

## Cardboard Boxes Game

You came back from a shoes store. After putting away all the shoes, you are left with a lot of cardboard boxes, which you want to save for later use. So you take one bag and stuff all the others into it.

In doing so, you realize that there are various ways of nesting the boxes, with all boxes viewed as identical.

If we use a matching pair of parentheses to represent a box, the ways are:

For 1 box, there's one way:

**[ ]      <- a box**

For 2 boxes, there is one way:

**[ [ ] ]    <- one box in another**

For 3 boxes, there is one way:

**[ [ [ ] ] ] <- 3 boxes nested Russian doll style**

**[ [ ] [ ] ] <- 2 boxes side by side, inside the third**

For 4 boxes, four:

**[ [ [ [ ] ] ] ]**

**[ [ [ ] ] [ ] ]**

**[ [ [ ] ] [ ] ]**

**[ [ [ ] ] [ ] ]**

*Note that because all boxes are identical, the two 4-box strings [ [ [ [ ] ] ] ] and [ [ [ ] ] [ ] ] represent the same configuration.*

It's easy to see that each configuration for n boxes represents a n-node rooted tree, where a box is a tree node, and a box with its content forms a subtree. The outermost box is the tree root.

Number of configurations for given n is given by OEIS A81.

### Task

Write a program that, when given **N**, enumerates all ways of nesting n boxes. You should use the parentheses notation above.

This task asks for enumeration of trees only; for counting solutions without enumeration, that OEIS page lists various formulas, but that's not encouraged by this task, especially if implementing it would significantly increase code size.

### Tips

*You could write 3 functions:*

- *Boxes()*
- *Boxchain()*
- *Replace\_brackets()*

**Input**

The first line of input contains a integer N the ways of nesting n boxes.

**Output**

The output displays N lines, each with a way of nesting n boxes.

**Sample input**

**Sample output**

3	[[[]]] [[[]]]
5	[[[[[]]]]] [[[[[]]]]] [[[[[]]]]] [[[]][[]]] [[[]][[]]] [[[]][[]]] [[[]][[]]] [[[]][[]]] [[[]][[]]] [[[]][[]]]
14	Wrong