

全国青少年信息学奥林匹克竞赛

NOIP2023模拟

时间：8:00-12:20

题目名称	查找替换	拖拉机路径	玛娜系列	子树激活
题目类型	传统型	传统型	传统型	传统型
目录	replace	paths	collection	activation
可执行文件名	replace	paths	collection	activation
输入文件名	replace.in	paths.in	collection.in	activation.in
输出文件名	replace.out	paths.out	collection.out	activation.out
每个测试点时限	2.0秒	4.0秒	5.0秒	2.0秒
内存限制	256 MB	512 MB	512MB	256MB
子任务数目	10	10	20	20
测试点是否等分	是	是	是	是

提交源程序文件名

对于C++语言	replace.cpp	paths.cpp	collection.cpp	activation.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）

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++ 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。
3. 提交的程序代码文件的放置位置请参照各省的具体要求。
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. 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。
8. 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。
9. 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。

查找替换 (replace)

【问题描述】

你有一个字符串 S ，最开始里面只有一个字符 a ，之后你要对这个字符串进行若干次操作，每次将其中每一个字符 c 替换成某个字符串 s （例如对于字符串 $ball$ ，将其中的 l 替换为 na 后将会变为 $banana$ ）。现在给定 l, r ，你需要输出 $S_{l \dots r}$ （也就是 S 的第 l 个字符到第 r 个字符对应的子串）是什么。

【输入格式】

第一行三个整数，分别表示 l, r 和操作次数。

接下来每一行一个字符 c 和一个字符串 s ，意义见题目描述。

【输出格式】

输出一行，表示对应的子串。

【样例输入1】

```
3 8 4
a ab
a bc
c de
b bbb
```

【样例输出1】

```
bdebbb
```

【样例1解释】

字符串转换如下： $a \rightarrow ab \rightarrow bcb \rightarrow bdeb \rightarrow bbbdebbb$

【数据范围及约定】

$l, r \leq \min(|S|, 10^{18})$, $r - l + 1 \leq 2 \times 10^5$, $\sum |s| \leq 2 \times 10^5$ 。

所有的字符串都只包含小写字母 $a - z$ 。

测试点 1 ~ 4，满足： $r - l + 1 \leq 2000$, $\sum |s| \leq 2000$ 。

测试点 5 ~ 10，无限制条件。

拖拉机路径 (paths)

【问题描述】

FJ 有 N 台拖拉机, 第 i 台拖拉机只能在闭区间 $[l_i, r_i]$ 中使用。拖拉机区间的端点满足 $l_1 < l_2 < \dots < l_N$ 且 $r_1 < r_2 < \dots < r_N$ 。一些拖拉机是特别的。

如果 $[l_i, r_i]$ 和 $[l_j, r_j]$ 相交, 则称两台拖拉机 i 和 j 为**相邻的**。FJ 可以从一台拖拉机转移到任意相邻的拖拉机上。两拖拉机 a 和 b 之间的**路径**是由一个转移序列组成的, 这个序列中第一台拖拉机是 a , 最后一台拖拉机是 b , 且序列中每两台连续的拖拉机都是相邻的。保证存在一条从拖拉机 1 到拖拉机 N 的路径。路径的长度是转移的次数 (或者等价地, 是序列中拖拉机个数减 1)。

给你 Q 个询问, 每个询问指定了一对拖拉机 a 和 b 。对于每个询问, 输出两个整数:

- 拖拉机 a 和 b 之间任意最短路径的长度。
- 有多少特别的拖拉机, 满足至少有一条从拖拉机 a 到拖拉机 b 的最短路径包含它。

【输入格式】

第一行两个整数 N ($2 \leq N \leq 2 \cdot 10^5$) 和 Q ($1 \leq Q \leq 2 \cdot 10^5$)。

接下来一行一个长度为 $2N$ 的字符串, 字符串中只包含 L 和 R 两种字符, 表示排好序后的左右端点。保证对于这个字符串的所有正规前缀 (不包括空串和全串), L 的个数多于 R 的个数。

接下来一行一个长为 N 的二进制串, 表示每台拖拉机是否是特别的。

接下来 Q 行每行两个整数 a 和 b ($1 \leq a < b \leq N$), 表示一组询问。

【输出格式】

对于每个询问, 输出一行两个整数, 整数之间用一个空格隔开。

【样例输入1】

```
8 10
LLLLRLLLLRRRRRRR
11011010
1 2
1 3
1 4
1 5
1 6
1 7
1 8
2 3
2 4
2 5
```

【样例输出1】

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

【样例1解释】

这 8 个拖拉机区间按顺序为 $[1, 5]$, $[2, 10]$, $[3, 11]$, $[4, 12]$, $[6, 13]$, $[7, 14]$, $[8, 15]$, $[9, 16]$ 。

对于第 4 个询问，第 1 台拖拉机和第 5 台拖拉机之间有 3 条最短路径：

$1 \rightarrow 2 \rightarrow 5$, $1 \rightarrow 3 \rightarrow 5$, $1 \rightarrow 4 \rightarrow 5$ 。这些最短路径长度都是 2。

此外，拖拉机 1, 2, 3, 4, 5 都是前面提到的三条最短路径中的一部分，由于拖拉机 1, 2, 4, 5 是特别的，所以有 4 台特别的拖拉机，满足至少有一条从拖拉机 1 到 5 的最短路径包含它。

【样例2】

见下发样例文件中。

【数据范围及约定】

测试点 1 ~ 2 组数据： $N, Q \leq 5000$ ；

测试点 3 ~ 5 组数据：最多有 10 个特别的拖拉机

测试点 6 ~ 10 组数据：无附加限制

玛娜系列 (collection)

【问题描述】

Bessie 最近对魔法产生了兴趣，并且为了一个非常重要的咒语收集法力。Bessie 有 N 个法力池，第 i 个池子每秒可以收集 m_i 单位的法力。这些池子被一组 M 条有向边 (a_i, b_i, t_i) 相连，表示她可以在 t_i 秒内从 a_i 到 b_i 。Bessie 无论何时出现在一个池子处，她都可以收集那个池子中所有的法力，也就是清空它。在时刻 0，所有的法力池都是空的，并且 Bessie 可以选择任意的法力池作为起点。

回答 Q 个询问，每个询问用两个整数 s 和 e 表示。对于每个询问，确定如果 Bessie 在第 s 秒结尾必须在池子 e 处的情况下，她所能收集到的最多法力值是多少。

【输入格式】

第一行两个整数 N ($1 \leq N \leq 18$) 和 M ($0 \leq M \leq N(N-1)$)。

接下来一行为 m_1, m_2, \dots, m_N ($1 \leq m_i \leq 10^8$)。

接下来 M 行，每行三个整数 a_i, b_i, t_i ($1 \leq a_i, b_i \leq N, a_i \neq b_i, 1 \leq t_i \leq 10^9$)。输入中没有有序数对 (a_i, b_i) 会出现超过一次。

接下来一行一个整数 Q ($1 \leq Q \leq 2 \cdot 10^5$)。

接下来 Q 行，每行两个整数 s ($1 \leq s \leq 10^9$) 和 e ($1 \leq e \leq N$)。

【输出格式】

输出 Q 行，表示对询问的回答。

【样例输入1】

```
2 1
1 10
1 2 10
4
5 1
5 2
100 1
100 2
```

【样例输出1】

```
5
50
100
1090
```

【样例1解释】

第一个询问：Bessie 在 5 秒后从池子 1 中获取 5 单位法力。

第二个询问：Bessie 在 5 秒后从池子 2 中获取 50 单位法力。

第三个询问：Bessie 在 100 秒后从池子 1 中获取 100 单位法力。

第一个询问：Bessie 在 90 秒后从池子 1 中获取 90 单位法力，然后在 100 秒后从池子 2 中获取 1000 单位法力。

【样例输入2】

```
4 8
50000000 100000000 20000000 70000000
1 2 20
2 1 50
2 3 90
1 3 40
3 1 10
4 1 25
1 4 5
4 3 70
3
8 3
1000000000 1
500000 4
```

【样例输出2】

```
1600000000
2399999988050000000
119992550000000
```

【样例2解释】

一个 Bessie 可以获取大量法力的样例。

【数据范围及约定】

测试点 1 ~ 2 组数据: $N \leq 10, Q \leq 100$
测试点 3 ~ 6 组数据: $N \leq 10$
测试点 7 ~ 10 组数据: $Q \leq 100$
测试点 11 ~ 13 组数据: $N = 16$
测试点 14 ~ 16 组数据: $N = 17$
测试点 17 ~ 20 组数据: 无附加限制

子树激活 (activation)

【问题描述】

为了庆祝新年, Bessie 和她的朋友们建造了一棵巨大的树, 上面有许多发光的彩灯。Bessie 可以通过遥控来打开和关闭这些灯。在太阳升起之前, 她想按一定的顺序开关一些灯 (可能一盏灯要开关一次以上), 使得开始和结束时树上都没有亮的灯。Bessie 认为, 如果亮的灯的集合正好是以某个顶点为根的子树, 那么这棵树看起来就很酷。她希望她开关的灯的顺序能满足这样一个属性: 对于每一棵子树, 在某个时间点, 它正好是所有亮的灯的集合。此外, 开关灯需要能量, Bessie 不想浪费能量, 所以她想找到她能执行的最小的开关次数。

形式化地, 有一棵 N 个节点的树, 节点编号为 $1 \dots N$, 根为节点 1。每个节点最初都是未被激活的。一次操作中, 你可以翻转一个节点的状态 (从未激活变为已激活, 反之亦然)。输出满足如下两个条件的最短操作序列的长度:

- 定义以节点 r 为根的子树由所有满足 r 位于 1 到 v 的路径上 (包括两端) 的节点 v 组成。对于这棵树的 N 棵子树中的每一棵, 都存在一个时刻, 满足已激活的顶点集合恰好是这棵子树的点集。
- 在结束所有操作后所有节点都是未被激活的。

【输入格式】

第一行一个整数 N ($2 \leq N \leq 2 \cdot 10^5$)。

第二行包含 p_2, \dots, p_N ($1 \leq p_i < i$), 其中 p_i 表示树上节点 i 的父节点。

【输出格式】

输出最小可能长度。

【样例输入1】

```
3
1 1
```

【样例输出1】

```
6
```

【样例1解释】

共有三棵子树, 分别为 $\{1, 2, 3\}$, $\{2\}$ 和 $\{3\}$ 。下面是一个可能的最短长度的操作序列。

- 激活节点 2 (以 2 为根的子树中所有节点都被激活了)
- 激活节点 1
- 激活节点 3 (以 1 为根的子树中所有节点都被激活了)
- 失活节点 1
- 失活节点 2 (以 3 为根的子树中所有节点都被激活了)
- 失活节点 3

【数据范围及约定】

测试点 1 ~ 2 组数据: $N \leq 8$

测试点 3 ~ 8 组数据: $N \leq 40$

测试点 9 ~ 14 组数据: $N \leq 5000$

测试点 15 ~ 20 组数据: 无附加限制