快速排序：

//对长度n的整数数列排序

#include <bits/stdc++.h>

using namespace std;

const int N=1e6+10;//限定范围（10^6+10）

int n;//输入长度

int q[N];//含有n个数字的数列

//模板：快速排序quick\_sort

void sort(int q[],int l,int r)//三个参数：数列q[]，左边界l，右边界r

{

    if (l >= r) return;//若左边界比右边界大就直接打断函数避免出错

    int i = l - 1, j = r + 1,x=q[l+r>>1];

    //左指针i从左边界-1开始，右指针同理，定义分界点x

    //此处使用右移符，l+r和并移1，而每右移一位相当于除以2

    while (i<j)//若左指针还未与右针相遇

    {

        do i++; while(q[i] < x);//则i++并当i与x相遇时结束

        do j--; while(q[j] > x);//同理

        if (i < j) swap(q[i],q[j]);//如果i小于j则交换其值

    }

    sort(q,l,j);//递归

    sort(q,j+1,r);//递归

}

int main(){

    scanf("%d",&n);

    for(int k = 0;k < n;k ++) scanf("%d",&q[k]);//循环输入

    sort(q,0,n-1);

    for(int k = 0;k < n;k ++) printf("%d",q[k]);//循环输出

    return 0;

}

//难点：边界分析

进行边界分析是为了避免分治时出现被分为0和n的情况，造成无限分治内存超限问题。

若以j为分界点，

对于quick\_sort(q, l, j), quick\_sort(q, j + 1, r)，

j有可能取l~r的任何一个值，

若j取l，则quick\_sort(q, l + 1, r)执行时会产生分割，不会出现0和n的情况；

若j取r，则quick\_sort(q, l, r)执行时会分割为n，会导致无限分治。

本条结论：若在递归分治前保持j = r，那么就会出现无限分治的情况

所以只要在进入分治前不要让j取到r就可以了。那什么时候会取到r呢？

初始化完毕时，j的值为r + 1，当执行过一次do j--; while(q[j] > x)后，j变为r，并且恰好在此之后j都不会发生改变，即do j--; while(q[j] > x)只会执行一次，

如果保持j = r不变的话，那么i会在此之前一直自增到i = r，此时j = r;

i = r不满足i < j循环结束，此时，整个while(i < j) {}循环只进行了1轮，分治，从而导致分治出了0和n两个情况。

本条结论：若要在递归分治前保持j = r，那么while(i < j) {}只能执行一次

现在把焦点转移到x的取值上，第3点说到，若出现无限分治问题，i会一直自增到i = r，若出现这种情况，那么x的取值一定是q[r]，因为如果x的值不为q[r]，那么一定会在x处存在q[i] == x，而q[i] == x会导致i自增暂时停止，那么就会往下执行，执行do j--;，判断后进入第二轮while(i < j) {}循环，进入第二轮循环会使j自减至少两次，而他的初值为r + 1，也就是说，j的值不会一直保持在j = r上，也就不会导致无限分治。本条结论：若x的取值不为q[r]，那么while(i < j) {}会至少执行两次，因此在进行递归分治前，j的值是一定小于r的。

l + r >> 1的值一定是小于r的，不会取到r，而l + r + 1 >> 1的值一定是大于l的，不会取到l

所以综合2、3、4、5的结论就得出了若以j为分界点，x取q[l + r >> 1]，此时不会出现无限分治的情况；若以i为分界点，x取q[l + r + 1 >> 1]，此时不会出现无限分治的情况

作者：yueluz

链接：https://www.acwing.com/solution/content/26252/

来源：AcWing

著作权归作者所有。商业转载请联系作者获得授权，非商业转载请注明出处。

//思想图

[快排思想图](https://cdn.acwing.com/media/article/image/2019/05/19/1130_1196bbb27a-2.gif)

//stl写法

C++ STL快速算法

#include <bits/stdc++.h>

using namespace std;

const int N=1000000+100;

int a[N],n,m,i,j;

int main()//C++ Stl使人快乐

{

ios::sync\_with\_stdio(false);

cin>>n;

for(int i=1;i<=n;i++)

cin>>a[i];

sort(a+1,a+1+n);

for(int i=1;i<=n;i++)

cout<<a[i]<<" ";

cout<<endl;

}

作者：秦淮岸灯火阑珊

链接：https://www.acwing.com/solution/content/2096/

来源：AcWing

著作权归作者所有。商业转载请联系作者获得授权，非商业转载请注明出处。