CSP2024 Ex Simulation

by ForgotMe

2024 年 10 月 19 日　08:00 ~ 12:00

题目名称	冒泡排序	染色	图	山峦
题目类型	传统型	传统型	传统型	传统型
目录	bubble	color	graph	mountain
可执行文件名	bubble	color	graph	mountain
输入文件名	bubble.in	color.in	graph.in	mountain.in
输出文件名	bubble.out	color.out	graph.out	mountain.out
每个测试点时限	2.0 秒	3.0 秒	5.0 秒	2.0 秒
内存限制	512 MiB	1024 MiB	512 MiB	1024 MiB
子任务/测试点数目	10	25	10	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	bubble.cpp	color.cpp	graph.cpp	mountain.cpp
对于 C 语言	bubble.c	color.c	graph.c	mountain.c

编译选项

对于 C++ 语言	-lm -O2 -std=c++14 -Wl,-stack=1000000000
对于 C 语言	-lm -O2

注意事项

1. 选手请直接提交源程序至 becoder.com.cn 上的对应比赛。
2. 文件名（包括程序名和输入输出文件名）必须使用英文小写。
3. C++ 中函数 main() 的返回值类型必须是 int，值必须为 0。
4. 若无特殊说明，输入文件中同一行内的多个整数、浮点数、字符串等均使用一个空格分隔。
5. 若无特殊说明，结果比较方式为忽略行末空格、文末回车后的全文比较。
6. 只提供 Windows 格式附加样例文件。

冒泡排序（bubble）

【题目描述】

小 L 学习了冒泡排序，按理来说，正常的冒泡排序写法如下：

```
for(int i=1;i<n;i++)
  for(int j=1;j<=n-i;j++)
    if(a[j+1]<a[j])swap(a[j+1],a[j]);
```

无意间，小 L 发现了一个奇怪的冒泡排序写法，他称之为 K-BubbleSort：

```
for(int i=1;i<n;i++)
  for(int j=1;j<=n-i;j++)
    if(a[j+k]<a[j])swap(a[j+k],a[j]);
```

注意：在上述代码中可能出现 $j+k>n$ 的情况，此时请忽略这种情况，也就是认为 j 满足 $\leq n-k$。

可是 K-BubbleSort 无法一定使得一个序列有序，现在小 L 有一个序列 A 与一个常数 k，他想知道，如果对序列 A 做一次 K-BubbleSort，序列 A 将会变成什么模样?

【输入格式】

从文件 ***bubble.in*** 中读入数据。

输入第一行包含两个整数 n,k。

输入第二行包含 n 个整数，第 i 个整数表示 A_i。

【输出格式】

输出到文件 ***bubble.out*** 中。

输出共有一行包含 n 个整数，表示执行了 K-BubbleSort 之后的序列 A。

【样例 1 输入】

```
3 2
3 2 1
```

【样例 1 输出】

```
1 2 3
```

见选手目录下的 ***bubble/bubble1.in*** 与 ***bubble/bubble1.ans***。

【样例 2 输入】

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

【样例 2 输出】

```
2 1 4 3
```

见选手目录下的 ***bubble/bubble2.in*** 与 ***bubble/bubble2.ans***。

【样例 3】

见选手目录下的 ***bubble/bubble3.in*** 与 ***bubble/bubble3.ans***。
该样例约束与测试点 $4 \sim 6$ 一致。

【数据范围】

对于 100% 的数据，$1 \le n \le 10^6$，$1 \le k \le n-1$，$1 \le A_i \le 10^9$。
对于不同的测试点，作如下约定：

测试点编号	n	特殊性质
$1 \sim 3$	≤ 3000	无
$4 \sim 6$	$\le 10^5$	$k=2$
$7 \sim 10$	$\le 10^6$	无

染色（color）

【题目描述】

小 L 有一块**已被着色**的画布，上面有 n 个格子，第 i 个格子的初始颜色为 c_i，保证 $c_i \in [1, m]$，其中 m 是一个给定的常数。

小 L 有一支神奇的画笔，可以进行以下操作：

- 选择一个位置 $x(1 \le x \le n-1)$，将第 x 个格子颜色变成第 $x+1$ 个格子的颜色，即 $c_x \leftarrow c_{x+1}$。
- 选择一个位置 $x(1 \le x \le n-1)$，将第 $x+1$ 个格子颜色变成第 x 个格子的颜色，即 $c_{x+1} \leftarrow c_x$。

小 L 可以用这支画笔进行任意次操作（可以是 0 次），他想知道他能够得到多少种**本质不同**的画布？小 L 认为两块画布本质不同当且仅当存在同一个格子两块画布的颜色不同。由于答案可能很大，请输出答案对 2 取模后的值。

由于某些原因，本题**多组测试**，且请注意本题**特殊的空间限制**。

【输入格式】

从文件 ***color.in*** 中读入数据。

输入第一行包含一个整数 T，表示数据组数。

对于每一组数据：

第一行包含一个整数 n，表示画布上的格子个数。

接下来一行包含 n 个整数，第 i 个整数表示 c_i。

【输出格式】

输出到文件 ***color.out*** 中。

请注意本题特殊的输出方式。

输出包含一个长度为 T 的 01 字符串，第 i 个字符表示第 i 组数据的答案。

【样例 1 输入】

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

【样例 1 输出】

```
10
```

【样例解释 1】

对于第一组数据，总共会产生下列 7 种本质不同的画布：
$[1,1,1],[1,2,1],[1,2,2],[2,2,1],[2,2,2],[1,1,2],[2,1,1]$。
故答案模 2 后为 1。

【样例 2 输入】

```
2
5 4
1 3 4 2 3
6 4
1 3 3 3 2 4
```

【样例 2 输出】

```
10
```

【样例 3】

见选手目录下的 ***color/color3.in*** 与 ***color/color3.ans***。
该样例约束与测试点 $8\sim10$ 一致。

【样例 4】

见选手目录下的 ***color/color4.in*** 与 ***color/color4.ans***。
该样例约束与测试点 $17\sim19$ 一致。

【数据范围与提示】

对于 100% 的测试数据，$1\le T\le 2\times10^5$，$1\le m\le n\le 10^5$，$1\le\sum n\le 2\times10^5$，$1\le m\le 2\times10^4$，$1\le c_i\le m$。

对于不同的测试点，作如下约定：

测试点编号	n	m	$\sum n$	特殊性质
$1 \sim 3$	≤ 6	≤ 6	≤ 100	无
$4 \sim 7$	≤ 20	≤ 20	≤ 100	无
$8 \sim 10$	≤ 500	≤ 500	$\le 3 \times 10^3$	无
$11 \sim 13$	$\le 3 \times 10^3$	$\le 3 \times 10^3$	$\le 1.2 \times 10^4$	无
$14 \sim 16$	$\le 8 \times 10^3$	$\le 8 \times 10^3$	$\le 2 \times 10^4$	无
$17 \sim 19$	$\le 10^5$	≤ 15	$\le 2 \times 10^5$	无
$20 \sim 21$	$\le 2 \times 10^4$	$= n$	$\le 2 \times 10^5$	保证 $c_i = i$
$22 \sim 25$	$\le 10^5$	$\le 2 \times 10^4$	$\le 2 \times 10^5$	无

图（graph）

【题目描述】

可曾听闻过二分图最小生成树?

小 L 有一个大小为 $2 \times n$ 的二分图 G，其左部点的个数为 n，右部点的个数为 n。

小 L 得到了大小都为 n 的两棵无向树 T_1, T_2，这两棵树上的边都带权。

小 L 定义 $d_1(x,y)$ 表示 T_1 上 x 到 y 的简单路径边权之和，同样的 $d_2(x,y)$ 表示 T_2 上 x 到 y 的简单路径边权之和。

小 L 喜欢最小生成树，他定义二分图 G 上左部点 i 与右部点 j 之间的边的边权为：$f(i,j) = \max_{x=1}^{n} d_1(i,x) + d_2(j,x)$。

你能帮助小 L 求出二分图 G 的最小生成树的边权之和吗?

【输入格式】

从文件 ***graph.in*** 中读入数据。

输入第一行包含一个整数，表示 n。

接下来 $n-1$ 行每行包含三个整数 u, v, w，表示 T_1 上的一条边 (u, v, w)。

接下来 $n-1$ 行每行包含三个整数 u, v, w，表示 T_2 上的一条边 (u, v, w)。

【输出格式】

输出到文件 ***graph.out*** 中。

输出只有一行包含一个整数，表示答案。

【样例 1 输入】

```
2
1 2 1
1 2 1
```

【样例 1 输出】

```
4
```

见选手目录下的 ***graph/graph1.in*** 与 ***graph/graph1.ans***。

【样例解释 1】

$f(1,1) = 2$，$f(1,2) = 1$，$f(2,1) = 1$，$f(2,2) = 2$。

两条边权为 1 的边都要选，然后任意选择一条边权为 2 的边都可以组成该二分图 G 的最小生成树。

【样例 2 输入】

```
4
1 2 3
1 3 1
1 4 5
1 2 6
1 3 4
3 4 7
```

【样例 2 输出】

```
106
```

见选手目录下的 ***graph/graph2.in*** 与 ***graph/graph2.ans***。

【样例 3】

见选手目录下的 ***graph/graph3.in*** 与 ***graph/graph3.ans***。
该样例约束与子任务 2 一致。

【样例 4】

见选手目录下的 ***graph/graph4.in*** 与 ***graph/graph4.ans***。
该样例约束与子任务 4 一致。

【样例 5】

见选手目录下的 ***graph/graph5.in*** 与 ***graph/graph5.ans***。
该样例约束与子任务 5 一致。

【样例 6】

见选手目录下的 ***graph/graph6.in*** 与 ***graph/graph6.ans***。
该样例约束与子任务 6 一致。

【样例 7】

见选手目录下的 ***graph/graph7.in*** 与 ***graph/graph7.ans***。
该样例约束与子任务 7 一致。

【样例 8】

见选手目录下的 ***graph/graph8.in*** 与 ***graph/graph8.ans***。
该样例约束与子任务 8 一致。

【数据范围】

对于 100% 的数据，$1 \le n \le 10^5$，$1 \le u, v \le n$，$1 \le w \le 10^5$。
由于某些原因，**本题捆绑测试**。
对于不同的子任务，作如下约定：

子任务编号	分数	n	特殊性质
1	5	≤ 500	无
2	10	$\le 2 \times 10^3$	无
3	10	$\le 6 \times 10^3$	无
4	10	$\le 10^5$	A
5	10	$\le 10^5$	B
6	10	$\le 10^5$	C
7	10	$\le 10^5$	D
8	10	$\le 3 \times 10^4$	无
9	10	$\le 5 \times 10^4$	无
10	15	$\le 10^5$	无

特殊性质 A：保证 T_1, T_2 都是以 1 为根的菊花。
特殊性质 B：保证 $T_1 = T_2$，即两棵树完全相同（包括边权）。
特殊性质 C：保证 T_1, T_2 都是形如 $1 - 2 - ... - n$ 的链。
特殊性质 D：保证 T_1 是形如 $1 - 2 - ... - n$ 的链。

【提示】

请注意程序的常数因子优化。

山峦（mountain）

【题目背景】

翻过这座山，前方就是更灿烂的风景！

【题目描述】

小 L 喜欢群山，放眼望去，连绵不断的群山**每行**整齐排列。

小 L 仔细数了数，发现一共有 n 行山，且在第 i 行有 c_i 座山。

小 L 定义第 i 行第 j 座山的高度为 $h_{i,j}$。

小 L 继续仔细观察发现，$\forall 1 \le i \le n$，$h_{i,1} = c_i$，且 $c_1 \ge c_2 \ge \dots \ge c_n$。

小 L 认为一种群山组成的风景是**美丽的**当且仅当满足以下条件：

- $\sum_{i=1}^{n} \sum_{j=1}^{c_i} h_{i,j} = S$。
- $\forall 1 \le i \le n, 2 \le j \le c_i$，$1 \le h_{i,j} \le \min(h_{i-1,j}, h_{i,j-1})$，特别地，认为 $h_{0,x} = +\infty$。

小 L 想要知道，对于所有 $S \in [1, m]$，有多少种满足条件的群山组成的风景是美丽的？由于这个答案可能很大，你只需要输出答案对 998244353 取模后的值。小 L 认为两种群山不同，当且仅当存在一对 x, y 使得两种方案下对应的 $h_{x,y}$ 不同。

【输入格式】

从文件 ***mountain.in*** 中读入数据。

输入第一行包含两个整数，分别表示 n, m。

接下来一行包含 n 个整数，第 i 个整数表示 c_i。

【输出格式】

输出到文件 ***mountain.out*** 中。

输出只有一行包含 m 个整数，第 i 个整数表示 $S = i$ 时的答案。

【样例 1 输入】

```
2 10
2 2
```

【样例 1 输出】

```
0 0 0 0 0 1 1 1 0 0
```

见选手目录下的 ***mountain/mountain1.in*** 与 ***mountain/mountain1.ans***。

【样例解释 1】

一共有三种不同的群山：

- $$\begin{pmatrix} 2 & 1 \\ 2 & 1 \end{pmatrix}$$
- $$\begin{pmatrix} 2 & 2 \\ 2 & 1 \end{pmatrix}$$
- $$\begin{pmatrix} 2 & 2 \\ 2 & 2 \end{pmatrix}$$

【样例 2】

见选手目录下的 ***mountain/mountain2.in*** 与 ***mountain/mountain2.ans***。
该样例约束与测试点 $10 \sim 12$ 一致。

【样例 3】

见选手目录下的 ***mountain/mountain3.in*** 与 ***mountain/mountain3.ans***。
该样例约束与测试点 $16 \sim 17$ 一致。

【数据范围】

对于 100% 的测试数据，$1 \le n \le 10$，$1 \le m \le 300$，$1 \le c_n \le c_{n-1} \le ... \le c_1 \le 10$。
对于不同的测试点，作如下约定：

测试点编号	n	m	特殊性质
$1 \sim 3$	≤ 3	≤ 20	无
$4 \sim 6$	$= 2$	≤ 300	无
$7 \sim 9$	$= 3$	≤ 300	无
$10 \sim 12$	≤ 5	≤ 50	$c_1 \le 4$
$13 \sim 15$	≤ 8	≤ 200	$c_1 \le 6$
$16 \sim 17$	≤ 9	≤ 250	$c_1 \le 8$
$18 \sim 20$	≤ 10	≤ 300	无