

2022 信息学国庆集训 B 班

Day2

时间：2022 年 10 月 3 日 08:00 ~ 12:30

题目名称	三角形	打字	序列	高铁
题目类型	非交互式程序题	非交互式程序题	非交互式程序题	非交互式程序题
目录	tri	type	seq	train
可执行文件名	tri	type	seq	train
输入文件名	tri.in	type.in	seq.in	train.in
输出文件名	tri.out	type.out	seq.out	train.out
每个测试点时限	1.0 秒	1.0 秒	1.0 秒	1.0 秒
内存限制	256 MB	256 MB	256 MB	256 MB
测试点数目	10	10	10	10
测试点是否等分	是	是	是	是

提交源程序文件名

对于 C++ 语言	tri.cpp	type.cpp	seq.cpp	train.cpp
-----------	---------	----------	---------	-----------

编译选项

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

【注意事项（请仔细阅读）】

1. 文件名（程序名和输入输出文件名）必须使用英文小写。
2. C/C++ 中函数 `main()` 的返回值类型必须是 `int`，程序正常结束时的返回值必须是 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，结果比较方式为忽略行末空格、文末回车后的全文比较。
5. 程序可使用的栈空间大小与该题内存空间限制一致。
6. 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。
7. 每道题目所提交的代码文件大小限制为 100KB。
8. 若无特殊说明，输入文件与输出文件中同一行的相邻整数均使用一个空格分隔。
9. 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。

10. 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。
11. 使用 `std::deque` 等 STL 容器时，请注意其内存空间消耗。
12. 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。

三角形 (tri)

【题目描述】

小 D 是华清小学的一名学生, 在这天的数学课上他学到了一个新知识: 三条边能够构成一个三角形当且仅当任意两条边的长度和都严格大于第三条边。为了在实践中更好掌握知识, 这天他捡来了三根长度分别为 a, b, c 的树枝。此外他还有 m 个长度为 1 的小树枝, 他可以往每根木棍上都接上一些树枝 (也可以不接), 每个小树枝会让接上的木棍的长度增加 1。他想知道用自己仅有的这些小树枝, 有多少种方案可以让三根木棍最终可以拼成一个面积大于 0 的三角形。两种方案不同, 当且仅当存在一根木棍最终长度不同。

注意: 捡来的三根树枝可能拥有相同的长度, 但仍被认为是不同的。而小树枝则全都被认为是相同的。

【输入格式】

从文件 *tri.in* 中读入数据。

输入数据只包括一行四个非负整数 a, b, c, m , 含义如题面所述。

【输出格式】

输出到文件 *tri.out* 中。

输出仅包括一行一个整数, 表示方案数。注意, 这个数字可能超过 32 位整形的表示范围。

【样例输入 1】

```
1 1 1 1 2
```

【样例输出 1】

```
1 4
```

【样例 2】

见选手目录下的 *tri/tri2.in* 与 *tri/tri2.ans*。

【测试点约束】

对于所有测试数据， $1 \leq a, b, c \leq 3 \cdot 10^5$ ， $0 \leq m \leq 3 \cdot 10^5$ 。

测试点编号	$a, b, c \leq$	$m \leq$
1, 2, 3	300	300
4, 5, 6	5000	5000
7, 8, 9, 10	300000	300000

打字 (type)

【题目描述】

小 Z 热爱打字。众所周知,有时候一些重复的内容可以通过复制粘贴来解决,小 Z 也熟练掌握了这个技巧。这天他熟练利用复制粘贴进行了一些操作,这些操作分为以下两种:

1 x 在文档末端输入一个数 x 。

2 $a\ b$ 复制文档的前 a 个数,并且在文档的最后粘贴 b 次。

小 Z 是在一个新的文档开始编辑的,所以一开始这个文档为空。在完成了所有操作之后,小 Z 想知道这个文档的一些位置的数字是什么?

【输入格式】

从文件 `type.in` 中读入数据。

输入数据第一行包括一个数字 n ,表示操作数。接下来 n 行,每行描述一个操作,格式如题目所述。接着一行一个数字 m ,表示询问次数。最后包含 m 行,每行一个数字 p_i ,表示这次询问第 p_i 个数。

【输出格式】

输出到文件 `type.out` 中。

对于每个数据输出一行一个数字,表示答案。

【样例输入 1】

```
1 4
2 1 1
3 1 2
4 2 2 2
5 2 5 2
6 4
7 1 2 12 16
```

【样例输出 1】

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

【样例解释 1】

最终文档内容为 1 2 1 2 1 2 1 2 1 1 2 1 2 1。

【样例 2】

见选手目录下的 *seq/seq2.in* 与 *seq/seq2.ans*。

【测试点约束】

用 N 表示文档大小。对于所有的数据，满足 $1 \leq n, m \leq 10^5$, $1 \leq x, a \leq 10^5$, $1 \leq b \leq 10^4$ 。

测试点编号	$N \leq$	特殊性质
1, 2, 3	10^7	
4, 5	10^{18}	A
6, 7	10^{18}	B
8, 9, 10	10^{18}	

特殊性质 A: 复制操作满足 $b = 1$ 。

特殊性质 B: 满足复制粘贴操作总是在输入操作之后。

保证输入数据合法，1 操作输入的数字在 $[1, 10^9]$ 之内。

序列 (seq)

【题目描述】

最长不下降子序列是一个经典问题。给定一个序列 $x_1x_2\dots x_n$, 你需要找到一个最大的 k , 使得我们能从原序列中按顺序选出 k 个元素 $x_{i_1}x_{i_2}\dots x_{i_k}$ 满足 $1 \leq i_1 < i_2 < \dots < i_k \leq n$ 且 $x_{i_1} \leq x_{i_2} \leq \dots \leq x_{i_k}$ 。今天便是要求一个序列的最长不下降子序列长度, 只不过这个序列是由一个小序列重复多次形成的。

【输入格式】

从文件 `seq.in` 中读入数据。

输入数据第一行包括两个数字 n, T , 表示小序列长度和重复次数。接下来一行 n 个数字, 表示给你的小序列。需要求解的序列为这个小序列重复 T 次, 长度为 nT 。

【输出格式】

输出到文件 `seq.out` 中。

输出一个数字表示答案。

【样例输入 1】

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

【样例输出 1】

```
1 5
```

【样例解释 1】

原序列为 3,1,4,2,3,1,4,2,3,1,4,2, 一个满足条件的最长不下降子序列为 1,2,3,3,4 (存在其他一样长的子序列)。

【样例 2】

见选手目录下的 `seq/seq2.in` 与 `seq/seq2.ans`。

【测试点约束】

对于所有测试数据, $1 \leq n \leq 100, 1 \leq T \leq 10^7, 1 \leq a_i \leq 300$ 。

测试点编号	$n \leq$	$T \leq$	$a_i \leq$
1, 2, 3	100	$\lfloor \frac{10^5}{n} \rfloor$	300
4, 5	100	10^7	10
6, 7, 8, 9, 10	100	10^7	300

高铁 (train)

【题目描述】

猪猪国有 n 座城市（编号为 1 到 n ），其中首都的编号为 1。这些城市之间有 m 条普快火车线路，第 i 条道路连接 u_i, v_i 两个城市，所需时间为 t_i ，这些火车线路将所有 n 个城市连接在一起。去年猪猪国修建了一些连接首都的高铁来减少城市到首都的距离，但高昂的运营成本却让猪猪国王很是心疼，他决定暂停一些高铁线路的运营，但最好不要影响到通行时间。他想知道最多可以暂停多少高铁线路，满足首都到所有城市的最短时间（可以任意搭乘火车或高铁，不考虑换乘时间）和现在相同。

【输入格式】

从文件 `train.in` 中读入数据。

第一行读入三个正整数 n, m, k ，依次表示城市的数量，原有火车线路的数量和新建高铁线路的数量。

接下来 m 行，每行三个数字 u_i, v_i, t_i ，表示一条原有的火车线路，其中 u_i 和 v_i 是连接的两座城市， t_i 是通行时间。

最后 k 行，每行两个数字 s_i, w_i ，表示一条新建的高铁线路连接 1 号城市和 s_i 号城市且距离为 w_i 。注意，相同的城市可能有多条高铁线路，且有可能有多条火车线路连接相同的两个城市。

【输出格式】

输出到文件 `train.out` 中。

输出一个数字表示最多能暂停的高铁线路数量。

【样例输入 1】

```
1 3 2 2
2 1 2 1
3 2 3 1
4 2 2
5 3 1
```

【样例输出 1】

1 1

【样例解释 1】

到城市 2 的高铁并没有火车快，可以暂停，但城市 3 的高铁就不能暂停了，否则最短的时间会从 1 增加到 2。

【样例 2】

见选手目录下的 *train/train2.in* 与 *train/train2.ans*。

【测试点约束】

对于所有测试数据，满足 $1 \leq t_i, w_i \leq 10^9$, $1 \leq u_i, v_i \leq n$, $2 \leq s_i \leq n$ 。

测试点编号	$n \leq$	$m \leq$	$k \leq$
1, 2	100	100	100
3, 4	50000	200000	4
5, 6, 7, 8, 9, 10	50000	200000	50000