

NOIP2024 模拟赛

时间：2024 年 4 月 20 日

题目名称	烧结核凝晶	斩荆辟路	阵·大群	大群·树
题目类型	传统题	传统题	传统题	传统题
目录	salt	card	puzzle	tree
可执行文件名	salt	card	puzzle	tree
输入文件名	salt.in	card.in	puzzle.in	tree.in
输出文件名	salt.out	card.out	puzzle.out	tree.out
每个测试点时限	1 秒	1 秒	2 秒	1 秒
内存限制	512 MB	512 MB	512 MB	512 MB
子任务数目	10	20	20	20
测试点是否等分	是	是	是	是

提交源程序文件名

对于 C++ 语言	salt.cpp	card.cpp	puzzle.cpp	tree.cpp
-----------	----------	----------	------------	----------

编译选项

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

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. 提交的程序代码文件的放置位置请参考各省的具体要求。
4. 因违反以上三点而出现的错误或问题，申诉时一律不予受理。
5. 若无特殊说明，结果的比较方式为全文比较（过滤行末空格及文末回车）。
6. 选手提交的程序源文件必须不大于 100KB。
7. 程序可使用的栈空间内存限制与题目的内存限制一致。
8. 只提供 Linux 格式附加样例文件。
9. 评测在当前最新公布的 NOI Linux 下进行，各语言的编译器版本以此为准。

烧结核凝晶 (salt)

【题目描述】

斯宝在维多利亚发现了一种新的物质——转质盐。

巨船的工匠发现转质盐可以制作一种名叫烧结核凝晶的材料，工匠们已经试制出了烧结核凝晶，可是他们不小心丢了一种关键合成物——转质盐聚块的合成配方，斯宝希望你能帮忙找回如何制作它的方法。

烧结核凝晶是一种特定高温环境下具有分子识别能力的材料。作为助剂能选择性地吸附源石，为精密加工源石材料、降低源石器件耗能提供新可能。

具体来说，转质盐聚块由糖组、半自然溶剂、转质盐组三种材料合成，工匠们先需要将糖组溶化至半自然溶剂中，再与打碎的转质盐组进行混合搅拌、高温熔化、塑型、冷却而成。工匠们丢失的配方记载的是将糖组溶化至半自然溶剂这一过程的配方比，但工匠们找到了当时的试验记录残破的一部分：

- 我们配置了 $1 \leq n \leq 10^5$ 份糖组与半自然溶剂的溶液组，其中第 i 组溶液中糖组的质量为 a_i ，整份溶液的质量为 b_i ；
- 由前面的实验可以得到，选 $1 \leq k \leq n$ 份溶液合成时成功率最大，并且糖的浓度越高，最后得到的转质盐聚块越稳定；
- 一位炎国的冶金学家用她特殊的源石技艺选出了这 k 份材料，让这 k 份材料混合后的 糖的质量分数取到最大值；

虽然这位冶金学家已经暂时离开了斯宝的船，但工匠们幸运地找到了当时实验的数据记录表，工匠们现在只需要知道 糖占整个溶液的质量分数，并且误差不超过 10^{-5} ，就能成功还原配方了。

斯宝希望你能在这位炎国的女士回归巨船前，帮工匠们还原配方。

【输入格式】

从文件 `salt.in` 中读入数据。

第 1 行两个整数 n, k 分别表示溶液份数，需要挑选出的溶液份数；

第 2 ~ $n + 1$ 行每行两个整数 $0 \leq a_i \leq b_i \leq 10^9$ ($b_i \geq 1$)，表示这份溶液的质量为 b_i ，其中糖的质量为 a_i 。

【输出格式】

输出到文件 `salt.out` 中。

一行一个非负实数，表示糖的质量分数的最大值。

【样例 1 输入】

```
1 5 3
2 3 9
3 1 3
4 8 9
5 1 1
6 6 8
```

【样例 1 输出】

```
1 0.8333333333333333
```

【样例 2 输入】

```
1 16 9
2 75579373 165397708
3 34131912 133724248
4 136529561 513645661
5 156095204 438431217
6 5436787 9820920
7 762299 142970462
8 83095223 391910866
9 664618010 798202849
10 95338462 321730008
11 71068417 220199418
12 337854578 925730526
13 560355959 991623378
14 463888355 794508044
15 33385952 240207110
16 53340465 182203667
17 159928732 352382098
```

【样例 2 输出】

1 0.572454037538333

【数据范围】

测试点编号	n	k
1	≤ 100000	$= 1$
2 ~ 3	≤ 20	≤ 20
4 ~ 5	≤ 1000	≤ 1000
6 ~ 10	$\leq 10^5$	$\leq 10^5$

斩荆辟路 (card)

【题目描述】

斯宝正在开船，突然海面上出现了一群海怪，他们挥舞着触须，开始进攻斯宝的船，斯宝紧急召集了他的船员抵抗海怪入侵。

斯宝发现，这些海怪有了一种特殊生物的领导后，可以使船员们受到剧烈的神经损伤，斯宝称其为海嗣。

在战斗中，斯宝注意到 6 星干员可以很好地猎杀海嗣，但是这次出海斯宝没有携带足够的干员，于是他联系了 k ，并使用他的电脑开始在卡池里抽 (kai) 卡 (gua)。

具体来说， k 的电脑修改了保底机制，让每一次抽卡的概率都产生了变化，规则是：如果前面抽了连续 i 次卡没有出 6 星干员，那么这次抽卡抽出 6 星干员的概率为 p_i ($0 \leq i \leq k \leq 100$)，但是不变的是第 $k+1$ 次抽卡必定抽出 6 星干员 (即 $p_k = 1$)。

为了保护他的船，斯宝决定把所有资源都投入卡池中，一共能抽 $0 \leq n \leq 2 \times 10^7$ 次卡。斯宝想知道，他期望能寻访到几个 6 星干员？

由于斯宝的 **不稳定 AI** 已经快速计算出了答案，你只需要验证 AI 的答案，具体的说，设期望寻访到 $\frac{p}{q}$ 个 6 星干员 ($\gcd(p, q) = 1$)，你只要输出一个数 $0 \leq ans < 998244353$ ，满足 $q \cdot ans \equiv p \pmod{998244353}$ 。

【输入格式】

从文件 *card.in* 中读入数据。

第 1 行两个整数 n, k 分别表示抽卡次数，连续抽不出 6 星干员的最多次数；

第 2 ~ $k+1$ 行每行两个整数 $0 \leq a_i < b_i < 998244353$ ，表示 $p_i = \frac{a_i}{b_i}$ ($0 \leq i < k$)。

【输出格式】

输出到文件 *card.out* 中。

一行一个非负整数 ans 。

【样例 1 输入】

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

【样例 1 输出】

1766352191

【样例 2】

见选手目录下的 `card/card2.in` 与 `card/card2.ans`。

【样例 3】

见选手目录下的 `card/card3.in` 与 `card/card3.ans`。

【样例 4】

见选手目录下的 `card/card4.in` 与 `card/card4.ans`。

【数据范围】

对于所有数据， $0 \leq n \leq 2 \times 10^7$ ， $0 \leq k \leq 100$ ， $0 \leq a_i < b_i < 998244353$ 。

测试点编号	n	k
1 ~ 4	≤ 10	≤ 5
5 ~ 8	$\leq 10^5$	≤ 100
9	$\leq 2 \times 10^7$	$= 0$
10 ~ 11	$\leq 2 \times 10^7$	$= 1$
12 ~ 15	$\leq 2 \times 10^7$	≤ 50
16 ~ 20	$\leq 2 \times 10^7$	≤ 100

温馨提示：图的总点数可能达到 10^6 。

阵·大群 (puzzle)

【题目描述】

经历了一番艰苦卓绝的战斗，斯宝终于驾船来到了阿戈尔的边缘，然而，一团朦胧的雾笼罩着海面，逐渐淹没了斯宝的巨船。

时间。时间。时间是它们的养分。深蓝的枝叶在无从计量的时间里缱绻生长，生命因此而摇曳舒展。这片区域容纳了一切，孕育了所有。

船上的科学家意识到这是大群的阵法，用来检验海怪或海嗣的身份，他们此前通过研究海嗣发现了他们经常接触两种物质，这两种神秘物质的粉末依附于海怪的触角皮肤中的褶皱上，科学家在海底探测到了 $n + k$ 个礁石，沾有两种物质的礁石分别由 n 个 A 物质、 k 个 B 物质。

在研究了一段时间后，科学家发现每个礁石上都印有神秘符文代表一个整数，第 i 个 A 物质礁石上刻着 a_i ，第 i 个 B 物质礁石上刻着 b_i ，为了解开大群的阵法，你需要按顺序触动每个 B 物质礁石，每次触动时对每个 A 物质礁石选择是否激活阵法，如果激活了，那么这个 A 物质礁石上的数会异或上 b_i 。

斯宝的科学家保证，当触动完 B 物质礁石后，如果 A 物质礁石上的数的和处于能操作出的最大值，那么阵法解开，否则你将面临大群的追杀。

形式化题意：

有 n 个整数 $a_i (0 \leq a_i < 2^m)$ ，你要执行 k 次操作，每次操作你可以选择任意一个整数 $b_i (0 \leq b_i < 2^m)$ ，对于每个 $i (1 \leq i \leq n)$ ，你可以选择将 a_i 变成 $a_i \oplus b_i$ 也可以选择保持不变。

请求出 k 次操作后 $\sum_{i=1}^n a_i$ 的最大值。

【输入格式】

从文件 `puzzle.in` 中读入数据。

第一行三个正整数 n, m, k 。

第二行一行 n 个整数，第 i 个整数为 a_i 。

【输出格式】

输出到文件 `puzzle.out` 中。

输出一行一个整数表示答案。

【样例 1 输入】

```
1 3 3 2
2 0 7 1
```

【样例 1 输出】

```
1 21
```

【样例 2 输入】

```
1 6 6 3
2 13 57 41 15 50 10
```

【样例 2 输出】

```
1 365
```

【样例 3】

见选手目录下的 *puzzle/puzzle3.in* 与 *puzzle/puzzle3.ans*。

【样例 4】

见选手目录下的 *puzzle/puzzle4.in* 与 *puzzle/puzzle4.ans*。

【数据范围】

对于 100% 的数据， $1 \leq n, m \leq 120$ ， $1 \leq k \leq 5$ 。

- 对于 5% 的数据， $1 \leq m \leq 10$ ；
- 对于另外 5% 的数据， $1 \leq n \leq 10$ ；
- 对于另外 10% 的数据， $1 \leq n, m \leq 60$ ， $k = 1$ ；
- 对于另外 20% 的数据， $1 \leq n, m \leq 60$ ， $k = 2$ ；
- 对于另外 20% 的数据， $1 \leq n, m \leq 60$ ， $k = 3$ ；
- 对于另外 20% 的数据， $k = 4$ ；
- 对于另外 20% 的数据， $k = 5$ 。

大群·树 (tree)

【题目描述】

斯宝突破了大群的封锁，登录了大群的一个小岛据点。

天色阴沉了下来，那些美丽而危险的、发光的潮水在此刻逐渐清晰，它们悄然蔓延到了岛上的岩洞口，我们的脚边。

干员们在小岛的岩洞口里发现了一个树状洞穴结构，斯宝决定派三个高级资深干员前往洞穴中侦察。

巨船上有一个基于源石技艺的传送法阵，可以将三名干员传送至洞穴的三个节点中。可是，斯宝的源石工匠发现岩洞内有剧烈的干扰源，巨船上的通信装置无法和岩洞里通讯。

船上的每个干员都有一个便携式通讯器，经测算，它们能在距离小于等于 k 的时候互相通讯（通讯的距离就是两个节点在树上的最短路径的边数）。由于通讯器没有中继装置，要求三个干员的通讯器两两距离不超过 k 才能互相通讯。

现在，巨船上的雷达以扫描出洞穴的结构，斯宝想知道，有多少个无序节点三元组满足如果干员传送至这三个不同的节点上，且他们能互相通信？

形式化题意：

一棵树，选 3 个不同点，两两之间距离小于等于 k 的方案数（距离是最短路径的边数）。

【输入格式】

从文件 *tree.in* 中读入数据。

第一行两个整数 n, k ，表示洞穴节点个数和通讯器能通讯的最大距离 k 。

后面 $n - 1$ 行，表示每条通道连了哪两个节点。

【输出格式】

输出到文件 *tree.out* 中。

一行一个整数，表示传送三个干员的方案数。

【样例 1 输入】

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

4	2 4
5	2 5

【样例 1 输出】

1	5
---	---

【样例 1 解释】

有 $(1, 2, 3), (1, 2, 4), (1, 2, 5), (1, 4, 5), (2, 4, 5)$ 共 5 种。

【样例 2】

见选手目录下的 *tree/tree2.in* 与 *tree/tree2.ans*。

【样例 3】

见选手目录下的 *tree/tree3.in* 与 *tree/tree3.ans*。

【数据范围】

对于 100% 的数据，满足 $3 \leq n \leq 10^5, 1 \leq k \leq 10^5$ 。

- 共 20 个测试点，每个点 5 分；
- 对于 20% 的数据，满足 $n \leq 100$ ；
- 对于 40% 的数据，满足 $n \leq 400$ ；
- 对于另外 20% 的数据，满足 $ans \leq 10^6$ ，其中 *ans* 是最后的答案，也就是你的输出。