

2022/10/05 的蒟蒻模拟实验题

题目名称	GYY的词典	1363的粉丝
题目类型	传统型	传统型
目录	char	1363
可执行文件名	char	1363
输入文件	char.in	1363.in
输出文件	char.out	1363.out
每个测试点时限	1s	1s
内存限制	512 MB	128 MB
测试点数目	10	20
测试点是否等分	是	是

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。
- \2. 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。
- \3. 若无特殊说明，结果比较方式为忽略行末空格、文末回车后的全文比较。
- \4. 若无特殊说明，输入文件与输出文件中同一行的相邻整数均使用一个空格分隔。
- \5. 题面经过删改，若有不通顺实属正常。一切请以简要题意为准。
- \6. 题目较为简单，AK 后请不要大声喧哗，闷声大發财。

GYY的词典(char)

Description

有 n 个字符串 T_1, \dots, T_n ，给出 m 个询问。

第 i 个询问给出一个字符串 S ，对于每个询问，我们可以得到一个长度为 n 的 $bool$ 数组 a ，其中 $a_i = [S$ 是否为 T_i 的前缀]

例如， $a = [0, 0, 1]$ 表示 S 是 T_3 的前缀，但不是 T_1, T_2 的前缀。对于每个询问给出的 a 数组，你的任务是求出它的最长全0子串长度

Input

第一行两个数 n, m ，表示有 n 个字符串， m 个询问。

接下来 n 行，每行一个字符串 T_i 。

再接下来 m 行，每行一个字符串 S 。

Sample Input

```
3 2
abcabc
aab
abbc
aa
ba
```

Sample Output

```
1
3
```

Hint

对于 20% 的数据， $n \leq 100, m \leq 100, \text{len}_{T_i} \leq 100, \sum \text{len}_S \leq 1000$

对于 40% 的数据， $n \leq 1000, m \leq 1000, \text{len}_{T_i} \leq 1000, \sum \text{len}_S \leq 10000$

对于 100% 的数据， $n \leq 100000, m \leq 100000, \sum \text{len}_{T_i} \leq 5000000, \sum \text{len}_S \leq 3000000$ ，

字符串中只包含 a, b, c 三种字母，数据随机。

1363的粉丝 (1363)

题目背景

1363 有一堆真正的粉丝。

(1363是一奇怪的人名)

题目描述

1363 太帅了。

因此,他有特别多忠实粉丝, 愿意献出自己寿命的 1s 等价交换为他的寿命 $+1\text{s}$ 。

1363与粉丝们在一个 $n * n$ 面积的城市中。1363可以吸收距离不超过 s 的粉丝所提供的寿命。

1363所在的十字路口 (a,b) 与粉丝所在的十字路口 (c,d) 之间距离是 $|a - c| + |b - d|$ 。

1363的洪荒之力会不断变化 q 次, 导致 s 的值也会变化多次。

现在要求你, 最聪明的粉丝, 计算出对于这 q 个 s , 每次最多增加多少秒的寿命。

输入格式

第一行四个整数 n, k, q 代表城市面积为 $n * n$, 粉丝数目 k , q 次功力变化;

- 接下来 k 行, 每行两个整数, 为粉丝坐标 xi, yi ;
- 接下来 q 行, 每行一个整数, 为当前最大距离 s 。

输出格式

对于每次询问, 输出一行。每一行只包含一个整数, 代表最多增加多少秒的寿命。

样例 #1

样例输入 #1

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

样例输出 #1

```
3
3
5
```

提示

坐标 $x_i, y_i > 0$ 。

对于30%的数据: $k \leq 100, n \leq 50, q \leq 5$;

对于70%的数据: $k \leq 10000$, $n \leq 500$, $q \leq 20$;

对于100%的数据: $k \leq 500000$, $n \leq 1000$, $q \leq 20$, xi , $yi \leq n$, $s \leq 10000$ 。