

# Atcoder Rugular Contest 121

[比赛链接](#)

## D - 1 or 2

### 题意

给定  $n$  个数，要求将它们分组。每组可以有  $1 \sim 2$  个数，每组的权重为这组里面的所有数的和。

要求最小化权重最大的组和权重最小的组的权重差。

### 题解

首先考虑如果强制每组必须有两个数，则对任意两组  $(a_1, a_2), (a_3, a_4)$  设  $a_1$  是这四个数中的最小值，则必有  $a_2$  是这四个数中的最大值。

否则假定  $a_4$  是这四个数中的最大值，考虑分组  $(a_1, a_4), (a_2, a_3)$

则有  $\max(a_1+a_4, a_2+a_3) \leq a_3+a_4, \min(a_1+a_4, a_2+a_3) \geq a_1+a_2$  显然更优。

于是设所有数为  $a_1 \leq a_2 \leq \dots \leq a_{2k}$  则最优分组为  $(a_1, a_{2k}), (a_2, a_{2k-1}), \dots, (a_k, a_{k+1})$

一个数一组相当于这个数和  $0$  一组。于是可以枚举向原序列中加入  $0 \sim n$  个  $0$  的情况，然后按上述方法分组计算答案。

时间复杂度  $O(n^2)$

```
const int MAXN=5e3+5,inf=2e9;
int a[MAXN<<1];
int main()
{
    int n=read_int();
    for(i,0,n)a[i]=read_int();
    sort(a,a+n);
    int ans=inf;
    rep(i,n,n*2){
        if(i%2==0){
            int maxv=-inf,minv=inf;
            for(int j=0;j<i/2;j++){
                maxv=max(maxv,a[j]+a[i-1-j]);
                minv=min(minv,a[j]+a[i-1-j]);
            }
            ans=min(ans,maxv-minv);
        }
    }
}
```

```
int pos=0;
while(pos<i&&a[pos]<=0)pos++;
for(int j=i-1;j>=pos;j--)
a[j+1]=a[j];
a[pos]=0;
}
enter(ans);
return 0;
}
```

## E - Directed Tree

### 题意

### 题解

From:  
<https://wiki.cvbbacm.com/> - CVBB ACM Team

Permanent link: [https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:legal\\_string:jxm2001:contest:arc\\_121&rev=1625109892](https://wiki.cvbbacm.com/doku.php?id=2020-2021:teams:legal_string:jxm2001:contest:arc_121&rev=1625109892)

Last update: 2021/07/01 11:24