排序

1. **计数排序**

**投票：**有n(n<=999)名候选人，每名候选人编号分别从1到n，现在收集到了m(m<200 0000)张选票，每张选票都写了一个候选人编号。现在想把这些选票按照投票数字从小到达排序。输入n和m以及m个选票上的数字，求出排序后的选票编号。

#include<bits/stdc++.h>  
using namespace std;  
int a[1010]={0},n,m,tmp;  
int main(){  
 cin>>n>>m;  
 for(int i=0;i<m;++i){  
 cin>>tmp;  
 ++a[tmp];  
 }  
 for(int i=1;i<=n;++i)  
 for(int j=0;j<a[i];++j)  
 cout<<i<<' ';  
 cout<<endl;  
 return 0;  
}

input:

5 10

2 5 2 2 5 2 2 2 1 2

output:

1 2 2 2 2 2 2 2 5 5

1. **选择排序**

输入n(n<1000)个数字ai(ai<10^9)，将其从小到大排序后输出。

for(int i=0;i<n-1;++i){  
 for(int j=i+1;j<n;++j){  
 if(a[j]<a[i]){  
 int p=a[i];  
 a[i]=a[j];  
 a[j]=p;  
 }  
 }  
}

1. **冒泡排序**

输入n(n<1000)个数字ai(ai<10^9)，将其从小到大排序后输出。

for(int i=0;i<n-1;++i){  
 for(int j=0;j<n-i-1;++j){  
 if(a[j]>a[j+1]){  
 int p=a[j];  
 a[j]=a[j+1];  
 a[j+1]=p;  
 }  
 }  
}

1. **插入排序**

输入n(n<1000)个数字ai(ai<10^9)，将其从小到大排序后输出。

for(int i=1;i<n;++i){  
 int now=a[i],j;//记录一下待插牌，等下还要放回去   
 for(j=i-1;j>=0;--j)  
 if(a[j]>now)  
 a[j+1]=a[j];  
 else break;  
 a[j+1]=now;  
}

1. **快速排序**

输入n(n<10 0000)个数字ai(ai<10^9)，将其从小到大排序后输出。

void qsort(int a[],int l,int r){//引入数组的地址  
 int i=l,j=r,flag=a[(l+r)/2],tmp;  
 do{  
 while(a[i]<flag)++i;//从左找比哨兵大的数   
 while(a[i]>flag)--j;//从右找比哨兵小的数  
 if(i<=j){//交换  
 tmp=a[i];a[i]=a[j];a[j]=tmp;  
 ++i;--j;  
 }  
 }while(i<=j);  
 if(l<j)qsort(a,l,j);  
 if(i<r)qsort(a,i,r);  
}

**sort(a,a+n,cmp)：**对a数组从a[0]到a[n-1]进行排序。cmp是指自定义排序函数，如果是将数组a从小到大排序，那么这一项可以省略。

如果要求从大到小排序，只需要定义一个名字是cmp的自定义比较函数即可，代码如下：

bool cmp(int a,int b){  
 return a>b;  
}

**unique(a,a+n)：**对a数组从a[0]到a[n-1]进行去重，要求a数组已经有序，返回去重后最后一个元素对应的指针。核心代码如下：

sort(a,a+n);  
cnt=unique(a,a+n)-a;  
cout<<cnt<<endl;  
for(int i=0;i<cnt;++i)cout<<a[i]<<' ';