

快速排序

笔记本： 算法

创建时间： 2021/6/24 18:02

更新时间： 2021/6/25 17:12

标签： 排序算法

快速排序

算法步骤：

1. 确定分界点 x ($arr[(l + r) / 2]$ 或者 随机数)
2. 调整区间
3. 递归处理左右两段

调整区间

方法一： 暴力做法

设置两个数组 $a[]$, $b[]$; 扫描一遍 $arr[l]$ 到 $arr[r]$; 每当遇到小于等于 x 的数, 把它放入数组 a 中, 每当遇到大于 x 的数, 把它放入 b 中;

最后再将 $a[]$ 和 $b[]$ 中的数放入 $arr[]$ 分界点 x 的两边。

会有额外的空间复杂度, 但是时间复杂度为 $O(n)$;

方法二：

双指针法, 最开始设置一个指针 i , 数组末尾再设置一个指针 j 。每次选取数组中间的一个数作为标杆。

```
import java.io.BufferedReader;
import java.io.IOException;
import java.io.InputStreamReader;

public class TestJavaGrammer
{
    public static void quick_sort(int[] arr, int l, int r){
```

```

// 判断边界，如果边界中只有一个数或者没有数，就返回
if( l >= r) return;
int x = arr[(l + r)/2], i = l-1 , j = r+1;
while(i<j){
    do i++; while( arr[i] < x);
    do j--; while( arr[j] > x);
    if(i < j) {
        int t = arr[i];
        arr[i] = arr[j];
        arr[j] = t;
    }
}
quick_sort(arr,l,j);
quick_sort(arr,j+1,r);
}

public static void main(String[] args) throws IOException {
    BufferedReader in = new BufferedReader(new
InputStreamReader(System.in));
    int n = Integer.parseInt(in.readLine());
    int[] arr = new int[n];
    String[] strarr = in.readLine().split(" ");
    for(int i = 0 ;i < n ; i++){
        arr[i] = Integer.parseInt(strarr[i]);
    }
    quick_sort(arr,0,n-1);
    for(int i = 0; i< n; i++)
        System.out.print(arr[i]+ " ");
    in.close();
}
}

```