2026 CCF 非专业级软件能力认证

CSP-J/S 2026 第二轮

提高组

时间：2026 年 10 月 31 日 14:30 ～ 18:30

题目名称	能量水晶	地铁	迁徙	研发基地
题目类型	传统型	传统型	传统型	传统型
目录	`crystal`	`subway`	`migrate`	`hub`
可执行文件名	`crystal`	`subway`	`migrate`	`hub`
输入文件名	`crystal.in`	`subway.in`	`migrate.in`	`hub.in`
输出文件名	`crystal.out`	`subway.out`	`migrate.out`	`hub.out`
每个测试点时限	1.0 秒	1.0 秒	1.0 秒	3.0 秒
内存限制	1024 MiB	1024 MiB	1024 MiB	1024 MiB
测试点数目	20	25	25	20
测试点是否等分	是	是	是	是

提交程序源文件名

对于 C++ 语言	`crystal.cpp`	`subway.cpp`	`migrate.cpp`	`hub.cpp`

编译选项

对于 C++ 语言	`-O2 -std=c++14 -static`

注意事项（请仔细阅读）

1. 文件名（程序名和输入输出文件名）必须使用英文小写。
2. `main` 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。
3. 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换行）。
4. 选手提交的程序源文件大小不得超过 100 KiB。
5. 程序可使用的栈空间内存限制与题目的内存限制一致。
6. 禁止在提交的程序源文件中改变编译器参数（如使用 `#pragma` 命令），禁止使用系统结构相关指令（如内联汇编）或其他可能造成不公平的方法。
7. 提交的程序源文件的放置位置请参考各省的具体要求。
8. 因违反上述规定而出现的问题，申诉时一律不予受理。
9. 只提供 Linux 格式附加样例文件。
10. 全国统一评测时采用的机器配置为：Intel Core Ultra 9 285K CPU @ 3.70 GHz（关闭睿频与能效核），内存 96 GB。上述时限以此配置为准。
11. 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此为准。

能量水晶（crystal）

【题目描述】

小 R 有 n 颗能量水晶，其中第 i $(1 \le i \le n)$ 颗的能量值为整数 a_i。

小 R 想用其中若干颗能量水晶合成一颗大水晶，用于合成的水晶必须具有相同的能量值。小 R 希望参与合成的水晶尽可能多。他可以进行**至多** m 次操作。每次操作可以任选一颗水晶，将其能量值增加 1；同一颗水晶可以被重复选择。

你需要帮助小 R 求出，进行至多 m 次操作后，他最多能用多少颗能量水晶合成一颗大水晶。

【输入格式】

从文件 ***crystal.in*** 中读入数据。

输入的第一行包含一个正整数 n 和一个非负整数 m，分别表示能量水晶的数量和最多可以进行的操作次数。

输入的第二行包含 n 个整数 $a_1, a_2, \ldots, a_n$，依次表示每颗能量水晶的能量值。

【输出格式】

输出到文件 ***crystal.out*** 中。

输出一行一个正整数，表示进行至多 m 次操作后，小 R 最多能用多少颗能量水晶合成一颗大水晶。

【样例 1 输入】

```
5 5
1 2 4 8 9
```

【样例 1 输出】

```
3
```

【样例 1 解释】

小 R 有 5 颗能量水晶，能量值分别为 $1, 2, 4, 8, 9$。他可以对能量值为 1 和 2 的水晶分别操作 3 次和 2 次，使原来能量值为 $1, 2, 4$ 的三颗水晶的能量值都变成 4，共进行 5 次操作。因此小 R 可以用 3 颗能量水晶合成一颗大水晶。

【样例 2 输入】

```
5 8
-3 -1 2 2 7
```

【样例 2 输出】

```
4
```

【样例 3】

见题目目录下的 ***crystal/3.in*** 与 ***crystal/3.ans***。
该样例满足测试点 $8, 9$ 的约束条件。

【样例 4】

见题目目录下的 ***crystal/4.in*** 与 ***crystal/4.ans***。
该样例满足测试点 $10 \sim 12$ 的约束条件。

【数据范围】

对于所有测试数据，均有：

- $1 \le n \le 5 \times 10^5$，$0 \le m \le 10^{18}$；
- 对于所有 i $(1 \le i \le n)$，均有 $|a_i| \le 10^{12}$。

测试点编号	$n \le$	$m \le$	$\max_{i=1}^{n}\|a_i\| \le$	特殊性质
$1, 2$	5	10	20	无
$3 \sim 5$	16	$1\,000$		
$6, 7$	$5\,000$	10^6	100	
$8, 9$		10^9	10^5	
$10 \sim 12$	5×10^5	10^8	100	
$13 \sim 15$		10^{18}	10^{12}	A
$16 \sim 18$				B
$19, 20$				无

特殊性质 A：对于所有 i $(1 \le i \le n)$，均有 $a_i = i$。
特殊性质 B：$a_1, a_2, \ldots, a_n$ 构成等差数列。

地铁（subway）

【题目描述】

B 市共有 m 个地铁站，编号为 1 到 m。每个地铁站都有一个站厅，地铁口与站厅之间有一条通道，往返耗时相同。从第 v $(1 \le v \le m)$ 个地铁站的地铁口走到站厅需要 w_v 的时间。

B 市共有 n 条地铁线路，编号为 1 到 n，其中第 i $(1 \le i \le n)$ 条线路经过 k_i 个地铁站，编号依次为 $a_{i,1}, a_{i,2}, \ldots, a_{i,k_i}$。在第 $a_{i,j}$ $(1 \le j \le k_i)$ 个地铁站处，从站厅走到第 i 条线路的站台需要 $t_{i,j}$ 的时间，往返耗时相同。同一条线路不会重复经过同一个地铁站，即 $a_{i,1}, a_{i,2}, \ldots, a_{i,k_i}$ 两两不同。

由于空间有限，每个地铁站最多有 d 条线路经过。此外，B 市的地铁网络是连通的，即任意两个地铁站都可以通过若干条线路互相到达。

小 R 有 q 个出行计划。对于第 r $(1 \le r \le q)$ 个出行计划，小 R 将从第 x_r 个地铁站的地铁口出发，乘坐地铁到达第 y_r 个地铁站的地铁口，途中可能进行若干次站内换乘。每次站内换乘都需要从一条线路的站台先走到站厅，再从站厅走到另一条线路的站台。每个出行计划的总步行时间为进出站和换乘的步行时间之和。小 R 希望每个出行计划的总步行时间都尽可能少。

你需要帮助小 R 求出，每个出行计划的最小总步行时间。

【输入格式】

从文件 ***subway.in*** 中读入数据。

输入的第一行包含五个非负整数 c, n, m, d, q，分别表示测试点编号、地铁线路数、地铁站数、每个地铁站最多被几条线路经过，以及出行计划数。特别地，若 $c = 0$，则该测试点为样例。

输入的第二行包含 m 个正整数 $w_1, w_2, \ldots, w_m$，依次表示从每个地铁站的地铁口走到站厅需要的时间。

输入的第 $i + 2$ $(1 \le i \le n)$ 行包含 $1 + 2k_i$ 个正整数，其中第一个正整数为 k_i，表示第 i 条线路经过的地铁站数；接下来 $2k_i$ 个正整数为 $a_{i,1}, t_{i,1}, a_{i,2}, t_{i,2}, \ldots, a_{i,k_i}, t_{i,k_i}$，其中 $a_{i,j}$ 表示第 i 条线路经过的第 j 个地铁站，$t_{i,j}$ 表示从该地铁站的站厅走到第 i 条线路的站台需要的时间。

输入的第 $r + n + 2$ $(1 \le r \le q)$ 行包含两个正整数 x_r, y_r，分别表示第 r 个出行计划的起点和终点所在的地铁站编号。

【输出格式】

输出到文件 ***subway.out*** 中。

输出 q 行，其中第 r $(1 \le r \le q)$ 行包含一个非负整数，表示第 r 个出行计划的最小总步行时间。

【样例 1 输入】

```
0 2 5 20 2
4 1 9 2 5
3 1 5 2 7 4 6
3 2 3 3 4 5 7
1 3
1 2
```

【样例 1 输出】

```
32
17
```

【样例 1 解释】

B 市有 5 个地铁站，从各地铁口走到站厅需要的时间分别为 $4, 1, 9, 2, 5$。

B 市有 2 条地铁线路，其中第 1 条线路依次经过第 $1, 2, 4$ 个地铁站，第 2 条线路依次经过第 $2, 3, 5$ 个地铁站，两条线路在第 2 个地铁站相交。

对于第 1 个出行计划，小 R 将从第 1 个地铁站前往第 3 个地铁站：他可以先从第 1 个地铁站的地铁口走到站厅，再走到第 1 条线路的站台，乘坐第 1 条线路到达第 2 个地铁站后走回站厅，再走到第 2 条线路的站台，乘坐第 2 条线路到达第 3 个地铁站后走回站厅，最后走到第 3 个地铁站的地铁口，总步行时间为

$$w_1 + t_{1,1} + (t_{1,2} + t_{2,2}) + t_{2,3} + w_3 = 4 + 5 + (7 + 3) + 4 + 9 = 32.$$

对于第 2 个出行计划，小 R 将从第 1 个地铁站前往第 2 个地铁站：他可以先从第 1 个地铁站的地铁口走到站厅，再走到第 1 条线路的站台，乘坐第 1 条线路到达第 2 个地铁站后走回站厅，最后走到第 2 个地铁站的地铁口，总步行时间为

$$w_1 + t_{1,1} + t_{1,2} + w_2 = 4 + 5 + 7 + 1 = 17.$$

【样例 2 输入】

```
0 3 7 20 2
8 3 6 1 7 2 9
3 1 4 2 5 5 100
3 2 6 3 1 7 2
4 3 2 5 100 4 6 6 7
1 6
2 3
```

【样例 2 输出】

```
35
16
```

【样例 2 解释】

B 市有 7 个地铁站与 3 条地铁线路，其中第 1 条线路依次经过第 $1, 2, 5$ 个地铁站，第 2 条线路依次经过第 $2, 3, 7$ 个地铁站，第 3 条线路依次经过第 $3, 5, 4, 6$ 个地铁站。

对于第 1 个出行计划，小 R 将从第 1 个地铁站前往第 6 个地铁站：他可以先从第 1 个地铁站的地铁口走到站厅，再走到第 1 条线路的站台，乘坐第 1 条线路到达第 2 个地铁站后走回站厅，再走到第 2 条线路的站台，乘坐第 2 条线路到达第 3 个地铁站后走回站厅，再走到第 3 条线路的站台，乘坐第 3 条线路到达第 6 个地铁站后走回站厅，最后走到第 6 个地铁站的地铁口，总步行时间为

$$w_1 + t_{1,1} + (t_{1,2} + t_{2,2}) + (t_{2,3} + t_{3,3}) + t_{3,6} + w_6 = 35.$$

对于第 2 个出行计划，小 R 将从第 2 个地铁站前往第 3 个地铁站：他可以先从第 2 个地铁站的地铁口走到站厅，再走到第 2 条线路的站台，乘坐第 2 条线路到达第 3 个地铁站后走回站厅，最后走到第 3 个地铁站的地铁口，总步行时间为

$$w_2 + t_{2,2} + t_{2,3} + w_3 = 16.$$

【样例 3】

见题目目录下的 ***subway/3.in*** 与 ***subway/3.ans***。
该样例满足测试点 $4 \sim 6$ 的约束条件。

【样例 4】

见题目目录下的 ***subway/4.in*** 与 ***subway/4.ans***。
该样例满足测试点 $9 \sim 11$ 的约束条件。

【样例 5】

见题目目录下的 ***subway/5.in*** 与 ***subway/5.ans***。
该样例满足测试点 12 ~ 15 的约束条件。

【样例 6】

见题目目录下的 ***subway/6.in*** 与 ***subway/6.ans***。
该样例满足测试点 16 ~ 19 的约束条件。

【样例 7】

见题目目录下的 ***subway/7.in*** 与 ***subway/7.ans***。
该样例满足测试点 22, 23 的约束条件。

【数据范围】

对于所有测试数据，均有：

- $0 \le c \le 25$，$1 \le n \le 500$，$1 \le m \le \sum_{i=1}^{n} k_i \le 5 \times 10^5$，$1 \le d \le 20$，$1 \le q \le 10^5$；
- 对于所有 v $(1 \le v \le m)$，均有 $1 \le w_v \le 10^9$；
- 对于所有 i $(1 \le i \le n)$，均有 $1 \le k_i \le m$；
- 对于所有 i, j $(1 \le i \le n,\ 1 \le j \le k_i)$，均有 $1 \le a_{i,j} \le m$，$1 \le t_{i,j} \le 10^9$；
- 对于所有 i $(1 \le i \le n)$，均有 $a_{i,1}, a_{i,2}, \ldots, a_{i,k_i}$ 两两不同；
- 对于所有 v $(1 \le v \le m)$，经过第 v 个地铁站的线路数均不超过 d；
- 任意两个地铁站都可以通过若干条线路互相到达；
- 对于所有 r $(1 \le r \le q)$，均有 $1 \le x_r, y_r \le m$，且 $x_r \ne y_r$。

<table>
<tr><th>测试点编号</th><th>$n \le$</th><th>$\sum_{i=1}^{n} k_i \le$</th><th>$d \le$</th><th>$q \le$</th><th>特殊性质</th></tr>
<tr><td>$1 \sim 3$</td><td>3</td><td>10</td><td rowspan="9">20</td><td>10</td><td rowspan="2">无</td></tr>
<tr><td>$4 \sim 6$</td><td>50</td><td>300</td><td rowspan="2">200</td></tr>
<tr><td>$7, 8$</td><td>200</td><td>400</td><td rowspan="2">A</td></tr>
<tr><td>$9 \sim 11$</td><td rowspan="3">500</td><td>1 000</td><td>10^5</td></tr>
<tr><td>$12 \sim 15$</td><td>2×10^4</td><td>2 000</td><td rowspan="2">B</td></tr>
<tr><td>$16 \sim 19$</td><td rowspan="4">5×10^5</td><td rowspan="4">10^5</td></tr>
<tr><td>$20, 21$</td><td>2</td><td rowspan="3">无</td></tr>
<tr><td>$22, 23$</td><td rowspan="2">500</td><td>2</td></tr>
<tr><td>$24, 25$</td><td>20</td></tr>
</table>

特殊性质 A：每条线路至多经过 2 个地铁站，即对于所有 i $(1 \le i \le n)$，均有 $k_i \le 2$。

特殊性质 B：对于所有线路经过的所有地铁站，从站厅走到对应线路站台的步行时间都相同，即存在正整数 T，使得对于所有 i, j $(1 \le i \le n,\ 1 \le j \le k_i)$，均有 $t_{i,j} = T$。

迁徙（migrate）

【题目描述】

海上有 n 座岛屿，自北向南排成一条直线，其中第 i $(1 \le i \le n)$ 座岛屿在这条直线上的坐标为整数 x_i，且 $x_1 < x_2 < \cdots < x_n$。每年秋季，一群海鸟会沿着这条直线在南北方向上迁徙，小 P 正在观察它们。

海鸟的体力有限，每次飞行的距离都不超过 d。若两座岛屿 i, j $(1 \le i, j \le n)$ 满足 $|x_i - x_j| \le d$，海鸟就可以从第 i 座岛屿直接飞到第 j 座岛屿。

对于两座不同的岛屿 u, v $(1 \le u, v \le n)$，一条从第 u 座岛屿到第 v 座岛屿的路径形如 $y_1 \to y_2 \to \cdots \to y_m$，其中 $y_1 = u$，$y_m = v$，且对于所有 l $(1 \le l \le m-1)$，海鸟都能够从第 y_l 座岛屿直接飞到第 y_{l+1} 座岛屿；其长度为 $m-1$，即海鸟在路径中飞行的次数。

小 P 发现，海鸟在迁徙时总是会选择长度最短的路径。对于两座岛屿 u, v $(1 \le u < v \le n)$，若从第 u 座岛屿到第 v 座岛屿的长度最短的路径**恰好**有两条，海鸟可以分成两群，分别沿其中一条飞往第 v 座岛屿；此时称 (u, v) 为**双径岛屿对**。两条路径不同，当且仅当它们经过的岛屿序列不完全相同。

你需要帮助小 P 求出双径岛屿对的数量。

【输入格式】

从文件 ***migrate.in*** 中读入数据。

本题包含多组测试数据。

输入的第一行包含两个非负整数 c, t，分别表示测试点编号与测试数据组数。特别地，若 $c = 0$，则该测试点为样例。

接下来依次输入 t 组测试数据。对于每组测试数据：

- 第一行包含两个正整数 n, d，分别表示岛屿数量和海鸟一次最多能飞行的距离。
- 第二行包含 n 个严格递增的整数 $x_1, x_2, \ldots, x_n$，依次表示每座岛屿的坐标。

【输出格式】

输出到文件 ***migrate.out*** 中。

对于每组测试数据，输出一行一个非负整数，表示双径岛屿对的数量。

【样例 1 输入】

```
0 2
4 2
1 2 3 4
```

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

【样例 1 输出】

```
1
2
```

【样例 1 解释】

该样例共包含两组测试数据。

对于第一组测试数据，仅有 1 个双径岛屿对 $(1,4)$。从第 1 座岛屿到第 4 座岛屿有两条长度最短的路径，分别为 $1 \to 2 \to 4$ 和 $1 \to 3 \to 4$。

对于第二组测试数据，共有 2 个双径岛屿对，分别为 $(1,4)$ 和 $(2,5)$。从第 1 座岛屿到第 5 座岛屿只有一条长度最短的路径 $1 \to 3 \to 5$，因此 $(1,5)$ 不是双径岛屿对。

【样例 2 输入】

```
0 2
5 3
-2 -1 0 1 2
6 2
-4 -3 -2 -1 3 6
```

【样例 2 输出】

```
0
1
```

【样例 2 解释】

该样例共包含两组测试数据。

对于第一组测试数据，共有 0 个双径岛屿对。从第 1 座岛屿到第 5 座岛屿有三条长度最短的路径，分别为 $1 \to 2 \to 5$，$1 \to 3 \to 5$ 和 $1 \to 4 \to 5$，因此 $(1,5)$ 不是双径岛屿对。

对于第二组测试数据，仅有 1 个双径岛屿对 $(1,4)$。从第 1 座岛屿到第 4 座岛屿有两条长度最短的路径，分别为 $1 \to 2 \to 4$ 和 $1 \to 3 \to 4$。第 5 座和第 6 座岛屿均无法到达任何其他岛屿。

【样例 3】

见题目目录下的 ***migrate/3.in*** 与 ***migrate/3.ans***。
该样例满足测试点 $3 \sim 6$ 的约束条件。

【样例 4】

见题目目录下的 ***migrate/4.in*** 与 ***migrate/4.ans***。
该样例满足测试点 $11, 12$ 的约束条件。

【样例 5】

见题目目录下的 ***migrate/5.in*** 与 ***migrate/5.ans***。
该样例满足测试点 $15 \sim 18$ 的约束条件。

【样例 6】

见题目目录下的 ***migrate/6.in*** 与 ***migrate/6.ans***。
该样例满足测试点 $19 \sim 21$ 的约束条件。

【数据范围】

设 N 为单个测试点内所有测试数据的 n 的和。对于所有测试数据，均有：

- $0 \le c \le 25$，$2 \le t \le 10^5$；
- $1 \le n \le 2 \times 10^6$，$N \le 2 \times 10^6$，$1 \le d \le 2 \times 10^{18}$；
- $-10^{18} \le x_1 < x_2 < \cdots < x_n \le 10^{18}$。

测试点编号	$N \le$	特殊性质
$1, 2$	9	无
$3 \sim 6$	300	
$7 \sim 10$	$5\,000$	
$11, 12$	2×10^6	A
$13, 14$		B
$15 \sim 18$	10^6	C
$19 \sim 21$	2×10^5	无
$22 \sim 25$	2×10^6	

特殊性质 A：$x_1, x_2, \ldots, x_n$ 构成等差数列。
特殊性质 B：存在一座岛屿，使得海鸟可以从该岛屿直接飞到任意其他岛屿。
特殊性质 C：$x_n - x_1 \le 10d$。

研发基地（hub）

【题目描述】

小 W 曾是一名科研爱好者。在 C 国工作时，他担任测试员，日复一日地测试各种新型传送门，在城市之间来回穿梭。久而久之，他厌倦了这样的生活，于是辞去工作，来到没有任何传送门的 D 国，希望在这里重新找回对研发的热爱。

在开始研发之前，小 W 需要建造研发基地。D 国共有 n 座城市，编号为 1 到 n。这 n 座城市由 $n-1$ 条道路连接，其中第 i $(1 \le i \le n-1)$ 条道路连接第 u_i 座和第 v_i 座城市。若将每座城市视作一个点，将每条道路视作连接两个点的边，则这 n 座城市和 $n-1$ 条道路构成一棵树 T。

小 W 的经费有限，他计划选择**恰好** k 座城市，在每座选中的城市里各建造一个研发基地。为了便于这些基地之间的协作，小 W 希望选中的城市在 T 中对应的点形成**不超过两个连通块**。形式化地，定义建造方案 $S \subseteq \{1, 2, \ldots, n\}$ 是**合理的**，当且仅当 $|S| = k$，且 S 在 T 中形成的连通块数量不超过 2。

对于合理的建造方案 S，定义第 i $(1 \le i \le n)$ 座城市的**通勤距离**为 $d(i) = \min_{j \in S} \mathrm{dist}(i, j)$，其中 $\mathrm{dist}(i, j)$ 表示从第 i 座城市到第 j 座城市所需经过的道路的最少条数。特别地，若 $i \in S$，则 $d(i) = 0$。定义该建造方案的**通勤半径**为所有城市通勤距离的最大值，即 $\max_{i=1}^{n} d(i)$。

你需要帮助小 W 求出，所有合理的建造方案的通勤半径的最小值。

【输入格式】

从文件 ***hub.in*** 中读入数据。

本题包含多组测试数据。

输入的第一行包含两个非负整数 c, t，分别表示测试点编号与测试数据组数。特别地，若 $c = 0$，则该测试点为样例。

接下来依次输入 t 组测试数据。对于每组测试数据：

- 第一行包含两个正整数 n, k，分别表示城市数量和小 W 计划建造的研发基地数量。
- 第 $i+1$ $(1 \le i \le n-1)$ 行包含两个正整数 u_i, v_i，分别表示第 i 条道路连接的两座城市的编号。

【输出格式】

输出到文件 ***hub.out*** 中。

对于每组测试数据，输出一行一个非负整数，表示所有合理的建造方案的通勤半径的最小值。

【样例 1 输入】

```
0 2
5 2
1 2
2 3
3 4
4 5
10 3
1 2
1 3
2 4
2 5
3 6
3 7
3 8
7 9
9 10
```

【样例 1 输出】

```
1
2
```

【样例 1 解释】

该样例共包含两组测试数据。

对于第一组测试数据，共有 10 种合理的建造方案。按通勤半径分类如下：

- 通勤半径为 1 的方案：$S = \{1, 4\}$，$S = \{2, 4\}$，$S = \{2, 5\}$；
- 通勤半径为 2 的方案：$S = \{1, 3\}$，$S = \{1, 5\}$，$S = \{2, 3\}$，$S = \{3, 4\}$，$S = \{3, 5\}$；
- 通勤半径为 3 的方案：$S = \{1, 2\}$，$S = \{4, 5\}$。

因此，所有合理的建造方案的通勤半径的最小值为 1。

对于第二组测试数据，一种合理的建造方案为 $S = \{2, 7, 9\}$，其中第 2 座城市单独构成一个连通块，第 7 座和第 9 座城市构成一个连通块。该建造方案的通勤半径为 2。

【样例 2】

见题目目录下的 ***hub/2.in*** 与 ***hub/2.ans***。

该样例满足测试点 2, 3 的约束条件。

【样例 3】

见题目目录下的 ***hub/3.in*** 与 ***hub/3.ans***。
该样例满足测试点 8 的约束条件。

【样例 4】

见题目目录下的 ***hub/4.in*** 与 ***hub/4.ans***。
该样例满足测试点 11, 12 的约束条件。

【样例 5】

见题目目录下的 ***hub/5.in*** 与 ***hub/5.ans***。
该样例满足测试点 13, 14 的约束条件。

【数据范围】

设 N 为单个测试点内所有测试数据的 n 的和。对于所有测试数据，均有：

- $0 \le c \le 20$，$1 \le t \le 5$；
- $2 \le n \le 10^6$，$N \le 2 \times 10^6$，$2 \le k \le n$；
- 对于所有 i $(1 \le i \le n-1)$，均有 $1 \le u_i, v_i \le n$，且所有 (u_i, v_i) 构成一棵树。

测试点编号	$n \le$	$N \le$	$k \le$	特殊性质
1	5	15	5	无
2, 3	10	30	10	
4 ∼ 6	100	300	100	
7	1 000	3 000	2	
8			1 000	
9, 10	4 000	10^4	4 000	
11, 12	10^5	3×10^5	2	
13, 14			10^5	A
15	5×10^5	10^6	10	无
16, 17			5×10^5	
18 ∼ 20	10^6	2×10^6	10^6	

特殊性质 A：存在正整数 m 满足 $n = 2m + 1$，且对于所有 i $(1 \le i \le m)$，均有 $u_i = i$ 且 $v_i = i + 1$；对于所有 i $(m + 1 \le i \le n - 1)$，均有 $u_i = i - m$ 且 $v_i = i + 1$。