2023 年广东省重点中学信息学邀请赛 (GDKOI 2023)

普及组 第一试

2023 年 3 月 11 日

注意事项

1. 严格按照题目所要求的格式进行输入、输出，否则严重影响得分。
2. 题目测试数据有严格的时间限制，超时不得分。
3. 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。
4. 输入文件格式不用判错；输入输出文件名均已给定，不用键盘输入。
5. 评测环境为 NOI 系列活动标准竞赛环境，编译器版本为 `g++ 9.4.0`。
6. 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)。
7. 对于 C++ 选手，64 位整数输入输出格式为 `%lld`。
8. 选手提交的程序源文件必须不大于 100KB。
9. 对于 C++ 语言的编译选项为 -O2 -std=c++14

试题名称	小学生数学题	Macaron	淋雨	置换
提交文件名	math.cpp	macaron.cpp	rain.cpp	permutation.cpp
输入文件名	math.in	macaron.in	rain.in	permutation.in
输出文件名	math.out	macaron.out	rain.out	permutation.out
时间限制	1 秒	2 秒	1 秒	1 秒
空间限制	512 MB	256 MB	512 MB	512 MB
满分	100	100	100	100

第一题 小学生数学题

提交文件:	math.cpp
输入文件:	math.in
输出文件:	math.out
时间空间限制:	1 秒, 512 MB

Moon 是一名小学生，在做作业时遇到了这样一个问题，对于给定正整数 n, k，求出下面表达式的值：

$$\sum_{i=1}^{n} \frac{i!}{i^k}$$

其中 $i!$ 表示 i 的阶乘运算，即 $i! = 1 * 2 * 3 * 4 * ... * i$。这个式子太难了，所以 Moon 希望得到你的帮助。但是因为 Moon 只学过整数运算，还没有学过实数运算，所以希望你可以帮助他求出这个式子在模 998244353 意义下的值。也就是说，如果最终的结果假如化简成为最简分数 $\frac{p}{q}$，只需要输出 $p * q^{-1} \bmod 998244353$ 即可，其中 q^{-1} 为 q 在模 998244353 下的逆元。

输入格式

第一行两个整数 n, k。

输出格式

一行，一个整数，代表模 998244353 意义下的答案。

样例数据

math.in	math.out
5 1	34
100 100	523011929
10000000 10000000	686183373

样例解释

样例 1 中，因为 $i!/i = (i-1)!$，所以原式等价于 $\sum_{i=1}^{5} (i-1)!$=34。

数据范围

对于所有的数据，有 $1 \le n, k \le 2 * 10^7$;

对于 30% 的数据，有 $k = 1$;

对于另外 30% 的数据，有 $1 \le k \le 3$。

第二题 Macaron

提交文件:	macaron.cpp
输入文件:	macaron.in
输出文件:	macaron.out
时间空间限制：	2 秒, 256 MB

给出 $n*m$ 的一块二维平面作为 Nana 的家，左上角墙角为 $(0,0)$ ，右下角墙角为 $(n+1,m+1)$。其中家里有 k 个家具，每个家具会占其中一个点，题目将会给出每个家具的坐标。

马卡龙是一只扫地机器人，半径为 r 的圆形的它可以向上下左右四个方向移动，移动前后必须保持圆心在整点上，并且不能穿过家具或外墙进行打扫，即身躯不可以与家具或墙壁有重合部分（允许相切）。它初始圆心位置为 (x_s,y_s) ，将会从此出发，打扫它能到达的区域。

马卡龙想知道自己可以打扫到多大面积。你只需要告诉马卡龙，它出发后它的圆心可以到达的平面内的整点数量。

对了，你只用将答案告诉马卡龙就够了，不需要告诉 Nana ，因为马卡龙不希望伤心的 Nana 会为这些琐事烦心。

输入格式

第一行有两个整数 n,m ，表示 Nana 的家的大小。

第二行有一个整数 r^2 ，表示马卡龙半径的平方。

第三行有两个整数 x_s,y_s ，表示马卡龙出发的位置，保证在其初始位置上，马卡龙不会与家具有重合部分。

第四行有一个整数 k，接下来 k 行里每行给出两个整数 x,y ，表示其中一个家具的坐标。

输出格式

仅一个整数 ans ，表示答案。

样例数据

macaron.in	macaron.out
10 10 5 4 5 5 7 10 6 10 5 9 4 9 4 10	29
见/example/macaron/下的 macaron1.in	见/example/macaron/下的 macaron1.out

数据范围

对所有数据满足 $0 \le k \le n * m,\ 1 \le r \le \min\left(\left\lfloor\frac{n}{2}\right\rfloor, \left\lfloor\frac{m}{2}\right\rfloor\right),\ 1 \le x_s \le n,\ 1 \le y_s \le m;$

其中有 30% 的数据点满足 $1 \le n, m \le 100;$

剩下 70% 的数据点满足 $1 \le n, m \le 1000$。

第三题 淋雨

提交文件:	rain.cpp
输入文件:	rain.in
输出文件:	rain.out
时间空间限制:	1 秒, 512 MB

Moon 发现自己来到了一个二维平面上，但是自己只能在 y=0 的直线上以不超过 v_c m/s 的速度行走（可以折返来回行走）。这个时候天空开始下了倾盆大雨，一共有 n 个雨滴，第 $i(1 \le i \le n)$ 个雨滴以 v_g m/s 的速度从 (x_i, y_i) 开始匀速下落，同时开始刮起了速度为 v_w m/s，方向为 x 轴正方向的大风，可以认为每个雨滴在水平方向上有了和风速一样的速度，以及风不会影响人的行走速度。

Moon 非常喜欢淋雨，为了简单起见把每个雨滴和 Moon 都视为是一个点，只有某个雨滴到达 x 轴的位置的同时，Moon 也正好在这个位置上，Moon 才可以被这个雨滴淋到。现在给出 q 个询问，第 $i(1 \le i \le q)$ 次询问给出一个初始位置 $(s_i, 0)$，Moon 想知道自己从 $(s_i, 0)$ 出发，在整个运动过程中，最多可以被多少个雨滴淋到呢?

输入格式

第一行五个整数 n, q, v_g, v_w, v_c；

接下来 n 行每行两个整数 x_i, y_i；

再接下来 q 行每行一个整数 s_i。

输出格式

对于每个询问输出一行一个整数，表示 Moon 最多可以被淋到的雨滴数量。

样例数据

rain.in	rain.out
4 4 1 1 5 -3 2 4 1 0 4 2 3 -4 1 -2 0	2 3 2 3
见/example/rain/下的 rain1.in	见/example/rain/下的 rain1.out

数据范围

对于所有的数据，有 $1 \le n, q \le 10^5, 1 \le v_w, v_g, v_c, y_i \le 10^6, -10^6 \le x_i, s_i \le 10^6$；

对于 30% 的数据，有 $1 \le n, q \le 100$；

对于另外 30% 的数据，有 $1 \le q \le 5$。

第四题 置换

提交文件:	permutation.cpp
输入文件:	permutation.in
输出文件:	permutation.out
时间空间限制:	1 秒, 512 MB

Moon 最近在玩一款名为 Shadowverse 的卡牌游戏，在非常有趣的游戏过程中，Moon 想到这样一个关于洗牌的问题。假设当前牌堆中有 n 张牌，第 i 张牌的标号为 i，我们定义一种洗牌方式是一个排列 X={$x_1, x_2, ..., x_n$}，也就是把牌堆中第 i 张位置的牌变成第 x_i 张。那么假设现在 Moon 按照 X 的洗牌方式洗了 k 次牌，不妨设最终得到了一个排列 Y={$y_1, y_2, ..., y_n$}，y_i 表示洗完牌后第 i 张牌的标号。Moon 希望你可以帮助他算出有多少合法的洗牌方式 X，满足洗了 K 次后变成排列 Y，由于答案可能很大，所以你只需要输出对 998244353 取模的答案即可。

形式化而言，考虑对于排列 P={$p_1, p_2, ..., p_n$} 和排列 Q={$q_1, q_2, ..., q_n$}, 定义这两个排列的乘积:

$$P \times Q = \{q_{p_1}, q_{p_2}, ..., q_{p_n}\}$$

而排列 X 的 k 次幂 X^k 为 k 个排列 X 的乘积，现在考虑给定排列 Y 和正整数 k, 求满足方程 $X^k = Y$ 的排列 X 的数量，对 998244353 取模。

输入格式

第一行是一个整数 T 表示测试数据组数。

每组数据包括两行，第一行两个正整数 n, k，分别表示排列 X 和 Y 的长度、洗了 k 次牌。

第二行是 n 个 1 到 n 内互不相同的正整数 $\{y_1, y_2, ..., y_n\}$，表示排列 Y。

输出格式

T 行，每行一个整数, 表示合法的洗牌方式的数量，对 998244353 取模。

样例数据

permutation.in	permutation.out
1 5 6 2 1 4 3 5	2
见/example/permutation/下的 permutation1.in	见/example/permutation/下的 permutation1.out

样例解释

样例中，X=[3,4,2,1,5] 或者 [4,3,1,2,5], 共两个合法排列。

数据范围

对于所有的数据，有 $1 \le n \le 3000, 1 \le k \le 10^6, 1 \le T \le 10$;

对于 20% 的数据，有 $1 \le n, k \le 8$；

对于另外 10% 的数据，仅保证 $1 \le n \le 8$；

对于另外 30% 的数据，仅保证 $1 \le n \le 50$。