

- 一、JavaWeb
 - 1.1 学习目标
 - 1.2 学习途径
- 二、算法
 - 2.1 字符串规律
 - 2.2 找出数组中重复的数字
 - 2.3 实现二分查找
- 三、代码规范

一、JavaWeb

1.1 学习目标

- 掌握：
 - JavaWeb三大组件: Servlet, Filter, Listener
 - 请求处理对象: Request, Response
 - 请求(状态记忆)对象: Session, Cookie
- 了解: JSP、web域概念、MVC模式、三层架构

1.2 学习途径

- 视频: [黑马Web教程p84-p137](#)
- 博客:
 - [JavaWeb三大组件](#)
 - [JavaWeb开发入门](#)
 - [COOKIE和SESSION有什么区别?](#)
- 书籍: [Web应用开发:基于Spring MVC+MyBatis+Maven](#) 韩冬

二、算法

2.1 字符串规律

两个字符串: `pattern` 只包含小写字母, `str` 包含了由单个空格分隔的小写字母。判断 `str` 是否遵循相同的规律。这里的 **遵循** 指完全匹配, 例如, `pattern` 里的每个字母和字符串 `str` 中的每个非空单词之间存在着双向连接的对应规律。

示例1:

```
输入: pattern = "abba", str = "dog dog dog dog"
输出: false
```

解释: dog dog dog dog 属性 aaaa 型而非 abba

示例 2:

输入: pattern = "abba", str = "dog cat cat fish"
输出: false

解释: dog cat cat fish 不符合 abba 型

示例 3:

输入: pattern = "aaaa", str = "dog cat cat dog"
输出: false

示例 4:

输入: pattern = "abba", str = "dog cat cat dog"
输出: true

解释: dog cat cat dog 符合 abba 型

2.2 找出数组中重复的数字

在一个长度为 n 的数组 `nums` 里的所有数字都在 $0 \sim n-1$ 的范围内。数组中某些数字是重复的，但不知道有几个数字重复了，也不知道每个数字重复了几次。请找出数组中任意一个重复的数字。

示例：

输入：
[2, 3, 1, 0, 2, 5, 3]
输出: 2

2.3 实现二分查找

给定一个已经排好序的数组 `nums`，和一个目标数字 `target`，请使用二分查找的思路找到 `target` 在 `nums` 中的下标，如不存在，返回 `target` 应存放的位置的相反值-1。

思考: Java 内置的二分查找在找不到 `target` 的情况下为什么返回 `-index-1`

示例1：

输入：
`nums = [0,3,5,6,8,9]` `target=5`
输出: 2

示例2：

输入：
`nums = [0,3,5,6,8,9]` `target=7`
输出: -5

解释: `nums` 中没有7，应该放置下标为4，返回-5

三、代码规范

请复制以下内容到新建文件：Week7.java

```
import java.util.Random;

public class week7 {
    static Random random=new Random();
    public static void test(){
        System.out.println(task1("abbc","dog cat cat fish"));
        System.out.println(task2(new int[]{1,2,2,3,0}));
        for(int i=0;i<5;i++){
            int target = random.nextInt(15) - 3;
            System.out.println("target:"+target+"\tresult"+task3(new int[]
{0,4,5,6,8},target));
        }
    }
    public static void main(String[] args) {

    }

    public static boolean task1(String pattern, String str) {
        //第一题代码
        boolean res=true;

        return res;
    }

    public static int task2(int[] nums) {
        //第二题代码
        int res=0;

        return 0;
    }
    public static int task3(int []nums,int target){
        //第三题代码
        int res=0;

        return res;
    }
}
```