**问题描述**

|  |  |
| --- | --- |
| 试题编号： | 201503-2 |
| 试题名称： | 数字排序 |
| 时间限制： | 1.0s |
| 内存限制： | 256.0MB |
| 问题描述： | **问题描述**  　　给定*n*个整数，请统计出每个整数出现的次数，按出现次数从多到少的顺序输出。  **输入格式**  　　输入的第一行包含一个整数*n*，表示给定数字的个数。 　　第二行包含*n*个整数，相邻的整数之间用一个空格分隔，表示所给定的整数。  **输出格式**  　　输出多行，每行包含两个整数，分别表示一个给定的整数和它出现的次数。按出现次数递减的顺序输出。如果两个整数出现的次数一样多，则先输出值较小的，然后输出值较大的。  **样例输入**  12 5 2 3 3 1 3 4 2 5 2 3 5  **样例输出**  3 4 2 3 5 3 1 1 4 1  **评测用例规模与约定**  　　1 ≤ *n* ≤ 1000，给出的数都是不超过1000的非负整数。 |
| 代码程序： | #include<stdio.h>  #include<algorithm>  #include<string.h>  using namespace std;  typedef struct{  int number;  int count;  }NumberCount;  bool cmp(NumberCount a, NumberCount b);  int main()  {  NumberCount nc[1005];  memset(nc,0,sizeof(nc));  int n;  scanf("%d",&n);  int temp;  for(int i =0;i<n;i++)  {  scanf("%d",&temp);  nc[temp].number = temp;  nc[temp].count++;  }  sort(nc,nc+1001,cmp);  for(int i = 0; i<1000;i++)  {  if(nc[i].count>0)  {  printf("%d %d\n",nc[i].number,nc[i].count);  }else{  break;  }  }  return 0;  }  bool cmp(NumberCount a, NumberCount b)  {  if(a.count!=b.count) return a.count>b.count;  else return a.number < b.number;  } |