



链滴

# 日常算法——二分法查找

作者: [JackHoo](#)

原文链接: <https://ld246.com/article/1510228694693>

来源网站: [链滴](#)

许可协议: [署名-相同方式共享 4.0 国际 \(CC BY-SA 4.0\)](#)

## 二分法查找

二分法也叫折半查找，当数据量很大适宜采用该方法。采用二分法查找时，数据必须是有序且不重复。基本思想：假设数据是按升序排序的，对于给定值  $x$ ，从序列的中间位置开始比较，如果当前位置等于  $x$ ，则查找成功；若  $x$  小于当前位置值，则在数列的前半段中查找；若  $x$  大于当前位置值则在数的后半段中继续查找，直到找到为止。

## 代码实现

```
package Suanfa;

/**
 * Created by hushangjie on 2017/11/9.
 */
public class BinarySearch {
    private int[] arr;

    public BinarySearch(int[] arr) {
        this.arr = arr;
    }

    public int searchRecursion(int target) {
        if (arr != null) {
            searchRecursion(target, 0, arr.length);
        } else {
            return -1;
        }
    }

    public int searchRecursion(int target, int start, int end) {
        if (start > end) {
            return -1;
        }
        int mid = (start + end) / 2;
        if (arr[mid] == target) {
            return mid;
        } else if (target > arr[mid]) {
            searchRecursion(target, mid + 1, end);
        } else {
            searchRecursion(target, start, mid - 1);
        }
    }
}
```

## 特点

二分法查找要求数列本身有序，如果无序还需排序，当我们从数据库针对某个字段进行排序的时候，要找出某个值得元素时，就比较适合二分查找了。