第1关：排序

**任务描述**

本关任务：编写一个函数实现将整数序列排序 函数原型：void sort(int a[],int n,int k)； 功能说明：大小为n的整型数组a中元素满足：0≤a[i]≤k，这里，0≤i≤n-1，k≤n。设计一个时间复杂度为O（n）的算法，对a中元素进行递增排序。

**相关知识**

为了完成本关任务，你需要掌握：1.排序算法，2.排序算法的效率分析。 有关定义：

**编程要求**

根据提示，在右侧编辑器补充代码，完成函数sort(int a[],int n,int k)的编写。

**测试说明**

平台会自动读取输入数据，对你编写的代码进行测试，并输出结果。

测试输入：12 10 分别对应n与k值 7 6 5 1 2 6 3 10 6 8 9 4 预期输出：1 2 3 4 5 6 6 6 7 8 9 10

#include "def.h"

#include "stu.h"

int main()

{

int a[100], n, k,i;

scanf("%d%d",&n,&k);

for(i=0;i<n;i++)

    scanf("%d",&a[i]);

sort(a,n,k);

for(i=0;i<n;i++)

    printf("%d ",a[i]);

}

#include "stdio.h"

#include "stdlib.h"

#define TRUE 1

    #define FALSE 0

    #define OK 1

    #define ERROR 0

    #define YES 1

    #define NO 0

    #define INFEASIBLE -1

    #define OVERFLOW -2

    typedef int status;

    typedef int KeyType;

    typedef struct {

         KeyType  key;

         char others[20];

    } TElemType; //二叉树结点类型定义

    typedef struct BiTNode{  //二叉链表结点的定义

          TElemType  data;

          struct BiTNode \*lchild,\*rchild;

    } BiTNode, \*BiTree;

第2关：求中位数

**任务描述**

本关任务：编写一个函数实现求中位数 函数原型：int MidValue(int a[],int n)； 功能说明：求整型数组a的中位数并返回中位数，要求算法效率优于O(n^2)，同时不要将全部的数据进行排序再取中位数，否则扣分30%。

**相关知识**

为了完成本关任务，你需要掌握：1.排序算法，2.排序算法的效率分析。 有关定义：

**编程要求**

根据提示，在右侧编辑器补充代码，完成函数MidValue(int a[],int n)的编写

**测试说明**

平台会自动读取输入数据，对你编写的代码进行测试，并输出结果。

测试输入：10 7 6 5 1 2 3 10 8 9 4 预期输出：5

测试输入：5 17 26 55 11 22 预期输出：22

#include "stdio.h"

#include "stdlib.h"

#include "stu.h"

int main()

{

int ans,\*a,n,i;

scanf("%d",&n);

a=(int \*)malloc(sizeof(int)\*n);

for(i=0;i<n;i++)

    scanf("%d",&a[i]);

printf("%d",MidValue(a,n));

free(a);

return 1;

}