

Journey

Input file: **standard input**
Output file: **standard output**
Time limit: 2 seconds
Memory limit: 1024 megabytes

Ritmo and *Fisica* are building a tuning device. A tuning device consists of a sequence a in which every integer in $1 \sim n$ occurs exactly twice, and a **regular bracket sequence** s containing n pairs of brackets. A tuning device works correctly if and only if it satisfies the following conditions:

- Create one vertex for each index.
- For every pair of matching brackets whose indices in s are (l_i, r_i) , add an edge between these two vertices.
- For every value in a , let (x_i, y_i) be the indices of its two occurrences, and add an edge between these two vertices.
- The procedure above produces one cycle containing all $2n$ vertices and all $2n$ edges.

Ritmo has already installed a . Now you need to help *Fisica* find a corresponding sequence s . It can be proven that such a regular bracket sequence always exists.

Recall that a bracket sequence is **regular** if and only if it can be constructed using the following rules:

- $()$ is a regular bracket sequence.
- If A is a regular bracket sequence, then $(+ A +)$ is a regular bracket sequence.
- If A, B are regular bracket sequences, then $A + B$ is a regular bracket sequence.

Input

This problem contains multiple test cases.

The first line contains a positive integer T ($1 \leq T \leq 10^4$), representing the number of test cases.

For each test case:

- The first line contains an integer n ($2 \leq n \leq 2 \times 10^5$).
- The second line contains $2n$ integers a_i ($1 \leq a_i \leq n$).

It is guaranteed that the sum of n over all test cases does not exceed 2×10^5 , and that every integer in $1 \sim n$ occurs exactly twice in a .

Output

For each test case, output a regular bracket sequence s of length $2n$ representing a valid answer.

Example

standard input	standard output
5	()()
2	(())
1 2 1 2	()()()
2	()()()()
2 2 1 1	()()()()()()()
3	
1 2 3 1 2 3	
4	
1 2 3 4 3 2 1 4	
7	
3 6 7 1 5 6 2 2 4 3 4 7 1 5	