1. 快速排序
2. 思路

快速排序的思路是对于序列中的某一元素，调整序列中元素的位置，使得该元素左边的元素都不大于该元素，该元素右边的元素都大于该元素。

1. 解决方法

用two pointers的方法解决，设置两个指针，分别指向序列a的首尾。

1. 令两个下标指向首尾，即left指向a[1]，right指向a[n]，并将a[1]存至temp中。
2. 从序列末端开始，只要a[right]大于temp，则将right不断左移，当a[right]<= temp时，就将a[right]移到a[left]处。
3. 接着是序列首端，只要a[left]<=temp，则将left不断右移，当a[left] > temp时，将a[left[移到a[right]处。
4. 重复2, 3，直到left和right相遇，将temp放到相遇的地方。
5. 代码实现

//对区间进行快速排序

int Partition(int a[], int left, int right) {

int temp = a[left]; //将a[left]放在临时变量中

while (left < right) {

//从右边开始，不断左移

while (left < right && a[right] > temp)

right--;

//从左边开始，不断右移

while (left < right && a[left] <= temp)

left++;

}

//当left和right碰上时，将temp放到该处

a[left] = temp;

return left; //返回最后的下标

}

//快速排序

void quickSort(int a[], int left, int right) {

if (left < right) {

int pos = Partition(a, left, right);

//递归左右两边

quickSort(a, left, pos - 1);

quickSort(a, pos + 1, right);

}

}