

后端题目

1. 给定一个递增排序的数组，查找某个数字是否在数组中，如果在数组中，则返回该数字在数组中第一次出现的位置（从 0 开始）；如果不在数组中，返回 -1。务必使用二分查找的方式。
2. 已知二维平面内的点数组 `Point[] points`，找出距离点 `p` 最近的 `n` 个点。`Point` 类定义如下：

```
Point{
    double x;
    double y;
}
```

完成以下方法：

```
Point[] getNearestPoints(Point[] points, Point p, int n){

}
```

（扩展：如果有排序，能否自己实现排序算法？如果点很多，能否不用计算所有点到目标点的距离以提高效率？）
3. 附件中的“省市县三级联动.json”文件，保存了全国的省市区(县)信息，将省、市、区(县)分别定位为 1、2、3 级别，编写一个方法，传入级别 `level`，返回该级别及其上级的所有信息。比如
传入 1，返回{“北京市”，“天津市”，“河北省”，...}，
传入 2，返回{“北京市-市辖区”，“天津市-市辖区”，“河北省-石家庄市”，“河北省-唐山市”，...}，
传入 3，返回{“北京市-市辖区-东城区”，“北京市-市辖区-西城区”，...}，
（扩展：能否实现一个通用的方法，对所有的类似这种结构的数据都有效，比如有 5 个级别的省、市、县、镇/乡、村的数据？）
4. 编写方法计算两条线段的交点。`p1`、`p2` 是线段 1 的两个顶点，`p3`、`p4` 是线段 2 的两个顶点。返回线段 1 和线段 2 的交点坐标，如果没有交点返回 `null`，假设两条线段不会重叠。`Point` 类包含两个成员变量 `x` 和 `y`，用于表示平面内的坐标。

```
public Point intersection(Point p1, Point p2, Point p3, Point p4){

}
```

5. 编写两个线程，交替打印斐波拉契数列。

示例输出：

线程 1: 0,

线程 2: 1,

线程 1: 1,

线程 2: 2,

线程 1: 3,

...

6. 站点 pm10 数据表 “site_pm10_info” 中记录了各个站点的 pm10 数据，请查出某段监测时间内监测站点超标(pm10 > 80)次数大于 2 次的站点编号。

| | | |
|-------------|----------|--------|
| id | int | 说明 |
| site_name | varchar | 监测站点编号 |
| site_number | varchar | 站点编号 |
| data_time | datetime | 监测时间 |
| pm10 | decimal | Pm10 值 |

site_pm10_info 表