

NOIP2024 模拟试题

(2024年9月24日8:00~12:30)

(请选手务必仔细阅读本页内容)

一. 题目概况

	避难	吃药	寻宝	盗梦
英文题目名称	disaster	drag	treasure	dream
输入文件名	disaster.in	drag.in	treasure.in	dream.in
输出文件名	disaster.out	drag.out	treasure.out	dream.out
每个测试点时限	2.0s	1.0s	4.0s	2.0s
测试点数目	子任务	子任务	子任务	子任务
每个测试点分值	/	/	/	/
结果比较方式	文本比较	文本比较	文本比较	文本比较
题目类型	传统	传统	传统	传统
运行内存上限	512 MB	512 MB	512 MB	512 MB
提交源程序文件名	disaster.cpp	drag.cpp	treasure.cpp	dream.cpp

二. 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，不需要建子文件夹。
2. 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）。
3. 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. 编译附加命令为 `-lm -O2 -std=c++14 -Wl,-stack=2147483647`
5. 选手提交的程序源文件必须不大于 100KB。
6. 禁止在源代码中改变编译器参数（如使用 `#pragma` 命令），禁止使用系统结构相关指令（如内联汇编）和其他可能造成不公平的方法。
7. 题面不一定包含全部样例，请查看下发附加文件以得到所有样例。
8. 评测在 NOI Linux 下进行。
9. 评测机配置为 Intel(R) Core(TM) i7-10510U CPU @ 1.80GHz 2.30 GHz, 16GB。已确认时限均在 std 两倍运行时间的两倍以上。
11. 温馨提示：你可以使用 `size` 命令（与 `g++` 在同一目录下）来测试程序所使用的静态内存。

1 避难 (disaster)

1.1 题目描述

A 国的道路构成了一棵有根树形结构，其中有 $n - 1$ 条道路连接 A 国的 n 个城市，其中 1 号城市是首都，即树的根节点。A 国经常受到洪水威胁，某些城市已经修建了避难所。在接下来的 q 天中，每一天都会发洪水，其中第 i 天会有一些城市 $v_{i,1}, v_{i,2}, \dots, v_{i,k_i}$ 受到洪水威胁，这些城市的居民需要尽快逃到它们所在的道路上的最近避难所，其所需的时间等于沿途经过的道路的权值之和。在每一天的洪水威胁下，你可以选择一个城市并将其设为新的避难所，但你的目标是最小化所有人抵达避难所的最晚时间。需要注意的是，每一天的情况都是独立的。

如果某个城市本身已经有避难所，那么居民从该城市到达避难所的时间将为 0。

1.2 输入格式

第一行三个正整数 n, m, q ，表示城市个数，避难所个数，询问次数。

接下来一行 m 个数 r_1, \dots, r_m 表示避难所所处城市的编号。

接下来 $n - 1$ 行每行 3 个整数 u, v, w ，表示一条连接 u, v ，边权为 w 的道路。

接下来 q 行，每行第一个数 k_i 表示受到威胁的城市个数，接下来 k_i 个数 $v_{i,1}, \dots, v_{i,k_i}$ 表示这些城市的编号。

1.3 输出格式

对于 q 个询问，每行一个整数表示答案。

1.4 样例输入输出

共 2 个样例，见下发文件。

1.5 数据范围

测试包编号	$n \leq$	$\sum k_i \leq$	测试包分值
1	10^3	10^3	20
2	10^5	10^5	50
3	10^6	2×10^6	30

对于全部数据 $1 \leq m \leq n \leq 10^6, 1 \leq \sum k_i \leq 2 \times 10^6, 1 \leq w \leq 10^9$ 。保证 $v_{i,1}, \dots, v_{i,k_i}$ 互不相同。

时空限制：2s, 512MB。

2 吃药 (drag)

2.1 题目描述

可怜的小 B 在感冒的时候拉了肚子，接下来的 n 天，他每隔至多 A 天就要吃一颗感冒药，每隔至多 B 天就要吃一次腹泻药，但是一天不能同时吃两种药。他想知道，这 n 天至少要吃多少次药。

每隔至多 x 天就要吃药，即这 n 天中不存在连续的 x 天使得小 B 没有吃某一种药。

2.2 输入格式

一行 3 个正整数 n, A, B 。

2.3 输出格式

一行 1 个整数表示答案。

2.4 样例输入输出

共 4 个样例，见下发文件。

2.5 数据范围

测试包编号	$n \leq$	$A, B \leq$	测试包分值
1	10^3	10^3	20
2	10^6	10	30
3	10^6	100	30
4	10^6	10^6	20

对于全部数据 $2 \leq A, B \leq n \leq 10^6$ 。

每个子任务依赖于严格弱于它的子任务。

时空限制：1s, 512MB。

3 寻宝 (treasure)

3.1 题目描述

A, B 君在寻宝，地图共有 n 个点和 m 条有向边，以及 k 个宝藏，每个宝藏有价值 w_i 。 A, B 一开始分别在点 a, b ， A, B 君轮流移动， A 君先移动，每次可以走一条边，如果走到的点有宝藏则可以收集这个宝藏。 A, B 君都想最大化 ta 拿到的宝藏价值和减去对方拿到的宝藏价值和。游戏结束当且仅当当前状态重复出现或所有宝石被取走。

假设 A, B 都足够聪明，问最后 A 拿到的宝藏价值和减去 B 拿到的宝藏价值和。

图中可能有自环，但不会有重边。保证每个点有至少一条出边。

3.2 输入格式

第一行 4 个正整数 n, m, a, b 。

接下来的 m 行，每行两个数 u, v 表示一条边。

接下来一个整数 k ，最后 k 行，每行两个数 x_i, w_i 表示第 k 个宝藏所在的点以及其价值。

3.3 输出格式

一个整数表示答案。

3.4 样例输入输出

共 3 个样例，见下发文件。

3.5 数据范围

测试包编号	$n \leq$	$k \leq$	特殊性质	测试包分值
1	10	1	无	15
2	10	5	无	25
3	30	10	图是 DAG	15
4	20	8	无	25
5	30	10	无	20

对于全部数据 $1 \leq n \leq 30, 0 \leq m \leq n^2, 1 \leq k \leq 10, 1 \leq w_i \leq 10^3$ 。保证 x_i 互不相同，保证起始点 a, b 没有宝藏。注意起始点 a, b 可能相同。

每个子任务依赖于严格弱于它的子任务。

时空限制：4s, 512MB。

4 盗梦 (dream)

4.1 题目描述

n 个平行世界中的你进入了一个名为盗梦空间的奇特领域。一开始, n 个你们都位于第 500,000 层的梦境中。随后, 每个时间步都会执行以下操作之一:

1. $1, l, r$: 位于平行世界 $[l, r]$ 中的你进入下一层梦境。这将导致世界 $[l, r]$ 中的梦境层数增加 1。
2. $2, l, r$: 位于平行世界 $[l, r]$ 中的你的从梦境的一层中醒来。这意味着世界 $[l, r]$ 中的梦境层数减少 1, 并且如果在减少前当前层梦境存在混乱, 那么混乱也会消失。
3. $3, l, r$: 你们中的某些自己在平行世界中 $[l, r]$ 范围的梦境层中制造了一次混乱。这意味着在第 i 个世界的 x_i 层会出现一个混乱, 其中 x_i 表示世界 i 的当前梦境层数。
4. $4, l, r$: 平行世界中 $[l, r]$ 范围的混乱变得无法控制, 因此你需要回到最浅层存在混乱的梦境层来解决混乱。然后, 你的梦境层数将变为 x_i , 其中 x_i 表示世界 i 最浅层混乱所在的层, 并且所有混乱都会消失。
5. $5, l, r$: 你查询平行世界 l 中的自己所在的梦境层数。请注意, 这里保证 $l = r$ 。
希望这个表达方式更清晰明了! 如果还有其他问题或需要进一步帮助, 请随时提问。

4.2 输入格式

第一行 2 个正整数 n, q 。

接下来 q 行, 每行 3 个数表示一次操作 op, l, r 。

4.3 输出格式

对每个询问, 输出一个整数表示答案。

4.4 样例输入输出

共 3 个样例, 见下发文件。

4.5 数据范围

测试包编号	$n, q \leq$	特殊性质	测试包分值
1	5×10^3	无	10
2	10^5	没有操作 2	20
3	10^5	操作 5 次数 ≤ 100	10
4	10^5	操作 4 次数 ≤ 100	10
5	10^5	无	30
6	5×10^5	无	20

对于全部数据 $1 \leq \sum n, q \leq 5 \times 10^5$ 。

时空限制：2s, 512MB。