

## 【题目背景】

层叠样式表（Cascading Style Sheets，缩写 CSS）是一种用来为结构化文档（如 HTML 文档）添加样式（字体、间距和颜色等）的计算机语言。例如，对于以下的 HTML 文档：

```
<html>
  <head>
    <title>Sample</title>
  </head>
  <body>
    <h1>Hello</h1>
    <p id="subtitle">Greetings</div>
    <p>Hello, world!</p>
  </body>
</html>
```

配合以下 CSS 片段可以为其中的标题和段落设置相应的格式：

```
h1 { font-weight: bold; }
#subtitle { font-size: 12px; }
```

这段 CSS 片段为前面 HTML 文档添加了样式，使得标题“Hello”（第 6 行 `<h1>` 和 `</h1>` 标签之间的内容）具有粗体，使得段落“Greetings”（第 7 行 `<p id="subtitle">` 和 `</p>` 标签之间的内容）具有 12 个像素的字体大小。这里，CSS 片段第 1 行中出现的 `h1` 是一个选择器，它选中了 HTML 文档第 6 行的 `h1` 元素。CSS 片段第 2 行中出现的 `#subtitle` 也是一个选择器，选中了 HTML 文档第 7 行 `id` 属性为 `subtitle` 的 `p` 元素。注意它并没有选中 HTML 文档第 8 行不带属性的 `p` 元素。

## 【题目描述】

本题要实现一个简化版的元素选择器。给出一个结构化文档，和若干个选择器，对每个选择器找出文档中所对应选中的元素。

**结构化文档** 结构化文档由元素组成，一个元素可以包含若干个子元素（可以没有）。一个文档有一个根元素，在整体上形成树的结构。以下是本题结构化文档的一个例子：

```
html
..head
....title
..body
....h1
....p #subtitle
....div #main
.....h2
.....p #one
.....div
.....p #two
```

- 文档中每行表示一个元素，元素的标签由一个或者多个字母或数字组成。标签大小写不敏感，例如 `div`、`Div`、`DIV` 都是同一类标签。
- 元素可以附加一个 `id` 属性，属性值也是由一个或者多个字母或数字组成，之前有一个井号 `#`。`id` 属性大小写敏感，例如 `a` 和 `A` 是两个不同的 `id`。如果元素有 `id` 属性，标签和属性之间用一个空格字符分隔。
- 标签之前的缩进表示元素之间的包含关系：一个元素  $E$  所在行之后连续的缩进更深的行代表的元素是元素  $E$  的后代元素，其中缩进恰好深一层的是元素  $E$  的子元素。为了便于观察，每一级缩进用两个小数点符号 `..` 表示。

**选择器** 本题中会出现的选择器有三种，分别为：

- **标签选择器**：用标签来表示。例如 `p` 表示选择标签为 `p` 的所有元素。
- **id 选择器**：用 `id` 属性来表示。例如 `#main` 表示选择 `id` 属性为 `main` 的元素。题目保证文档中不同的元素不会有相同的 `id` 属性。
- **后代选择器**：复合表达式，格式为 `A B`，其中 `A` 和 `B` 均为标签选择器或 `id` 选择器，中间用一个空格字符分隔，表示选择满足选择器 `B` 的所有元素，且满足这些元素有祖先元素满足选择器 `A`。例如，选择器 `div p` 在上面的文档中会选中最后一行的元素 `p`，但不会选中 `id` 属性为 `subtitle` 的那个元素 `p`。注意，后代选择器可以有更多的组成部分构成，`div p` 是一个两级的后代选择器，而 `div div p` 则是一个三级的后代选择器。

### 【输入格式】

输入第一行是两个正整数  $n$  和  $m$ ，分别表示结构化文档的行数，和待查询的选择器的个数，中间用一个空格字符分隔。

第 2 行至第  $n + 1$  行逐行给出结构化文档的内容。

第  $n + 2$  行至第  $n + m + 1$  行每行给出一个待查询的选择器。记第  $n + 1 + i$  行的选择器为  $s_i, 1 \leq i \leq m$ 。

### 【输出格式】

输出共  $m$  行，每行有若干个整数。第  $i$  行表示选择器  $s_i$  选中的结果 ( $1 \leq i \leq m$ )。其中第一个整数  $r_i$  表示  $s_i$  选中的元素个数。随后  $r_i$  个整数，分别表示选中元素在结构化文档中出现的行号（行号从 1 开始编号）。行号按从小到大排序，相邻整数之间用一个空格字符分隔。

### 【样例输入】

```
11 5
html
..head
....title
..body
....h1
....p #subtitle
....div #main
.....h2
.....p #one
.....div
.....p #two
p
#subtitle
h3
div p
div div p
```

### 【样例输出】

```
3 6 9 11
1 6
0
2 9 11
1 11
```

### 【样例解释】

对于样例中查询的 5 个选择器：

1. `p` 选中所有的元素 `p`；
2. `#subtitle` 选中第 6 行 `id` 属性为 `subtitle` 的元素 `p`；
3. 由于没有标签为 `h3` 的元素，因此 `h3` 没有选中任何元素；
4. 第 9 行和第 11 行的 `p` 元素都有祖先是 `div` 元素，而第 6 行的 `p` 元素没有祖先是 `div` 元素；
5. `div div p` 要求选中的 `p` 元素有两级祖先都是 `div` 元素，只有第 11 行的 `p` 元素满足这个条件。

### 数据规模和约定

- $1 \leq n \leq 100$
- $1 \leq m \leq 10$
- 结构化文档和待查询的选择器每行长度不超过 80 个字符（不包括换行符）
- 保证输入的结构化文档和待查询的选择器都是合法的

测试点	结构化文档级数	id 属性	待查询选择器的类型
1	1	无	标签
2	2	无	标签
3	2	有	标签、id
4	2	无	标签、后代（两级，不含 id）
5	>2	无	标签
6	>2	有	标签、id
7	>2	无	标签、后代（两级，不含 id）
8	>2	有	标签、id、后代（两级）
9	>2	无	标签、后代（多级，不含 id）
10	>2	有	标签、id、后代（多级）

### 【提示】

多级的后代选择器在匹配时，可以采用贪心的策略：除最后一级外，前面的部分都可以尽量匹配层级小的元素。