

Basic Problem Solving on Recursion

Relevel
by Unacademy



List of problems involved

- Tiling Problem Recursively
- Subsequences Recursively
- Generate Parentheses Recursively
- Lexicographical Printing Recursively
- Count Maze Path
- Permutations Recursively



Tiling Problem

What is a Tiling Problem ?

Given a “2 x n” tile board with fixed width of 2 units and the length of the board is ‘n’ units, the area of the board is $n \times 2$ sq. units and tiles of size 2 x 1 sq. units.

The Tiling Problem asks us to find out in how many ways we can arrange these tiles to fully cover the tile board.
(A tile can either be placed horizontally i.e as a 1 x 2 tile or vertically i.e as 2 x 1 tile

Example Test Case:

For N = 2, i.e a grid of 2 x 2 there are two ways you can fill the 2 x 2 size grid using tiles of size 2 x 1 and 1 x 2



Tiling Problem

One way is fill both of the columns with two vertical tiles this would be one configuration,
Other one is you can fill it using two horizontal tiles. So there are two possible ways
For $N = 2$ i.e 2×2 would be 2.

Example Test Case:

For $N = 3$,

If we are given a grid of size 2×3 , then there are three possible ways you can fill the grid using horizontal and vertical tiles

First one is you have used all three vertical tiles and the second configuration is first vertical and then two horizontal tiles and the third one is two horizontal tiles first and then a vertical tile

So, for $N = 3$ i.e 2×3 there are only three possible ways you can fill the grid using vertical or horizontal tiles

Tiling Problem



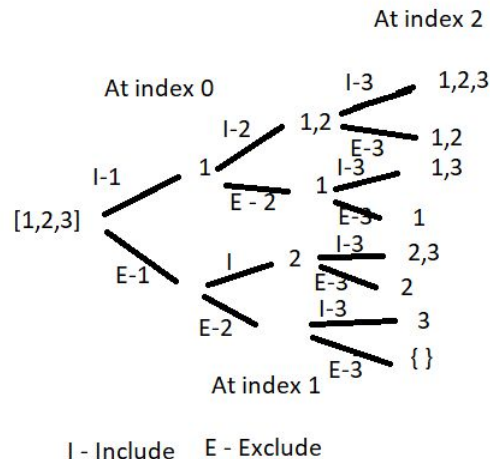
Code link: <https://codesandbox.io/s/autumn-microservice-9yj19?file=/src/index.js>

Subsequences Recursively

SubSequence: May not be contiguous but maintain the relative order.

Elements in the subsequence appear in the same order as they appear in the original array the only difference is that they may not be contiguous

The recursion tree for the array [1,2,3] is



Code link: <https://codesandbox.io/s/floral-monad-bt7tk?file=/src/index.js>

Generate Parentheses

Problem statement

You are given a 'n' pair of parentheses, write a function to generate all combinations of well formed parentheses

For n = 1, ["()"]

For n = 2, then we have to return the combinations of all parentheses

```
[ "()",  
  "()" ]
```

For n = 3

```
[ "((()))",  
  "(()())",  
  "(())()",  
  "()()()",  
  "()(())" ]
```

Generate Parentheses

Code link: <https://codesandbox.io/s/dreamy-archimedes-u3g6s?file=/src/index.js:798-834>

Lexicographical Printing Recursively

Given an number N, the task is to print all the numbers from 1 to N in lexicographical order.

Input N = 14

Output: 1 10 11 12 13 14 2 3 4 5 6 7 8 9

Input: N = 19

Output: 1 10 11 12 13 14 15 16 17 18 19 2 3 4 5 6 7 8 9

Input N= 15

Output: 1 10 11 12 13 14 15 2 3 4 5 6 7 8 9

Code link: <https://jsfiddle.net/fsw80bcn/>

Count maze problem

Here we have a two dimensional array it could be array of characters or integers but usually there will be ones and zeros

Where 0 is the valid path and -1 would be like the wall and here we use recursion to iterate over this array, here we need to find out the starting point, usually the starting point will be off (0,0) and ending at the end of the array. So you have to find the way by figuring out which path has 0 and then take the path and if you get a like a wall or there's nowhere else to go then you have to go back to where you started

Code link: <https://jsfiddle.net/fcatqb0k/>

Permutations Recursively

Given a string, print all the possible permutations of that string
permutations of a string means generating all possible words that could be formed using the characters in the provided string

For example:

If the string is AB, possible words would be AB and BA

If the string is ABC, possible combinations would be ABC, ACB, BAC, CAB, BCA, CBA

Code link: <https://jsfiddle.net/1c39jews/>

Practice H/W

1. Write a JavaScript program to compute the exponent of a number. Note : The exponent of a number says how many times the base number is used as a factor.

$8^2 = 8 \times 8 = 64$. Here 8 is the base and 2 is the exponent.

1. Write a JavaScript program for binary search.

Sample array: [0,1,2,3,4,5,6]

`console.log(l.br_search(5))` will return '5'

Thankyou!