

NOIP 2020 模拟赛

p_b_p_b

day2

一、题目概况

题目名称	马卡龙	长者	独白	吃瓜
题目类型	传统型	传统型	传统型	传统型
可执行文件名	macaron	genius	monologue	gourds
输入文件名	macaron.in	genius.in	monologue.in	gourds.in
输出文件名	macaron.out	genius.out	monologue.out	gourds.out
每个测试点时限	1.5 秒	1.0 秒	1.0 秒	1.0 秒
内存限制	512 MiB	256 MiB	256 MiB	256 MiB
子任务数目	6	2	5	7
测试点是否等分	否	否	否	否

二、提交源程序文件名

对于 C++ 语言	macaron.cpp	genius.cpp	monologue.cpp	gourds.cpp
对于 C 语言	macaron.c	genius.c	monologue.c	gourds.c

三、编译选项

对于 C++ 语言	-lm -O2 -std=c++17
对于 C 语言	-lm -O2

四、注意事项

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() 的返回类型必须是 int，程序正常结束时的返回值必须是 0。
3. 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）
4. 程序可使用的栈内存空间限制与题目的内存限制一致。
5. 评测时采用的机器配置为：Intel® Core™ i5-9500 CPU @ 3.00GHz，内存 8G，操作系统 Ubuntu 20.04，上述时限和各语言的编译器版本以此为准。
6. 难度顺序与题目顺序无关。
7. \sum 是求和运算符， $\sum_{i=1}^n a_i$ 的值等于 $a_1 + a_2 + \cdots + a_n$ 。
8. 豫阳市第三工程组提醒您：做题千万条，读题第一条；编程不规范，爆零两行泪。

马卡龙 (macaron)

【题目背景】

马卡龙，又称作玛卡龙、法式小圆饼，是一种用蛋白或 aquafaba、糖粉、蔗糖、扁桃仁粉以及食用色素制成的以蛋白脆饼为基础的法式甜点，通常在两块饼干之间夹有甘纳许、奶油乳酪或果酱等内馅。名称源于意大利语单词 macarone、maccarone 或 maccherone，一种意大利蛋白脆饼。——— 维基百科

【题目描述】

小 K 非常喜欢吃马卡龙。

某天，长者送给了他 n^2 个马卡龙，小 K 把这些马卡龙摆成了 $n \times n$ 的矩阵，第 i 行第 j 列的马卡龙的美味度是 $M_{i,j}$ 。

他一口气吃不下 n^2 个马卡龙，于是他想先吃 n 个试试口味。为了保持美观，每行每列都只能至多吃掉一个马卡龙。

吃掉 n 个马卡龙后，小 K 的快乐度是这些马卡龙的美味度的异或和。

小 K 想要知道他的快乐度可能是哪些值，由于他忙着吃东西，就把计算的任务丢给了你。

【输入格式】

第一行，一个正整数 n 。

第 i 行 ($2 \leq i \leq n+1$)， n 个自然数 $M_{i-1,j}$ 。

【输出格式】

一行，一堆自然数，表示他的快乐度可能是哪些值。

由于小 K 不想写 spj，你需要升序输出这些数。

【样例 1 输入】

```
1 3
  5 9 15
 19 7 2
4 1 0 0
```

【样例 1 输出】

```
2 7 9 10 26 28
```

【数据范围】

本题捆绑测试并子任务依赖。

对于所有数据, $n \leq 60, M_{i,j} < 2^{12}$ 。

- Subtask 1 (8 分): $n \leq 10$ 。
- Subtask 2 (15 分): $n \leq 14, M_{i,j} < 2^{10}$ 。
- Subtask 3 (15 分): $n \leq 18$, 依赖 Subtask 1,2。
- Subtask 4 (35 分): $n \leq 40$, 依赖 Subtask 3。
- Subtask 5 (7 分): $M_{i,j} < 2$ 。
- Subtask 6 (20 分): 无特殊限制, 依赖 Subtask 4,5。

长者 (genius)

【题目描述】

本题的编号方式为 0-based (从 0 开始)。

某一天，长者拿到了一张图，是 n 个点 m 条边的无向简单图 (V, E) ，第 i 个点有实数权值 c_i 。长者想要知道是否存在图中点的非空序列 p_0, p_1, \dots, p_{k-1} ，使得

1. 对于任意 $0 \leq i < j < k$ ，有 $p_i \neq p_j$ 。
2. 对于任意 $i \in [0, k)$ ，有 $(p_i, p_{(i+1) \bmod k}) \in E$ 。
3. $\sum_{i=0}^{k-1} c_{p_i} > \lfloor \frac{k}{2} \rfloor$ 。

如果存在这样的序列，TA 希望你按顺序输出任意一个合法的序列，否则你只需要告诉 TA 无解即可。

由于长者见多识广，你需要在一个测试点中处理 T 组数据的询问。

【输入格式】

从输入文件 `genius.in` 中读入数据。

第一行，一个正整数 T ，表示数据组数。之后对于每组数据：

- 第一行，两个自然数 n, m ，表示图的点数和边数。
- 第二行， n 个实数 c_i ，表示点的权值。
- 之后 m 行，每行两个自然数 u, v ，表示图的边。

【输出格式】

输出到文件 `genius.out` 中。

对于每组数据，一行，若无解则输出 0，否则输出序列长度 k 和 k 个自然数 p_i 。

【样例 1 输入】

```
3
2 4 4
0.5 0.5 0.5 0.5
0 1
5 1 2
2 3
3 0
8 6 7
0.5 0.5 0.334 0.5 0.333 0.334
```

```

    0 1
11 0 2
    1 3
    2 3
14 1 4
    2 5
    4 5
17 5 5
    1.0 0.0 0.667 0.5 0.0
    0 1
20 1 2
    2 3
    3 4
23 4 1

```

【样例 1 输出】

```

1 0
  5 4 5 2 0 1
  2 3 2

```

【样例 2】

见选手目录下的 `ex_genius2.in` 和 `ex_genius2.out`。

【数据范围】

本题的比较方式是自定义校验器 (special judge)，忽略行末空格和文末多余字符。

本题捆绑测试并子任务依赖。

对于所有数据， $T \leq 30, 1 \leq n \leq 500, 0 \leq m \leq 2000, 0 \leq c_i \leq 1, 10^3 \cdot c_i$ 是整数。

- Subtask 1 (10 分)：给定的图是树。
- Subtask 2 (90 分)：无特殊限制，依赖 Subtask 1。

独白 (monologue)

【题目描述】

本题的编号方式为 0-based (从 0 开始)。

小 K 的独白是一串无限正整数序列，开头 n 个数已被长者钦定为 a_i 。对于 $i \geq n$ 的情况，

$$a_i = \sum_{j=0}^{i-1} [j + a_j \geq i]$$

q 次询问，每次给出一个自然数 k ，求 a_k 。

【输入格式】

从输入文件 `monologue.in` 中读入数据。

第一行，两个正整数 n, q 。

第二行， n 个正整数 a_i 。

之后 q 行，每行一个自然数 k ，表示询问。

【输出格式】

输出到文件 `monologue.out` 中。

q 行，每行一个正整数，表示答案。

【样例 1-3 输入】

见选手目录下的 `ex_monologue?.in` 和 `ex_monologue?.out`。

【数据范围】

本题捆绑测试并子任务依赖。

对于所有数据， $n, q \leq 10^5, k \leq 10^{15}$ 。

对任意 $k \in [0, n)$ ，有 $|a_k - n| \leq 1$ 。

- Subtask 1 (10 分): $n, q, k \leq 10^3$ 。
- Subtask 2 (10 分): $k \leq 10^5$ ，依赖 Subtask 1。
- Subtask 3 (20 分): $k < n^2$ 。
- Subtask 4 (20 分): 对任意 $k \in [0, n)$ ，有 $a_k \neq n$ 。
- Subtask 5 (40 分): 无特殊限制，依赖 Subtask 2,3,4。

吃瓜 (gourds)

【题目描述】

又到了瓜果丰收的季节。Yazid、Zayid 和小 K 在分瓜吃。

他们有 n 个瓜，由于 $3|n$ ，于是他们想平分这些瓜。这些瓜被按顺序摆放成一列，他们对每个瓜的喜好程度可以用三个长为 n 的排列 a, b, c 表示出来，Yazid、Zayid 和小 K 对第 i 个瓜的喜好程度分别为 a_i, b_i 和 c_i 。

现在他们打算这样分配瓜：做 $\frac{n}{3}$ 次操作，每次操作他们分别选出之前还未选过的瓜中最喜欢的瓜，如果出现重复则他们会开始打架，分配就失败了。若自始至终都没有人打架，分配就成功了。

你知道了 a, b ，求有多少排列 c 使得分配成功。由于答案可能很大，你只需要输出答案对 $10^9 + 7$ 取模的值。

【输入格式】

从输入文件 `gourds.in` 中读入数据。

第一行，一个正整数 n 。

第二行， n 个正整数 a_i 。

第三行， n 个正整数 b_i 。

【输出格式】

输出到文件 `gourds.out` 中。

一行，一个整数，表示答案对 $10^9 + 7$ 取模的值。

【样例 1 输入】

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

【样例 1 输出】

```
2
```

【样例 1 解释】

可行的排列 c 有 $(3, 1, 2)$ 和 $(3, 2, 1)$ 。

【数据范围】

本题捆绑测试并子任务依赖。

对于所有数据， $n \leq 400, 3|n$ ， a, b 是 1 到 n 的排列。

子任务编号	分值	$n \leq$	子任务依赖
1	20	10	
2	20	20	1
3	10	30	2
4	10	50	3
5	10	70	4
6	10	100	5
7	20	400	6