

# 快速排序

## 快速排序

```
#include<stdlib.h>

int partition(vector<int>& arr, int start, int end){
    int index = start + (int)end * rand()/(RAND_MAX + 1); // 产生[start,
end] 内的随机整数
    swap(arr[index], arr[end]);
    int small = start-1;          /// 指示start的前一个
    for(int i= start;i<end;i++){
        if( arr[i] < arr[end]){
            small++;              ///
            if( small != i )      /// 如果碰到小的，而中间又有大的，将大的和当前
小的进行交换
                swap(arr[small],arr[i])
        }
    }

    /// 将最后一个元素交换到它该有的位置
    ++ small;
    swap(arr[small],arr[end]);

    return small;
}

void quick_sort(vector<int>& arr, int start, int end){
    if( start == end ) return;

    int index = partition(arr, 0, end);
    if( index > start )
        quick_sort(arr,0,index-1);

    if( index < end )
        quick_sort(arr,index+1, end);
}
```