

代码逻辑训练题

(第 6 次)

难度：简单，简单

限时：60 分钟

注：本次测试题为 2 题

- 文件编码必须为 UTF-8
 - Java 只允许使用 JDK 中的 API，不允许使用第三方 jar；
 - JavaScript 只允许使用原生，不要引入第三方 js 文件，且最终以 js 文件方式提交，不要将结果写在 html 中。
- 如违反上述条例均以签名错误判定。

题目一

问题描述

你正在郊区的街道上找一个地方停车。您只能在符合下列要求的位置停车：

- 1、不在私人车道。
- 2、不在公共汽车站。
- 3、不在公共汽车站前 10 米以内处。
- 4、不在小街车道。
- 5、不在小街车道前 5 米以内处。
- 6、不在小街车道后 5 米以内处。

街道将被表示为一个字符串，其中每个字符描述了一段 5 米长的街道。所以第一个字符描述了前 5 米的街道，第二个字符描述未来 5 米的街道，以此类推。街道将使用字母'D'表示私人车道，'B'表示巴士站，'S'表示小街车道和'-'表示其他类型的街道。车辆停在某位置上，车辆位置的索引即为街道字符串中相应字符的索引，车辆停于某位置之前，是指该车辆位置的索引低于街道中该位置字符的索引，车辆停于某位置之后，是指该车辆位置的索引高于街道中该位置字符的索引。请返回街道中满足上述条件的停车位的总数。

定义

Java

包名：	自己名字的缩写，如：package lhg;
类名：	StreetParking
方法：	freeParks

参数:	String
返回值:	int
方法签名:	public int freeParks(String street)

JavaScript

文件名:	StreetParking.js
函数名:	freeParks
参数:	字符串
返回值:	数字
方法签名:	function freeParks (street)

约束

1. 参数 `street` 只会包含 1 到 50 个字符（包含 1 和 50）
2. 参数 `street` 的字符只会包含 'D', 'B', 'S' and '-'。

示例

输入	返回
"---B--S-D--S--"	4 说明: ---B--S-D--S-- 可以停靠的位置如箭头所述, 返回 4
"DDBDDBDDBDD"	0
"--S--S--S--S--"	2

题目二

问题描述

一个数字集合是指一组彼此没有重复的数字, 我们可以定义以下集合操作:

两个集合 `a` 和 `b` 的“UNION”操作是指, 生成一个集合, 集合中包含 `a` 和 `b` 中的所有元素。

两个集合 `a` 和 `b` 的“INTERSECTION”操作是指, 生成一个集合, 集合中只包含既在 `a` 中, 也在 `b` 中的元素。

两个集合 `a` 和 `b` 的“SYMMETRIC DIFFERENCE”操作是指, 生成一个集合, 集合中只包含在 `a` 中或者 `b` 中的元素, 但是不包含既在 `a` 中, 也在 `b` 中的元素。

给定两个集合 `a` 和 `b`, 给定集合操作, 计算集合操作的结果, 以升序排列的 `int[]` 返回, 如果返回的集合为空, 则返回空的 `int[]`。

定义

Java

包名:	自己名字的缩写, 如: <code>package lhg;</code>
类名:	<code>Sets</code>
方法:	<code>operate</code>
参数:	<code>int[],int[] ,String</code>
返回值:	<code>int[]</code>
方法签名:	<code>public int[] operate(int[] a, int[] b, String operation)</code>

JavaScript

文件名:	<code>Sets.js</code>
函数名:	<code>operate</code>
参数:	数字数组,数字数组,字符串
返回值:	数字数组
方法签名:	<code>function operate (a, b, operation)</code>

约束

1. 参数 **a** 数组包含的元素数量只会在 0 到 50 之间（包含 0 和 50），其中每一个元素的数值只会在-1000000 到 1000000 之间（包含-1000000 和 1000000）。
2. 参数 **b** 数组包含的元素数量只会在 0 到 50 之间（包含 0 和 50），其中每一个元素的数值只会在-1000000 到 1000000 之间（包含-1000000 和 1000000）。
3. **a** 数组中不存在重复的元素。
4. **b** 数组中不存在重复的元素。
5. 参数 **operation** 只会是以下字符串之一: "UNION", "INTERSECTION", "SYMMETRIC DIFFERENCE"。

示例

输入	返回
<code>{1,2,3,4}</code> <code>{3,4,5,6}</code> <code>"INTERSECTION"</code>	<code>{ 3, 4 }</code> 说明: a 数组和 b 数组都包含的数字只有 3 和 4, 所以返回交集{3,4}
<code>{1,2,3,4}</code> <code>{3,4,5,6}</code> <code>"UNION"</code>	<code>{ 1, 2, 3, 4, 5, 6 }</code>

{432,756,123} {534,76,1209} "INTERSECTION"	{ }
{6,5,7,4} {7,6,4,10} "SYMMETRIC DIFFERENCE"	{ 5, 10 } 说明: 4, 6, 7 在两个集合中都有, 所以它们不被返回, 因此结果只包含 5 和 10.
{342,654,897,312,76,23,78} {21,43,87,98,23,756,897,234,645,876,123} "SYMMETRIC DIFFERENCE"	{ 21, 43, 76, 78, 87, 98, 123, 234, 312, 342, 645, 654, 756, 876 }