

告警抑制 告警抑制题目描述：

告警抑制，是指高优先级告警抑制低优先级告警的规则。高优先级告警产生后，低优先级告警不再产生。请根据原始告警列表和告警抑制关系，给出实际产生的告警列表。

注：不会出现循环抑制的情况。

告警不会传递，比如 $A \rightarrow B, B \rightarrow C$ ，这种情况下 A 不会直接抑制 C 。

但被抑制的告警仍然可以抑制其他低优先级告警。

输入描述：

第一行为数字 N ，表示告警抑制关系个数， $0 \leq N \leq 120$

接下来 N 行，每行是由空格分隔的两个告警 ID ，例如： $id1 id2$ ，表示 $id1$ 抑制 $id2$ ，告警 ID 的格式为 大写字母+0 个或者 1 个数字

最后一行为告警产生列表，列表长度 $[1,100]$

输出描述：

真实产生的告警列表

补充说明：

告警 ID 之间以单个空格分隔

示例 1

输入：

2

A B

B C

A B C D E

输出:

A D E

说明:

A 抑制了 B, B 抑制了 C, 最后实际的告警为 A D E

示例 2

输入:

4

F G

C B

A G

A0 A

A B C D E

输出:

A C D E

说明:

```
import java.util.*;
```

```
// 注意类名必须为 Main, 不要有任何 package xxx 信息
```

```
public class Main {
```

```
    public static void main(String[] args) {
```

```
        Scanner in = new Scanner(System.in);
```

```
        // 注意 hasNext 和 hasNextLine 的区别
```

```
        // while (in.hasNextInt()) { // 注意 while 处理多个 case
```

```
        //     int a = in.nextInt();
```

```
        //     int b = in.nextInt();
```

```
        //     System.out.println(a + b);
```

```
        // }
```

```
        int count = Integer.parseInt(in.nextLine());
```

```
        Map<String, String> map = new HashMap<>();
```

```
        for (int i = 0; i < count; i++) {
```

```
            String curLine = in.nextLine();
```

```
            //System.out.println(curLine);
```

```

        // System.out.println("curLine[1] -> " + curLine.split(" ")[1]);
        // System.out.println("curLine[0] -> " + curLine.split(" ")[0]);
        map.put(curLine.split(" ")[1], curLine.split(" ")[0]);
    }

    // AB
    String[] arr = in.nextLine().split(" ");
    Set<String> set = new HashSet<>();
    for (String s : arr) {
        // System.out.println("this : " + s);
        set.add(s);
    }

    //System.out.println("size: "+ map.size());
    StringBuilder sb = new StringBuilder();
    for (String s : arr) {
        String cur = map.get(s);
        if (set.contains(cur)) {
            continue;
        }
        sb.append(s + " ");
    }
    System.out.println(sb.toString());
}
}

```