

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	字符串	颜色	训猫	现代机器
题目类型	传统型	传统型	传统型	传统型
目录	string	color	cat	machine
可执行文件名	string	color	cat	machine
输入文件名	string.in	color.in	cat.in	machine.in
输出文件名	string.out	color.out	cat.out	machine.out
每个测试点时限	1.0秒	1.0秒	1.0秒	2.0秒
内存限制	256 MB	256 MB	2048MB	1024MB
子任务数目	20	10	<i>Subtask</i>	<i>Subtask</i>
测试点是否等分	是	是	否	否

提交源程序文件名

对于C++语言	string.cpp	color.cpp	cat.cpp	machine.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。

1.字符串 (string)

【问题描述】

小 X 在学数数。

有一个长度为 n 的三进制串 s (每个字符都是 0 或 1 或 2), 有 m 个形如 (l, r, x) 的限制条件, 表示 s_l 到 s_r 中 (包括 s_l, s_r) 必须**恰好**有 x 个不同的字符 ($x \in \{1, 2, 3\}$)。

小 X 想知道共有多少个满足条件的 s 。

【输入格式】

第一行一个正整数 T , 表示数据组数。

对于每组数据, 第一行一个正整数 n , 一个整数 m , 分别表示三进制串 s 的长度和限制条件个数。

接下来 m 行, 每行三个正整数 $l\ r\ x$, 表示一个限制条件。

【输出格式】

输出共 T 行, 对于每一组数据, 输出一行一个整数 x , 表示满足条件的三进制串 s 的个数对 $10^9 + 7$ 取模后的结果。

【样例输入1】

```
3
1 0
2 0
3 0
5 2
1 3 3
4 5 1
```

【样例输出1】

```
3
9
27
18
```

【样例1解释】

样例第 4 组数据中满足条件的 s 有 01200 01211 01222 02100 02111 02122 10200 10211 10222 12000 12011 12022 20100 20111 20122 21000 21011 21022。

【样例2】

见下发文件 `string/string2.in` 和 `string/string2.out`, 该样例满足 $x = 1$ 的条件。

【样例3】

见下发文件 *string/string3.in* 和 *string/string3.out*。

【数据范围及约定】

对于 30% 的数据, $n \leq 10, m \leq 100$ 。

对于另外 10% 的数据, 保证 $x = 1$ 。

对于 100% 的数据, $T \leq 10, n \leq 200, \sum m \leq 10^6$ 。

2.颜色 (color)

【问题描述】

一棵 n 个节点的树，每个点可以被染成 c 种颜色之一。

一种合法染色方案被定义为：给每个点染上一种颜色，使得相邻点颜色不同。

现给定这棵树上一个大小为 m 的点集 S ，请你求出所有合法染色方案中，点集 S 中的不同颜色数之和。

【输入格式】

第一行，三个正整数 n, c, m ，分别表示树的点数，颜色种类数，点集 S 的大小。

接下来 $n - 1$ 行，每行两个正整数 x, y ，表示树上存在一条边 (x, y) 。

接下来一行， m 个数，表示点集 S 。

【输出格式】

输出共一行，一个正整数，表示所有染色方案中，点集 S 中的不同颜色数之和对 $10^9 + 7$ 取模后的结果。

【样例输入1】

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

【样例输出1】

```
18
```

【样例1解释】

我们令 (x, y, z) 表示给 1 号点染上颜色 x ，2 号点染上颜色 y ，3 号点染上颜色 z 的一种方案。

在所有合法方案中：

$(1, 2, 3), (1, 3, 2), (2, 1, 3), (2, 3, 1), (3, 1, 2), (3, 2, 1)$ 六种方案点集中的不同颜色数为 2。

$(1, 2, 2), (1, 3, 3), (2, 1, 1), (2, 3, 3), (3, 1, 1), (3, 2, 2)$ 六种方案点集中的不同颜色数为 1。

所以答案为 $6 \times 2 + 6 \times 1 = 18$ 。

【样例2】

见下发文件中的 *color/color2.in* 和 *color/color2.ans*。该样例满足 $n, c \leq 5$ 的限制。

【样例3】

见下发文件中的 *color/color3.in* 和 *color/color3.ans*。该样例满足树为一条链的限制。

【样例4】

见下发文件中的 *color/color4.in* 和 *color/color4.ans*。

【数据范围及约定】

对于 20% 的数据, $n, c \leq 5$;

对于另外 10% 的数据, $m = 1$;

对于另外 20% 的数据, $m = n$;

对于另外 20% 的数据, 树为一条链;

对于 100% 的数据, $n \leq 10^5, c \leq 10^9$ 。

3.训猫 (cat)

【问题描述】

有 N 个猫爬架，从 1 到 N 编号。猫爬架 i ($1 \leq i \leq N$) 的高度为 P_i 。猫爬架的高度是 1 到 N 之间（包括两端）互不相同的整数。有 $N - 1$ 对相邻的猫爬架。对于每个 j ($1 \leq j \leq N - 1$)，猫爬架 A_j 和 B_j 相邻。最初，可以从任意猫爬架开始，通过移动到与其相邻的猫爬架到达任意其他猫爬架。

在最初，一只猫处在高度为 N 的猫爬架上。

然后我们训练这只猫。在训练中，我们不断选择一个猫爬架，然后在它上面放置一个障碍物。然而，我们不能把障碍物放在已经放有障碍物的猫爬架上。在这个过程中，如下事件会发生。

- 如果猫没在这个选中的猫爬架上，那么什么都不会发生。
- 如果猫处于这个选中的猫爬架上，并且所有与其相邻的猫爬架上都有障碍物，那么训练结束。
- 否则，在猫可以通过不断移动到相邻且没有障碍物的猫爬架的方式到达的所有猫爬架中，猫将选择除目前所在猫爬架以外最高的那个，并移动到那里去。在这个过程中，猫将选择最短路径移动。

给定猫爬架的高度信息和相邻的猫爬架对，写一个程序计算如果我们恰当地放置障碍物，每次操作中猫移动次数之和的最大值是多少。

【输入格式】

第一行一个整数 N 。

第二行 N 个整数 P_1, P_2, \dots, P_N 。

接下来 $N - 1$ 行，每行两个整数 A_j, B_j 。

【输出格式】

输出一行一个整数，表示每次操作中猫移动次数之和的最大值。

【样例输入1】

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

【样例输出1】

```
3
```

【样例1解释】

如果我们按如下方式进行训练，那么猫总共会移动 3 次。

- 我们把障碍物放在猫爬架 1 上，猫不会动。
- 我们把障碍物放在猫爬架 2 上，猫从猫爬架 2 移动到 3。然后又从 3 移动到 4。
- 我们把障碍物放在猫爬架 4 上，猫从猫爬架 4 移动到 3。
- 我们把障碍物放在猫爬架 3 上，然后训练结束。

因为没有能让猫移动 4 次及以上的方式，因此输出 3。

这组样例满足子任务 1, 2, 3, 4, 5, 7 的限制。

【样例输入2】

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

【样例输出2】

```
7
```

【样例2解释】

这组样例满足子任务 4, 6, 7 的限制。

【数据范围及约定】

对于全部数据，满足 $2 \leq N \leq 2 \times 10^5$, $1 \leq P_i \leq N$ ($1 \leq i \leq N$),
 $P_i \neq P_j$ ($1 \leq i < j \leq N$), $1 \leq A_j < B_j \leq N$ ($1 \leq j \leq N - 1$)。

最初，可以从任意猫爬架开始，通过移动到与其相邻的猫爬架到达任意其他猫爬架。

详细子任务附加限制及分值如下表所示。

子任务编号	附加限制	分值
1	$A_i = i, B_i = i + 1$ ($1 \leq i \leq N - 1$), $N \leq 16$	10
2	$A_i = i, B_i = i + 1$ ($1 \leq i \leq N - 1$), $N \leq 300$	10
3	$A_i = i, B_i = i + 1$ ($1 \leq i \leq N - 1$), $N \leq 5\,000$	10
4	$N \leq 5\,000$	15
5	$A_i = i, B_i = i + 1$ ($1 \leq i \leq N - 1$)	15
6	$A_i = \lfloor \frac{i+1}{2} \rfloor, B_i = i + 1$ ($1 \leq i \leq N - 1$), 其中 $\lfloor x \rfloor$ 是小于等于 x 的最大整数	20
7	无附加限制	20

4.现代机器 (machine)

【问题描述】

Bitaro 生日这天收到了一个 JOI 机作为生日礼物。JOI 机由一个球, N 条光带和 M 个按钮组成。光带从 1 到 N 编号。当 Bitaro 打开开关时, 光带 i ($1 \leq i \leq N$) 会发出颜色 C_i 的光 (蓝光 (B) 或红光 (R))。按钮从 1 到 M 编号。如果 Bitaro 按下按钮 j ($1 \leq j \leq M$), 将发生如下事情。

1. 把球放置在光带 A_j 上。
2. 光带 A_j 变成红色 (不管它原来是什么颜色)
3. 进行如下操作, 直到球被移除。

令 p 为球目前所在的光带编号。

- 如果光带 p 是蓝色, 光带 p 变为红色。在此之后, 如果 $p = 1$, 这个球就被移除。否则, 球移向光带 $p - 1$ 。
- 如果光带 p 是红色, 光带 p 变为蓝色。在此之后, 如果 $p = N$, 这个球就被移除。否则, 球移向光带 $p + 1$ 。

Bitaro 对 JOI 机十分感兴趣。他计划进行 Q 次实验。在第 k ($1 \leq k \leq Q$) 次实验中, 在 Bitaro 开启电源后, 他将按 $L_k, L_k + 1, \dots, R_k$ 的顺序按下这些开关。在 Bitaro 按下一个开关后, 他将等到球被移除后再按下下一个开关。

给定 JOI 机和实验的情况, 写一个程序计算对于每个实验, 当实验结束后红色的光带有多少。

注: 每次实验之间互相独立。

【输入格式】

第一行两个整数 N, M 。

第二行一个长度为 N 的字符串 C , 字符串中仅包含字符 R 和 B。

第三行 M 个整数 A_j 。

第四行一个整数 Q 。

接下来 Q 行, 每行两个整数 L_k, R_k 。

【输出格式】

输出 Q 行, 第 k ($1 \leq k \leq Q$) 行输出一个整数, 表示当第 k 个实验结束后红色的光带有多少。

【样例输入1】

```
5 1
RBRRB
4
1
1 1
```


【样例输出1】

1

【样例1解释】

第一个实验按如下进行。

1. Bitaro 按下按钮 1，然后球被放在光带 4 上。
2. 光带 4 变红。因为光带 4 本身就是红色的，所以颜色不变。
3. 在此之后，进行了如下操作。
 - (1) 因为目前光带 4 是红色的，所以光带 4 变蓝，然后球移动到光带 5 上。
 - (2) 因为目前光带 5 是蓝色的，所以光带 5 变红，然后球移动到光带 4 上。
 - (3) 因为目前光带 4 是蓝色的，所以光带 4 变红，然后球移动到光带 3 上。
 - (4) 因为目前光带 3 是红色的，所以光带 3 变蓝，然后球移动到光带 4 上。
 - (5) 因为目前光带 4 是红色的，所以光带 4 变蓝，然后球移动到光带 5 上。
 - (6) 因为目前光带 5 是红色的，所以光带 5 变蓝，然后球被移除。

在实验后，光带 1 是唯一红色的光带。所以输出 1。

这组样例满足子任务 1, 2, 3, 6, 7 的限制。

【样例输入2】

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

【样例输出2】

```
5
0
```

【样例2解释】

对于第一个实验，在结束后光带 1, 2, 3, 4, 5 都是红色的，所以输出 5。

对于第二个实验，在结束后没有光带是红色的，所以输出 0。

这组样例满足子任务 3, 6, 7 的限制。

【样例输入3】

```
10 3
BBRRBRBRBR
2 10 5
1
1 3
```

【样例输出3】

```
2
```

【样例3解释】

这组样例满足子任务 1, 2, 3, 6, 7 的限制。

【样例输入4】

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

【样例输出4】

```
4
8
10
0
9
```

【样例4解释】

这组样例满足子任务 3, 4, 5, 6, 7 的限制。

【样例输入5】

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

【样例输出5】

```
2
6
0
10
7
```

【样例5解释】

这组样例满足子任务 3, 5, 6, 7 的限制。

【样例输入6】

```
30 10
RRRBBRBBBBRBBBBRBRBBBBRRRBBBBBRBR
3 28 2 29 1 30 6 14 7 7
10
1 10
2 3
2 5
2 8
3 3
3 6
4 5
4 7
5 9
10 10
```

【样例输出6】

```
21
15
15
4
17
16
14
20
12
23
```

【样例6解释】

这组样例满足子任务 6, 7 的限制。

【数据范围及约定】

对于全部数据，满足： $3 \leq N \leq 1.2 \times 10^5$ ， $1 \leq M \leq 1.2 \times 10^5$ ， C_i ($1 \leq i \leq N$) 不是 B 就是 R， $1 \leq A_j \leq N$ ($1 \leq j \leq M$)， $1 \leq Q \leq 1.2 \times 10^5$ ， $1 \leq L_k \leq R_k \leq M$ ($1 \leq k \leq Q$)。

详细子任务附加限制及分值如下表。

子任务 编号	附加限制	分 值
1	$N \leq 300, M \leq 300, Q \leq 1$	10
2	$N \leq 7\,000, M \leq 7\,000, Q = 1$	15
3	$Q \leq 5$	10
4	$N = 10$ 且 C_i ($1 \leq i \leq N$) 是 R	10
5	存在一个整数 t ($0 \leq t \leq N$)，使得对于每个 $i \leq t$ 都有 C_i 是 R，对于每个 $i > t$ 都有 C_i 是 B	20
6	$A_j \leq 20$ 或 $A_j > N - 20$ ($1 \leq j \leq M$)	15
7	无附加限制	20