

计算最大乘积

题目描述：

给定一个元素类型为小写字符串的数组，请计算两个没有相同字符的元素 长度乘积的最大值，如果没有符合条件的两个元素，返回 0。

输入描述：

输入为一个半角逗号分隔的小写字符串的数组， $2 \leq \text{数组长度} \leq 100$ ， $0 < \text{字符串长度} \leq 50$ 。

输出描述：

两个没有相同字符的元素 长度乘积的最大值。

补充说明：

示例 1

输入：

```
iwdvpbn,hk,iuop,iikd,kadgpf
```

输出：

```
14
```

说明：

数组中有 5 个元素。

`iwdvpbn` 与 `hk` 无相同的字符，满足条件，`iwdvpbn` 的长度为 7，`hk` 的长度为 2，乘积为 14 (7×2)。

`iwdvpbn` 与 `iuop`、`iikd`、`kadgpf` 均有相同的字符，不满足条件。

`iuop` 与 `iikd`、`kadgpf` 均有相同的字符，不满足条件。

`iikd` 与 `kadgpf` 有相同的字符，不满足条件。

因此，输出为 14

```
import java.util.Scanner;
```

```
public class Main {
```

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

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

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

```
        String[] arr = input.split(",");
```

```
        int max = 0;
```

```
        for (int i = 0; i < arr.length; i++) {
```

```
            String local = arr[i];
```

```
            for (int j = i + 1; j < arr.length; j++) {
```

```
                String target = arr[j];
```

```
                char[] charArr = target.toCharArray();
```

```
                boolean flag = true;
```

```
                for (char c : charArr) {
```

```
                    if (local.contains(String.valueOf(c))) {
```

```
                        flag = false;
```

```
                        break;
```

```
        }
    }
    if (flag) {
        max = Math.max(max, local.length() * target.length());
    }
}
}
System.out.println(max);
}
```