

题目描述：

给定一个数组，编写一个函数来计算它的最大 N 个数与最小 N 个数的和。你需要对数组进行去重。

说明：

*数组中数字范围[0, 1000]

*最大 N 个数与最小 N 个数不能有重叠，如有重叠，输入非法返回-1

*输入非法返回-1

输入描述：

第一行输入 M， M 标识数组大小

第二行输入 M 个数，标识数组内容

第三行输入 N，N 表达需要计算的最大、最小 N 个数

输出描述：

输出最大 N 个数与最小 N 个数的和。

补充说明：

示例 1

输入：

5

95 88 83 64 100

2

输出：

342

说明：

最大 2 个数[100, 95], 最小 2 个数[83, 64], 输出为 342

示例 2

输入：

5

3 2 3 4 2

2

输出:

-1

说明:

最大 2 个数[4, 3], 最小 2 个数[3, 2], 有重叠输出为-1

```
#include <iostream>
#include <vector>
using namespace std;

vector<int> sort(vector<int> &v){ //从小到大排序
    for(int i=0;i<v.size();++i){
        for(int j=0;j<v.size()-i-1;++j){
            if(v[j]>v[j+1]){
                swap(v[j],v[j+1]);
            }
        }
    }
    return v;
}

int main() {
    int M;
    cin>>M;//数组大小
    vector<int> num;//存放数组元素
    for(int i=0;i<M;i++){
        int nn;
        cin>>nn;
        int temp=0;
        for(int j=0;j<num.size();j++){
            if(nn==num[j]){
                temp=1;//表示 num 中存在
            }
        }
        if(temp==0){
            num.push_back(nn);
        }
    }
    int N;
    cin>>N;
    sort(num);
    if(N*2>num.size()){
        cout<<-1<<endl;
    }
}
```

```
    else{
        int sum=0;
        for(int j=0;j<N;j++){
            sum+=num[j]+num[num.size()-1-j];
        }
        cout<<sum<<endl;
    }

}

// 64 位输出请用 printf("%lld")
```