

NOIP 2024 模拟赛

(请选手务必仔细阅读本页内容)

题目名称	花园	逆序对	步行	航行
题目类型	传统型	传统型	传统型	传统型
可执行文件名	garden	inverse	walk	sail
输入文件名	garden.in	inverse.in	walk.in	sail.in
输出文件名	garden.out	inverse.out	walk.out	sail.out
每个测试点时限	1.0 秒	1.5 秒	4.0 秒	1.0 秒
内存限制	512 MiB	512 MiB	512 MiB	512 MiB
测试点数量	10	10	25	25

提交源程序文件名

对于 C++ 语言	garden.cpp	inverse.cpp	walk.cpp	sail.cpp
-----------	------------	-------------	----------	----------

编译选项

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

注意事项与提醒 (请选手务必仔细阅读)

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照具体要求。
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. 程序可使用的栈内存空间限制与题目的内存限制一致。
7. 只提供 Linux 格式附加样例文件。
8. 评测在 NOI Linux 2.0 下进行, 各语言的编译器版本以其为准。
9. 下发了文件 `io.cpp` 供选手使用。

花园 (garden)

【题目描述】

现有一片 $n \times m$ 大小的花园，即 $(1,1) \sim (n,m)$ ，每个位置有一个权值 $c_{i,j}$ 。

你有一个装置，当你将它放置在位置 (x,y) 时，它将覆盖花园中所有形如 $(x+a_i, y+b_i)$ 的位置 ($1 \leq i \leq k$, a, b 都是长度为 k 的数组)。

现在你想选择花园中的两个位置放置该装置，要求最大化被覆盖的位置的权值和。

【输入格式】

从文件 *garden.in* 中读入数据。

第一行三个正整数 n, m, k ，意同题面。

接下来 n 行，每行一个长度为 m 的字符串，表示 $c_{i,j}$ 。

接下来 k 行，每行两个整数 a_i, b_i 。

【输出格式】

输出到文件 *garden.out* 中。

一行输出答案。

【样例输入 0】

```
1 3 3 9
2 111
3 111
4 111
5 00
6 01
7 02
8 10
9 11
10 12
11 20
12 21
13 22
```

【样例输出 0】

1 9

【样例输入 1 / 样例输出 1】

见下发文件 `ex_garden1.in` / `ex_garden1.out` 。

【数据范围】

对于前 50% 的数据： $1 \leq n, m \leq 30$ 。

对于前 70% 的数据： $1 \leq n, m \leq 50$ 。

对于所有数据： $1 \leq n, m \leq 100, 0 \leq k \leq 10, 0 \leq c_{i,j} \leq 9, |x_i|, |y_i| \leq 1000$ 。

逆序对 (inverse)

【题目描述】

给定一个长度为 n 的排列，可以进行下面的操作 **至多一次**：交换两个数的位置。
问最多能使排列的逆序对数减少多少。

【输入格式】

从文件 *inverse.in* 中读入数据。
第一行一个正整数 n ，意同题面。
第二行 n 个正整数 a_1, a_2, \dots, a_n 表示给定的排列。

【输出格式】

输出到文件 *inverse.out* 中。
一行输出答案。

【样例输入 1】

1 6
2 5 4 1 2 3 6

【样例输出 1】

1 3

【样例输入 2,3,4 / 样例输出 2,3,4】

见下发文件 *ex_inverse2,3,4.in* / *ex_inverse2,3,4.out* 。

【数据范围】

对于所有数据满足： $1 \leq n \leq 10^6$ 。
各测试点详细数据见表格：

测试点编号	$n \leq$	特殊性质
1	10	
2 ~ 3	100	
4 ~ 5	1000	
6 ~ 7	10^6	排列随机
8 ~ 10		

步行 (walk)

【题目描述】

你在一棵 n 个点的树上行走。你计划经过顶点 i 共 v_i 次。

我们称 walk 序列为一个长度为 $S = \sum_i v_i$ 的顶点序列，其中每个顶点 i 恰好出现 v_i 次。将 walk 序列顶点记为 w_1, w_2, \dots, w_S ，其“长度”被定义为 $\text{dist}(w_1, w_S) + \sum_{1 \leq i < S} \text{dist}(w_i, w_{i+1})$ ，其中 $\text{dist}(i, j)$ 表示 i, j 两顶点在树上的最短路径的长度。

现在你要处理 m 个事件，每个事件形如：断开 (a, b) 之间的边并连接 (c, d) 。若 (a, b) 之间不存在边，或连接 (c, d) 后不是一棵树（不连通），我们认为该事件的答案为 0。否则，该事件的答案记为所有 walk 序列中，长度的最大值。

每个事件是独立的。

由于输出量可能比较大，你只需要求出所有事件的答案的异或和。

【输入格式】

从文件 `walk.in` 中读入数据。

第一行两个整数 n, m ，意同题面。

第二行 n 个非负整数数， v_1, v_2, \dots, v_n ，意同题面。

接下来 $n - 1$ 行，每行两个正整数 u, v ，表示一条连接 u, v 的边。保证所有边构成一棵树。

接下来 m 行，每行四个正整数 a, b, c, d ，为每次事件的参数。

【输出格式】

输出到文件 `walk.out` 中。

一行输出所有事件的答案的异或和。

【样例输入 1】

```
1 6 4
2 1 1 4 5 1 4
3 1 2
4 2 3
5 1 4
6 4 5
7 4 6
8 1 2 1 3
9 1 4 2 6
```

10 1 4 2 4

11 1 6 3 6

【样例输出 1】

1 38

【样例解释 1】

前三次事件答案分别为 34, 36, 32，第四次事件非法，答案为 0。

【样例输入 2 / 样例输出 2】

见下发文件 `ex_walk2.in` / `ex_walk2.out`。

【样例输入 3 / 样例输出 3】

见下发文件 `ex_walk3.in` / `ex_walk3.out`。
该样例满足数据范围中的特殊性质 A。

【样例输入 4 / 样例输出 4】

见下发文件 `ex_walk4.in` / `ex_walk4.out`。
该样例满足数据范围中的特殊性质 B。

【样例输入 5 / 样例输出 5】

见下发文件 `ex_walk5.in` / `ex_walk5.out`。
该样例满足数据范围中的特殊性质 C。

【样例输入 6 / 样例输出 6】

见下发文件 `ex_walk6.in` / `ex_walk6.out`。

【数据范围】

对于所有数据满足： $1 \leq n \leq 3 \times 10^5, 1 \leq m \leq 1.5 \times 10^6, 1 \leq v_i \leq 10^8$ 。
设 $S = \sum_i v_i$ ，有 $1 \leq S \leq 10^{12}$ 。
 $1 \leq u, v \leq n$ 且所有边构成一棵树。
 $1 \leq a, b, c, d \leq n$ ，注意不保证 $a \neq b, c \neq d$ 。
各测试点详细数据见表格：

测试点编号	$n \leq$	$m \leq$	$S \leq$	特殊性质
1	5		10	
2	10			
3	20		100	
4	50			
5	100		10^{12}	
6	500			
7	1000			
8	3000		10^{12}	
9	5000			
10	5×10^4			A
11				
12	10^5			B
13				
14	3×10^5			A
15				B
16				C
17				
18	3×10^5	7×10^5		C
19				
20		1.5×10^6		A
21				B
22				C
23			10	
24		10^{12}		
25				

特殊性质 A: $\forall 2 \leq i \leq n$ 有边 $(i, \lfloor i/2 \rfloor)$ 。

特殊性质 B: 每个点度数不超过 2 。

特殊性质 C: $\{a, b\} \cap \{c, d\} \neq \emptyset$ 。

航行 (sail)

【题目描述】

你正在航行。从左到右有 n 条航道，标号分别为 $1, 2, \dots, n$ 。

航道 1 的左侧以及航道 n 的右侧皆为岸上。

对于每条航道 i 有参数 p_i ：每一时刻，该航道有 $p_i/100$ 的概率刮东风， $1-p_i/100$ 的概率刮西风。

在第 0 时刻，你的速度 $v = 0$ 。接下来的每一个时刻，若你处在位置 i ，根据航道 i 的风向，你的速度产生改变：若是东风，则 v 减少 1，否则 v 增加 1（即，我们以右为正方向）。接着，你的位置从 i 变为 $i + v$ ，并来到下一时刻。若某时有 $i \notin [1, n]$ 则表明你上岸了，停止航行。

若第 0 时刻，你处在位置 i ，你的期望上岸时间是多少？对每一个 $i \in [1, n]$ 输出答案模 998244353 意义下的结果，保证答案存在。若你始终无法上岸，输出 -1 。

【输入格式】

从文件 `sail.in` 中读入数据。

第一行一个正整数 n ，意同题面。

第二行 n 个整数 p_1, p_2, \dots, p_n ，意同题面。

【输出格式】

输出到文件 `sail.out` 中。

一行 n 个整数，输出每个位置的答案。

【样例输入 1】

```
1 2
2 50 50
```

【样例输出 1】

```
1 2 2
```

【样例输入 2 / 样例输出 2】

见下发文件 `ex_sail2.in` / `ex_sail2.out`。

【样例输入 3 / 样例输出 3】

见下发文件 `ex_sail3.in` / `ex_sail3.out` 。

该样例满足数据范围中的特殊性质 A 。

【样例输入 4 / 样例输出 4】

见下发文件 `ex_sail4.in` / `ex_sail4.out` 。

该样例满足数据范围中的特殊性质 B 。

【样例输入 5 / 样例输出 5】

见下发文件 `ex_sail5.in` / `ex_sail5.out` 。

【数据范围】

对于所有数据满足： $1 \leq n \leq 500, 0 \leq p_i \leq 100$ 。

各测试点详细数据见表格：

测试点编号	$n \leq$	特殊性质
1	1	A
2	2	
3	3	
4	4	
5	5	B
6	10	A
7		
8	15	
9		
10	20	
11		
12	30	A
13		
14	40	B
15		A
16		
17	80	A
18	100	
19	150	B
20	200	A
21	250	
22	300	B
23	350	A
24	400	
25	500	

特殊性质 A: p_i 在所有可行值中等概率均匀随机。

特殊性质 B: $p_i \in \{0, 100\}$ 。