq h_i \leq 10^6$

旅行 (travel)

【题目描述】

小 b 是个爱好旅行的孩子。

他尤其喜欢在观光公交上一边旅行，一边欣赏沿途的风景。

乘坐观光公交可以从一个城市旅行到另一个城市，并且可以欣赏沿途的风景，**观光公交的线路既有单向的也有双向的。**

这天他来到了 Q 省，发现这个省的风光甚好。他现在在省会城市 1，**他希望欣赏完这个省所有线路上的风景最后回到省会 1，并且他不希望重复欣赏一条线路的风景。**

值得注意的是：对于一条双向的线路，视作来回的风景是相同的。

你能告诉他这是否可能吗。

【输入格式】

从文件 *travel.in* 中读入数据。

第一行两个整数 T, type 表示数据组数与测试点类型, type=0 表示无特殊限制, **type=1 表示所有线路是双向的，type=2 表示所有线路是单向的。**

每组数据的第一行两个整数 n, m 表示城市最大编号，观光公交线路个数。

接下来 m 行，每行两个整数 u, v，一个字符串 dir，表示城市 u 到城市 v 有一条线路，**且 dir='D' 表示为单向，dir='U' 表示为双向。**

【输出格式】

输出到文件 *travel.out* 中。

对每一组数据，若能满足小 b 的需求，则输出"YES"，否则输出"NO"。

【输入输出样例 1】

| travel.in | travel.out |
|-----------|------------|
| 2 0 | YES |
| 5 6 | NO |
| 1 2 D | |
| 1 3 D | |
| 1 5 U | |
| 2 4 D | |
| 3 5 D | |
| 4 1 U | |
| 3 3 | |
| 1 2 D | |
| 2 3 U | |
| 1 3 D | |

【样例 2-4】

见下发文件。

【数据范围与提示】

| 子任务 | 分值 | 限制 |
|-----|----|--|
| 1 | 20 | $m \leq 10$ |
| 2 | 20 | $type \in \{1, 2\}$ |
| 3 | 60 | $n \leq 1000, m \leq \min\{6500, \frac{n(n-1)}{2}\}$ |

对于 100% 的数据, $1 \leq n \leq 1000, m \leq \min\{6500, \frac{n(n-1)}{2}\}, type \in \{0, 1, 2\}, T \leq 50$ 。

有些城市可能没有路线经过。

保证数据随机。