**问题描述**

|  |  |
| --- | --- |
| 试题编号： | 201612-1 |
| 试题名称： | 中间数 |
| 时间限制： | 1.0s |
| 内存限制： | 256.0MB |
| 问题描述： | **问题描述**  　　在一个整数序列*a*1, *a*2, …, *an*中，如果存在某个数，大于它的整数数量等于小于它的整数数量，则称其为中间数。在一个序列中，可能存在多个下标不相同的中间数，这些中间数的值是相同的。 　　给定一个整数序列，请找出这个整数序列的中间数的值。  **输入格式**  　　输入的第一行包含了一个整数*n*，表示整数序列中数的个数。 　　第二行包含*n*个正整数，依次表示*a*1, *a*2, …, *an*。  **输出格式**  　　如果约定序列的中间数存在，则输出中间数的值，否则输出-1表示不存在中间数。  **样例输入**  6 2 6 5 6 3 5  **样例输出**  5  **样例说明**  　　比5小的数有2个，比5大的数也有2个。  **样例输入**  4 3 4 6 7  **样例输出**  -1  **样例说明**  　　在序列中的4个数都不满足中间数的定义。  **样例输入**  5 3 4 6 6 7  **样例输出**  -1  **样例说明**  　　在序列中的5个数都不满足中间数的定义。  **评测用例规模与约定**  　　对于所有评测用例，1 ≤ *n* ≤ 1000，1 ≤ *ai* ≤ 1000。 |
| 代码程序 | **#include<stdio.h>**  **#include<algorithm>**  **#include<vector>**  **using namespace std;**  **typedef struct Node{**  **int data;**  **int less;**  **int great;**  **}Node;**  **bool cmp1(Node n1, Node n2);**  **bool cmp2(Node n1, Node n2);**  **int main()**  **{**  **int n;**  **scanf("%d", &n);**  **Node node;**  **vector<Node> vt;**  **for(int i = 0; i< n; i++)**  **{**  **scanf("%d", &node.data);**  **node.less = 0;**  **node.great = 0;**  **vt.push\_back(node);**  **}**  **sort(vt.begin(), vt.end(), cmp1);**  **for(int i = 1; i<vt.size();i++)**  **{**  **if(vt[i].data != vt[i-1].data)**  **{**  **vt[i].less = i;**  **}else{**  **vt[i].less = vt[i-1].less;**  **}**  **}**    **sort(vt.begin(), vt.end(), cmp2);**  **for(int i = 1; i<vt.size();i++)**  **{**  **if(vt[i].data != vt[i-1].data)**  **{**  **vt[i].great = i;**  **}else{**  **vt[i].great = vt[i-1].great;**  **}**  **}**  **vector<Node>::iterator it;**  **for (it = vt.begin(); it != vt.end(); ++it)**  **{**  **if(it->less == it->great)**  **{**  **break;**  **}**  **}**  **if(it!=vt.end())**  **{**  **printf("%d\n", it->data);**  **}else{**  **printf("-1\n");**  **}**  **}**  **bool cmp1(Node n1, Node n2)**  **{**  **return n1.data < n2.data;**  **}**  **bool cmp2(Node n1, Node n2)**  **{**  **return n1.data>n2.data;**  **}** |