

第一届你要当原皮杯算法大赛第一轮

The 1st NIOP Round 1

Little H's Travel Notes of Silver Tomb

Mine_King

时间：2024 年 11 月 23 日 7:40 ~ 12:00

题目名称	旅游高手	重复	桜树街道	朱雀湖
题目类型	传统型	传统型	传统型	传统型
目录	expert	repeat	sakura	lake
可执行文件名	expert	repeat	sakura	lake
输入文件名	expert.in	repeat.in	sakura.in	llake.in
输出文件名	expert.out	repeat.out	sakura.out	lake.out
每个测试点时限	1.0 秒	1.0 秒	1.0 秒	4.0 秒
内存限制	512 MiB	512 MiB	512 MiB	512 MiB
子任务数目	3	20	11	10
子任务是否等分	否	否	否	是

提交源程序文件名

对于 C++ 语言	expert.cpp	sakura.cpp	lake.cpp	lovers.cpp
-----------	------------	------------	----------	------------

编译选项

对于 C++ 语言	-std=c++14 -lm -O2
-----------	--------------------

注意事项（请仔细阅读）

1. 文件名（包括程序名和输入输出文件名）必须使用英文小写。
2. C++ 中函数 main() 的返回值类型必须是 int，值必须为 0。
3. 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。
4. 若无特殊说明，输入文件中同一行内的多个整数、浮点数、字符串等均使用一个空格进行分隔。
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. 在终端下可使用命令 `ulimit -s unlimited` 将栈空间限制放大，但你使用的栈空间大小不应超过题目限制。
8. 评测在最新公布的 NOI Linux 下进行，使用 LemonLime 评测。
9. 由于某种原因，题目不一定按难度顺序排列。
10. 题目背景纯属虚构，无现实原型，如果你觉得有，那就是有。

旅游高手 (expert)

【题目背景】

由于一些未知原因，题目背景消失了。

【题目描述】

给定非负整数 n, k ，请你构造非负整数 x 和正整数 y ，使得：

$$x + y = n \wedge x \bmod y = k$$

特别地，如果不存在符合条件的 x, y ，请输出 **-1**。

【输入格式】

从文件 *expert.in* 中读入数据。

本题有多组数据。

第一行一个正整数 T 表示数据组数。

每组数据一行两个非负整数 n, k 。

【输出格式】

输出到文件 *expert.out* 中。

输出共 T ，第 i 行表示第 i 组数据的答案。

【样例 1 输入】

```
1 3
2 1 0
3 13 3
4 198818800000 122122200000
```

【样例 1 输出】

```
1 0 1
2 8 5
3 -1
```

【数据范围】

本题使用打包评测，只有通过一个 Subtask 的所有测试点后才能获得对应的分数。

子任务编号	$n \leq$	分值
1	10^6	20
2	10^{12}	40
3	10^{18}	40

对于 100% 的数据，保证 $1 \leq T \leq 5, 0 \leq n, k \leq 10^{18}$ 。

【提示】

你可以使用下发的 checker 来判断你的输出是否正确。使用方法为：

使用 `g++ checker.cpp -o checker -std=c++14 -O2` 编译 checker。

使用命令 `./checker <inputfile> <outputfile> <answerfile>` 检测答案是否正确。

重复 (repeat)

【题目描述】

小 S 在数轴上旅行。它从数轴上的原点出发，前往目的地—— E 这个数在数轴上的位置。每秒小 S 可以选择沿正方向前进 1 单位长度、沿负方向前进 1 单位长度或停留在原处。

在 $(0, E)$ 这段区间上，有 n 个景点是小 S 想参观的，其中第 i 个景点的坐标为 x_i 。当小 S 位于 x_i 时，小 S 可以一瞬间参观完第 i 个景点。

景色会随着天气及时间而变化。因此小 S 希望在第一次参观某个景点后至少间隔 T 秒再重新参观这个景点。当然，小 S 也可以在这个景点停留 T 秒，进行第二次参观后再离开。

小 S 希望你帮忙求出如果对每个景点都像这样参观两次，从 0 到 E 最少需要多少时间。

【输入格式】

从标准输入读入数据。

输入的第一行包含三个正整数 n, E, T ，分别表示景点数量、终点位置和参观间隔。

输入的第二行包含 n 个正整数 x_1, x_2, \dots, x_n ，分别表示每个景点的坐标。

【输出格式】

输出到标准输出。

输出一个正整数，表示最少需要的时间。

【样例 1 输入】

3 9 3

1 3 8

【样例 1 输出】

16

【样例 2】

见题目目录下的 *2.in* 与 *2.ans*。

【子任务】

对于 100% 数据，保证 $1 \leq n \leq 100000$, $1 \leq E, T \leq 10^9$, $0 < x_1 < x_2 < \cdots < x_n < E$ 。
所有测试数据的范围和特点如下表所示：

测试点编号	n	T
1	≤ 3	$= 1$
2		$\leq 10^9$
3	≤ 6	$= 3$
4		$\leq 10^9$
5	≤ 9	$= 1$
6	≤ 10	$\leq 10^9$
7	≤ 12	$= 3$
8	≤ 13	$\leq 10^9$
9	≤ 100	$= 1$
10		$\leq 10^9$
11	≤ 500	$= 3$
12		$\leq 10^9$
13	≤ 1000	
14	≤ 2000	
15	≤ 10000	$= 1$
16	≤ 50000	$\leq 10^9$
17		$= 3$
18	≤ 100000	$\leq 10^9$
19		
20		

桜树街道 (sakura)

【题目背景】

H 桑，故乡的撒库拉开了。

感觉不如外面的撒库拉。——小 H 在评价盆景园内的盆景时如是说道。

【题目描述】

众所周知撒库拉是树。因为小 H 喜欢撒库拉，爱屋及乌，小 H 也很喜欢树。因此小 H 想要请你做一道树上问题。

小 H 准备了 n 个烧杯，每个烧杯的高度都是 m 。烧杯 i 的杯底厚度为 h_i ，也就是说烧杯 i 只有 $m - h_i$ 的高度可以用来装水。

小 H 在第 i 个烧杯的 h_i 高度处打了一个洞，然后连接上一个大水管，大水管分出若干小水管，小水管再连接其他烧杯的洞。一共有 $n - 1$ 根小水管，将所有烧杯连在一起，因此烧杯和水管形成了一个树形结构。

现在小 H 要在每个烧杯中加一定量的水，如果烧杯 i 加了 a_i 高度的水，则水位就是 $h_i + a_i$ 。当两个被水管连接的烧杯水位不同时，水会从水位高的烧杯流向水位低的烧杯。形式化地，如果两个烧杯 i, j 被一根水管连接，且 $a_i > 0, h_i + a_i > h_j + a_j$ ，则水会从 i 流向 j 。

小 H 想要知道有多少种加水的方法，使得每个杯子的水位都是整数，且水不会流动。但是小 H 已经退役了，无法思考这个问题，所以这个问题就交给你了。

【输入格式】

从文件 `sakura.in` 中读入数据。

第一行两个整数 n, m 表示烧杯数量和烧杯高度。

接下来 $n - 1$ 行每行两个正整数 u_i, v_i 表示一根水管。

接下来一行 n 个整数 h_1, h_2, \dots, h_n 。

【输出格式】

输出到文件 `sakura.out` 中。

输出答案模 $10^9 + 7$ 的结果。

【样例 1 输入】

```
1 4 10
2 1 2
3 1 3
```

```
4 1 4
5 7 1 2 7
```

【样例 1 输出】

```
1 45
```

【样例 2】

见选手目录下的 *sakura/sakura2.in* 与 *sakura/sakura2.ans*。
此样例满足子任务 3 的限制。

【样例 3】

见选手目录下的 *sakura/sakura3.in* 与 *sakura/sakura3.ans*。
此样例满足子任务 5 的限制。

【数据范围】

本题使用打包评测，只有通过一个 Subtask 的所有测试点后才能获得对应的分数。

子任务编号	$n \leq$	$m \leq$	特殊性质	分值
1	4	10	无	10
2	10^6	10^6	$u_i = i, v_i = i + 1$	20
3	1000	1000	无	15
4		10^6		20
5	10^5			15
6	10^6			20

对于 100% 的数据，保证 $1 \leq n \leq 10^6, 0 \leq h_i \leq m \leq 10^6$ 。

朱雀湖 (lake)

【题目背景】

走两步就走两步，CityWalk 懂不懂。——小 H

小 H 毕竟还是旅游大佬，这次他找到了朱雀湖作为旅行的目的地。

朱雀湖毕竟是国家 2B 级景区，质量还是相当高的，但是这种公园类的景点毕竟还是略显无趣，于是小 H 就打算和 MK 玩一个游戏。

【题目描述】

小 H 会给 MK 一个长度为 n 的字符串 S 和一个长度为 m 的字符串 P ，两个字符串的字符集大小均为 k ，并且 P 中每个字符至多出现 2 次。MK 要在 S 中找到尽量多的互不相交的和 P 相同的子串。

这个游戏对 MK 来说太简单了，因此小 H 将游戏难度升级。设原本游戏的答案为 $f(S, P)$ ，MK 要回答所有满足条件的 S, P 的 $f(S, P)$ 的和模 $10^9 + 7$ 的结果。

小 H 承诺如果 MK 能够回答正确的答案就 v MK 50 元，但是 MK 太菜了不会做，于是请你来帮他。如果你成功帮助了 MK，他会给你 $50 \bmod 10$ 元作为感谢。

【输入格式】

从文件 *lake.in* 中读入数据。

一行三个正整数 n, m, k 。

【输出格式】

输出到文件 *lake.out* 中。

输出一行一个整数表示答案。

【样例 1 输入】

```
1 4 2 3
```

【样例 1 输出】

```
1 228
```

【样例 2 输入】

```
1 57 14 28
```


【样例 2 输出】

1 341760267

【样例 3 输入】

1 1919 810 514

【样例 3 输出】

1 643804262

【样例 4 输入】

1 999999 1999 12345678

【样例 4 输出】

1 52352722

【数据范围】

本题使用打包评测，只有通过一个 Subtask 的所有测试点后才能获得对应的分数。

子任务编号	$n \leq$	$m \leq$	$k \leq$	特殊性质	分值
1	5	5	3	无	5
2	10^6	3	10^9		10
3	100	100	10^6		7
4	2000	2000	10^9	$m \geq 1000$	9
5			10^9		11
6		100	10^6	无	7
7	2000	2000	10^9		9
8			10^9		11
9		100	10^6		9
10	10^6	2000	10^6		10
11			10^9		12

对于 100% 的数据，保证 $1 \leq n \leq 10^6, 1 \leq m \leq 2000, m \leq n, 1 \leq k \leq 10^9$ 。