

IOI 2017 中国代表队选拔赛 暨全国青少年信息学奥林匹克精英赛

CCF CTSC 2017

第二试

时间：2017 年 5 月 10 日 08:30 ~ 13:30

题目名称	吉夫特	最长上升子序列	投影
题目类型	传统型	传统型	传统型
目录	gift	lis	projection
可执行文件名	gift	lis	projection
输入文件名	gift.in	lis.in	projection.in
输出文件名	gift.out	lis.out	projection.out
每个测试点时限	1.5 秒	1.0 秒	5.0 秒
内存限制	512 MB	512 MB	512 MB
测试点数目	20	20	10
每个测试点分值	5	5	10

提交源程序文件名

对于 C++ 语言	gift.cpp	lis.cpp	projection.cpp
对于 C 语言	gift.c	lis.c	projection.c
对于 Pascal 语言	gift.pas	lis.pas	projection.pas

编译选项

对于 C++ 语言	-O2 -lm	-O2 -lm	-O2 -lm
对于 C 语言	-O2 -lm	-O2 -lm	-O2 -lm
对于 Pascal 语言	-O2	-O2	-O2

吉夫特 (gift)

【问题描述】

简单的题目，既是礼物，也是毒药。

B 君设计了一道简单的题目，准备作为 gift 送给大家。

输入一个长度为 n 的数列 $a_1, a_2, \dots, a_n$ 问有多少个长度大于等于 2 的不上升的子序列 $a_{b_1}, a_{b_2}, \dots, a_{b_k}$ 满足

$$\prod_{i=2}^k \binom{a_{b_{i-1}}}{a_{b_i}} \bmod 2 = \binom{a_{b_1}}{a_{b_2}} \times \binom{a_{b_2}}{a_{b_3}} \times \dots \times \binom{a_{b_{k-1}}}{a_{b_k}} \bmod 2 > 0$$

输出这个个数对 1000000007 取模的结果。

G 君看到题目后，为大家解释了一些基本概念。

我们选择任意多个整数 b_i 满足

$$1 \leq b_1 < b_2 < \dots < b_{k-1} < b_k \leq n$$

我们称 $a_{b_1}, a_{b_2}, \dots, a_{b_k}$ 是 a 的一个子序列。

如果这个子序列同时还满足

$$a_{b_1} \geq a_{b_2} \geq \dots \geq a_{b_{k-1}} \geq a_{b_k}$$

我们称这个子序列是不上升的。

组合数 $\binom{n}{m}$ 是从 n 个互不相同的元素中取 m 个元素的方案数，具体计算方法如下：

$$\binom{n}{m} = \frac{n!}{m!(n-m)!} = \frac{n \times (n-1) \times \dots \times 2 \times 1}{(m \times (m-1) \times \dots \times 2 \times 1)((n-m) \times (n-m-1) \times \dots \times 2 \times 1)}$$

这里要特别注意，因为我们只考虑不上升子序列，所以在求组合数的过程中，一定满足 $n \geq m$ ，也就是 $\binom{a_{b_{i-1}}}{a_{b_i}}$ 中一定有 $a_{b_{i-1}} \geq a_{b_i}$ 。

我们在这里强调取模 $x \bmod y$ 的定义：

$$x \bmod y = x - \left\lfloor \frac{x}{y} \right\rfloor \times y$$

其中 $\lfloor n \rfloor$ 表示小于等于 n 的最大整数。

$x \bmod 2 > 0$ ，就是在说 x 是奇数。

与此同时，经验告诉我们一个长度为 n 的序列，子序列个数有 $O(2^n)$ 个，所以我们通过对答案取模来避免输出过大。

B 君觉得 G 君说的十分有道理，于是再次强调了这些基本概念。

最后，G 君听说这个题是作为 gift 送给大家，她有一句忠告。

“Vorsicht, Gift!”

“小心……剧毒！”

【输入格式】

从文件 *gift.in* 中读入数据。

第一行一个整数 n 。

接下来 n 行，每行一个整数，这 n 行中的第 i 行，表示 a_i 。

【输出格式】

输出到文件 *gift.out* 中。

一行一个整数表示答案。

【样例 1 输入】

4
15
7
3
1

【样例 1 输出】

11

【样例 2~9】

见选手目录下的 *gift/gift2~9.in* 与 *gift/gift2~9.ans*。

【数据范围与约定】

- 对于前 10% 的测试点， $n \leq 9, 1 \leq a_i \leq 13$;
- 对于前 20% 的测试点， $n \leq 17, 1 \leq a_i \leq 20$;
- 对于前 40% 的测试点， $n \leq 1911, 1 \leq a_i \leq 4000$;
- 对于前 70% 的测试点， $n \leq 2017$;
- 对于前 85% 的测试点， $n \leq 100084$;
- 对于 100% 的测试点， $1 \leq n \leq 211985, 1 \leq a_i \leq 233333$ 。所有的 a_i 互不相同，也就是说不存在 i, j 同时满足 $1 \leq i < j \leq n$ 和 $a_i = a_j$ 。

最长上升子序列 (lis)

【问题描述】

猪小侠最近学习了最长上升子序列的相关知识。对于一个整数序列 $A = (a_1, a_2, \dots, a_k)$ ，定义 A 的子序列为：从 A 中删除若干个元素后（允许不删，也允许将所有 k 个元素都删除），剩下的元素按照原来的顺序所组成的序列。如果这个子序列的元素从左到右严格递增，则称它为 A 的一个上升子序列。其中包含元素数量最多的上升子序列称为 A 的最长上升子序列。例如， $(2, 4, 5, 6)$ 和 $(1, 4, 5, 6)$ 都是 $(2, 1, 1, 4, 7, 5, 6)$ 的最长上升子序列，长度都为 4。

现在猪小侠遇到了这样一个问题：给定一个序列 $B_m = (b_1, b_2, \dots, b_m)$ ，设 C 是 B_m 的子序列，且 C 的最长上升子序列的长度不超过 k ，则 C 的长度最大能是多少？

猪小侠觉得这个问题太简单了，缺乏挑战，他决定提出一个更难的问题。于是他给了你这样一个序列 $B = (b_1, b_2, \dots, b_n)$ ，以及若干次询问。每次询问会给定两个整数 m 和 k ，你需要对于 B 序列的前 m 个元素构成的序列 $B_m = (b_1, b_2, \dots, b_m)$ 和 k 回答上述问题。

【输入格式】

从文件 *lis.in* 中读入数据。

第一行两个整数 n, q ，其中 n 是序列 B 的长度， q 是询问次数。

第二行是空格隔开的 n 个正整数 $b_1, b_2, \dots, b_n$ 。

接下来 q 行，其中第 i 行包含两个整数 m_i, k_i ，表示对 $m = m_i, k = k_i$ 进行询问。

【输出格式】

输出到文件 *lis.out* 中。

输出共 q 行，按顺序每行一个整数作为回答。

【样例 1 输入】

```
11 6
9 6 3 1 5 12 8 4 2 2 2
5 1
7 2
9 1
9 2
11 1
11 11
```

【样例 1 输出】

4
6
5
8
7
11

【样例 2】

见选手目录下的 *lis/lis2.in* 与 *lis/lis2.ans*。

【样例解释】

询问 1: 对于序列 (9,6,3,1,5), 可以选取子序列 (9,6,3,1), 它的最长上升子序列长度为 1。

询问 2: 对于序列 (9,6,3,1,5,12,8), 可以选取子序列 (9,6,3,1,12,8), 它的最长上升子序列长度为 2。

询问 3: 对于序列 (9,6,3,1,5,12,8,4,2), 可以选取子序列 (9,6,5,4,2), 它的最长上升子序列长度为 1。

询问 4: 对于序列 (9,6,3,1,5,12,8,4,2), 可以选取子序列 (9,6,3,1,12,8,4,2), 它的最长上升子序列长度为 2。

询问 5: 对于序列 (9,6,3,1,5,12,8,4,2,2,2), 可以选取子序列 (9,6,5,4,2,2,2), 它的最长上升子序列长度为 1。

询问 6: 对于序列 (9,6,3,1,5,12,8,4,2,2,2), 可以选取子序列 (9,6,3,1,5,12,8,4,2,2,2), 它的最长上升子序列长度为 3。

【数据规模】

测试点	n	约束
1	≤ 50000	$k_i = 1$
2	≤ 300	$k_i \leq 2$
3	≤ 3000	
4	≤ 50000	$b_i \leq 5$
5		$b_i \leq 8$
6	≤ 100	$m_i = n$
7		
8	≤ 800	$m_i = n, k_i \leq 10$
9	≤ 1500	
10	≤ 10000	$m_i = n, k_i \leq 30$
11	≤ 15000	$m_i = n, k_i \leq 40$
12	≤ 20000	$m_i = n, k_i \leq 50$
13		$k_i \geq 10000$
14	≤ 8000	$m_i = n$
15	≤ 25000	没有约束
16	≤ 40000	
17	≤ 45000	
18	≤ 50000	
19		
20		

对于 100% 的数据, $1 \leq n \leq 50000, 1 \leq b_i \leq 50000, 1 \leq q \leq 200000, 1 \leq k_i \leq m_i \leq n$ 。

投影 (projection)

【问题描述】

换一个角度看, 世界可能就不同。—— 小强

k 维空间中有 n 个黑点与 n 个白点。我们为每一个黑点确定一个互不相同的对应的白点, 这样一共有 $n!$ 种对应方法。我们定义这 n 个黑点与 n 个白点之间的“移动距离”为, 在所有的对应方法中, 对应的黑点与白点之间的 n 个欧几里德距离的和的最小值。

例如: 考虑一维中的三个黑点 $\{1, 5, 6\}$ 与三个白点 $\{2, 3, 4\}$, 那么它们之间的移动距离为: $|1-2| + |5-3| + |6-4| = 4$ 。你可以验证一下这确实是距离和最小的一种对应方法。

你得到了三维空间中的 n 个黑点与 n 个白点。你想把它们投影到一个 k ($1 \leq k \leq 2$) 维子空间上。一维子空间就是三维空间中的一条直线, 二维子空间则是三维空间中的一个平面。一个点在一个子空间中的投影点就是这个子空间中距离它最近的点。例如, $(0, 0, 0)$, $(1, 1, 0)$, $(1, 0, 0)$, $(0, 1, 0)$ 这四个点投影到 $x - y = 0, z = 0$ 这条直线上之后, 得到的投影点是 $(0, 0, 0)$, $(1, 1, 0)$, $(0.5, 0.5, 0)$, $(0.5, 0.5, 0)$ 。

你希望这 n 个黑点和 n 个白点投影到这个 k 维子空间之后的移动距离最大。请你计算这个最大值除以 n 。

【输入格式】

从文件 *projection.in* 中读入数据。

每个测试文件中可能有多个测试用例, 每个测试用例的格式如下:

第一行两个数 n 和 k , 表示点数以及降到的维度。

后面 $2n$ 行, 每行三个数, 表示点的坐标。

文件最后有一行包含两个数 $-1 - 1$ 。

【输出格式】

输出到文件 *projection.out* 中。

对文件中的每个测试用例, 输出一行一个数 (长度不要超过 30), 表示结果。

你的答案与参考答案的相对误差不超过 10^{-7} 时被认为是正确的。

只有一个文件中所有的测试用例的结果都是正确的才能获得这个测试文件所对应的分数。

【样例 1 输入】

```
2 1
0 0 0
1 1 0
```

```
0 1 0
1 0 0
-1 -1
```

【样例 1 输出】

0.70710678118655

【样例 2 输入】

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

【样例 2 输出】

0.73883404559321

【样例 3】

见选手目录下的 *projection/projection3.in* 与 *projection/projection3.ans*。

【数据规模】

对于 30% 的数据, $k = 1$, $n \leq 1000$, 所有的点的 z 都是 0。一个文件中最多包含一个测试用例。

对于另外 40% 的数据, $k = 1$, $n \leq 1000$, 所有的点的坐标都是 $[-1, 1]$ 之间均匀独立随机生成的, 一个文件中最多包含十个测试用例。

对于另外 30% 的数据, $k = 2$, $n \leq 20$, 所有的点的坐标都是 $[-1, 1]$ 之间均匀独立随机生成的, 一个文件中最多包含十个测试用例。

对于 100% 的数据, 点的坐标的范围都是 -1 到 $+1$ 之间的, 答案不小于 0.01。