C. Pairing Resistors

Time Limit: $4\texttt{s}$

Memory Limit: $1024\texttt{MB}$

Volt designed a test problem about resistor pairing.

The testing device consists of n parallel wires. There are $2n$ resistors, each with a fixed resistance value $a_1, a_2, \ldots, a_{2n}$.

The engineer's task is to distribute these resistors across the wires so that each wire has exactly two resistors. The equivalent resistance of each wire is **the sum** of the resistance values of the two resistors assigned to it.

For each possible pairing of resistors, we compute the equivalent resistances of the n wires and sort them in ascending order.

Let b_i be the i-th smallest equivalent resistance after sorting. For every $i \in \{1, 2, \ldots, n\}$, Volt wants to know the total sum of b_i over all possible pairing schemes. Output the answer modulo 998244353.

Two matching schemes are considered different if and only if there exists at least one resistor that is paired differently in the two schemes. It is easy to see that there are a total of $\dfrac{(2n)!}{n!2^n}$ distinct matching schemes.

Input Format

- The first line contains an integer n.
- The second line contains $2n$ integers $a_1, a_2, \ldots, a_{2n}$.

Output Format

- A single line containing n integers.

Sample 1

input

```
2
3 1 4 2
```

output

```
12 18
```

Explanation

- First pairing scheme: $\{a_1 + a_2, a_3 + a_4\}$. Equivalent resistances before sorting: $[4, 6]$, after sorting: $[4, 6]$.
- Second pairing scheme: $\{a_1 + a_3, a_2 + a_4\}$. Equivalent resistances before sorting: $[7, 3]$, after sorting: $[3, 7]$.
- Third pairing scheme: $\{a_1 + a_4, a_2 + a_3\}$. Equivalent resistances before sorting: $[5, 5]$, after sorting: $[5, 5]$.
- The total sum of b_1 over all pairing schemes is $4 + 3 + 5 = 12$.
- The total sum of b_2 over all pairing schemes is $6 + 7 + 5 = 18$.

Sample 2

input

```
3
1 1 1 1 2 2
```

output

```
30 42 48
```

Sample 3-7

See the download attachment for details.

Constraints

For 100% of the test cases:

- $1 \leq n \leq 400$
- $1 \leq a_i \leq 10^8$

Subtask ID	$n \leq$	Points
1	8	10
2	15	15
3	30	20
4	100	25
5	400	30