Problem G. Trans

Input file:	`standard input`
Output file:	`standard output`
Time limit:	2 seconds
Memory limit:	512 mebibytes

Bob is interested in popcount and some strange transforms. Currently, he is attacking the following problem:

There is an array of 2^n integers $a_0, a_1, a_2, \ldots, a_{2^n-1}$. The task is, for each i $(0 \leq i \leq 2^n - 1)$, to calculate

$$b_i = \sum_{j=0}^{2^n-1} (\mathrm{popcount}(i \text{ and } j) \bmod 2) \cdot a_j,$$

where "popcount(x)" denotes the number of ones in the binary representation of x, and "and" denotes the bitwise AND operation.

Although Bob is very smart, he still can't solve the problem fast. Can you help him calculate all b_i?

Input

The first line contains a single integer n $(1 \leq n \leq 20)$.

The second line contains 2^n integers describing the array a $(1 \leq a_i \leq 10^9)$.

Output

Print one line with 2^n integers, the i-th of them being the value b_i.

Example

standard input	standard output
2 1 2 3 4	0 6 7 5