

# 全国青少年信息学奥林匹克竞赛

## NOIP2023模拟

时间：7:50-12:20

题目名称	区间逆序对	排列	有向图最小环	二分图最大匹配
题目类型	传统型	传统型	传统型	传统型
目录	interval	permutation	cycle	matching
可执行文件名	interval	permutation	cycle	matching
输入文件名	interval.in	permutation.in	cycle.in	matching.in
输出文件名	interval.out	permutation.out	cycle.out	matching.out
每个测试点时限	2.0秒	1.0秒	1.0 秒	2.0秒
内存限制	512 MB	512 MB	512MB	512 MB
子任务数目	20	20	Subtask	20
测试点是否等分	是	是	否	是

### 提交源程序文件名

对于C++语言	interval.cpp	permutation.cpp	cycle.cpp	matching.cpp
---------	--------------	-----------------	-----------	--------------

### 编译选项

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

### 注意事项与提醒（请选手务必仔细阅读）

- 文件名（程序名和输入输出文件名）必须使用英文小写。
- C++ 中主函数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。
- 提交的程序代码文件的放置位置请参照各省的具体要求。
- 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。
- 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）。
- 程序可使用的栈内存空间限制与题目的内存限制一致。
- 全国统一评测时采用的机器配置为：Intel(R) Core(TM) i7-8700K CPU @ 3.70GHz，内存 32GB。上述时限以此配置为准。
- 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其为准。
- 终评测时所用的编译命令中不含编译选项之外的任何优化开关。

## 区间逆序对 (interval)

### 【问题描述】

有一个长度为  $n$  的序列  $a$ 。进行  $m$  次询问，每组询问中给定  $l, r$ ，求有多少组  $i, j$  满足  $l \leq i < j \leq r, a_i > a_j$ 。

### 【输入格式】

第一行，共两个整数，表示  $n, m$ 。

第二行，共  $n$  个整数，表示  $a_1, a_2, \dots, a_n$ 。

接下来  $m$  行，每行两个整数，表示一组询问中的  $l, r$ 。

### 【输出格式】

输出共  $m$  行，每行一个整数，表示答案。

### 【样例输入1】

```
4 6
1 4 2 3
1 2
1 3
1 4
2 3
2 4
3 4
```

### 【样例输出1】

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

### 【样例2】

见选手目录下的 *interval/interval2.in* 与 *interval/interval2.ans*。

### 【数据范围及约定】

对于 100% 的数据， $1 \leq n, m \leq 10^6, 1 \leq l \leq r \leq n, \forall i \in [1, n], 1 \leq a_i \leq 50$ 。

测试点 1 ~ 4 满足： $n, m \leq 10^3$ 。

测试点 5 ~ 12 满足： $n, m \leq 10^5$ 。

测试点 13 ~ 20 满足：无特殊限制。

本题输入输出量较大，请使用较快的输入输出方式。

## 2.排列 (permutation)

### 【题目描述】

给定一个  $1 \sim n$  的排列  $a$ 。

有一个初始为空的序列  $b$ 。你可以进行任意多次操作，每次操作为以下三种之一：

- 设  $a'$  满足  $\forall i \in [1, n], a'_i = a_{(i \bmod n) + 1}$ ，将  $a$  替换为  $a'$ 。相当于将  $a$  向左循环移位一格。
- 设  $a'$  满足  $\forall i \in [1, n], a'_i = a_{((i+n-2) \bmod n) + 1}$ ，将  $a$  替换为  $a'$ 。相当于将  $a$  向右循环移位一格。
- 选择一个  $i \in [1, m]$ ，并将  $a_i$  加入到  $b$  的末尾。

小  $L$  希望通过进行一些操作使得  $b$  的长度为  $n$  且从前往后依次为  $1 \sim n$ 。

你需要求出最少需要进行多少次前两种操作。

### 【输入格式】

第一行，共两个整数，表示  $n, m$ 。

第二行，共  $n$  个整数，表示排列  $a$ 。

### 【输出格式】

输出共一行，一个整数，表示答案。

### 【样例输入1】

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

### 【样例输出1】

```
6
```

### 【样例1解释】

一次操作一，此时  $a = (5, 1, 4, 2, 3)$ 。

一次操作三，选择  $i = 2$ ，此时  $b = (1)$ 。

两次操作二，此时  $a = (2, 3, 5, 1, 4)$ 。

两次操作三，依次选择  $i = 1, i = 2$ ，此时  $b = (1, 2, 3)$ 。

一次操作二，此时  $a = (4, 2, 3, 5, 1)$ 。

一次操作三，选择  $i = 1$ ，此时  $b = (1, 2, 3, 4)$ 。

两次操作二，此时  $a = (5, 1, 4, 2, 3)$ 。

一次操作三，选择  $i = 1$ ，此时  $b = (1, 2, 3, 4, 5)$ 。

共使用 6 次前两种操作。

## 【样例2】

见选手目录下的 *permutation/permutation2.in* 与 *permutation/permutation2.ans*。

## 【数据范围及约定】

对于 100% 的数据,  $1 \leq m \leq n \leq 5 \times 10^5$ 。

测试点 1 ~ 4 满足:  $n \leq 100$ 。

测试点 5 ~ 12 满足:  $n \leq 10^4$ 。

测试点 13 ~ 20 满足: 无特殊限制。

温馨提示: 某些 STL 真的非常非常慢, 使用前请谨慎考虑。

### 3.有向图最小环 (cycle)

#### 【题目描述】

有一个有向图  $G$ 。  $\forall u, v \in [1, n], u \neq v$ , 如果  $a_{u,v} \neq 0$ , 那么  $G$  中存在一条  $u \rightarrow v$  的边权为  $a_{u,v}$  的边, 否则  $G$  中不存在  $u \rightarrow v$  的边。

小  $L$  想从点 1 沿着  $G$  中的边走到点  $n$ , 然后再走回点 1。

你需要求出所有至少经过一次的边的边权和的最小值。保证有解。

#### 【输入格式】

第一行, 共一个整数, 表示  $n$ 。

接下来  $n$  行, 每行  $n$  个整数, 第  $i$  行的第  $j$  个数表示  $a_{i,j}$ 。

当  $i = j$  时  $a_{i,j}$  没有意义, 保证此时  $a_{i,j} = 0$ 。

#### 【输出格式】

输出共一行, 一个整数, 表示答案。

#### 【样例输入1】

```
4
0 1 0 0
0 0 1 0
1 0 0 1
0 1 0 0
```

#### 【样例输出1】

```
5
```

#### 【样例2】

见选手目录下的 *cycle/cycle2.in* 与 *cycle/cycle2.ans*。

#### 【数据范围及约定】

对于 100% 的数据,  $1 \leq n \leq 200, \forall i, j \in [1, n], 0 \leq a_{i,j} \leq 10^9$ 。

Subtask1(30%):  $n \leq 10$ 。

Subtask2(10%):  $\forall i, j \in [1, n], a_{i,j} = a_{j,i}$ 。

Subtask3(40%):  $n \leq 60$ 。

Subtask4(20%): 无特殊限制。

## 4.二分图最大匹配 (matching)

### 【题目描述】

有一个二分图  $G$ , 左右两边各有  $n$  个点, 有不超过  $m$  条边, 允许有重边。

左边的点标号为  $1 \sim n$ , 右边的点标号为  $n + 1 \sim 2n$ 。

设  $deg_i$  为点  $i$  的度数, 即  $G$  中与  $i$  相连的边数。

定义  $match(G)$  表示  $G$  的最大匹配。

你需要对于每个  $k \in [1, n]$  求出  $match(G) = k$  时  $\sum_{i=1}^{2n} a_{i, deg_i}$  的最大值。

### 【输入格式】

第一行, 共两个整数, 表示  $n, m$ 。

接下来  $2n$  行, 每行共  $m + 1$  个整数, 第  $i$  行的第  $j$  个整数表示  $a_{i, j-1}$ 。

### 【输出格式】

共一行,  $n$  个整数, 第  $i$  个数表示  $match(G) = i$  时的答案。

### 【样例输入1】

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

### 【样例输出1】

```
4 3
```

### 【样例2】

见选手目录下的 *matching/matching2.in* 与 *matching/matching2.ans*。

### 【数据范围及约定】

对于 100% 的数据,  $1 \leq n \leq 30, 1 \leq m \leq 60, n \leq m, \forall i \in [1, 2n], j \in [0, m], 0 \leq a_{i, j} \leq 10^7$ 。

测试点 1 ~ 4 满足:  $n \leq 4, m \leq 8$ 。

测试点 5 ~ 8 满足:  $n \leq 8, m \leq 16$ 。

测试点 9 ~ 16 满足:  $n \leq 20, m \leq 40$ 。

测试点 17 ~ 20 满足: 无特殊限制。

