RECURSION STRATEGY:

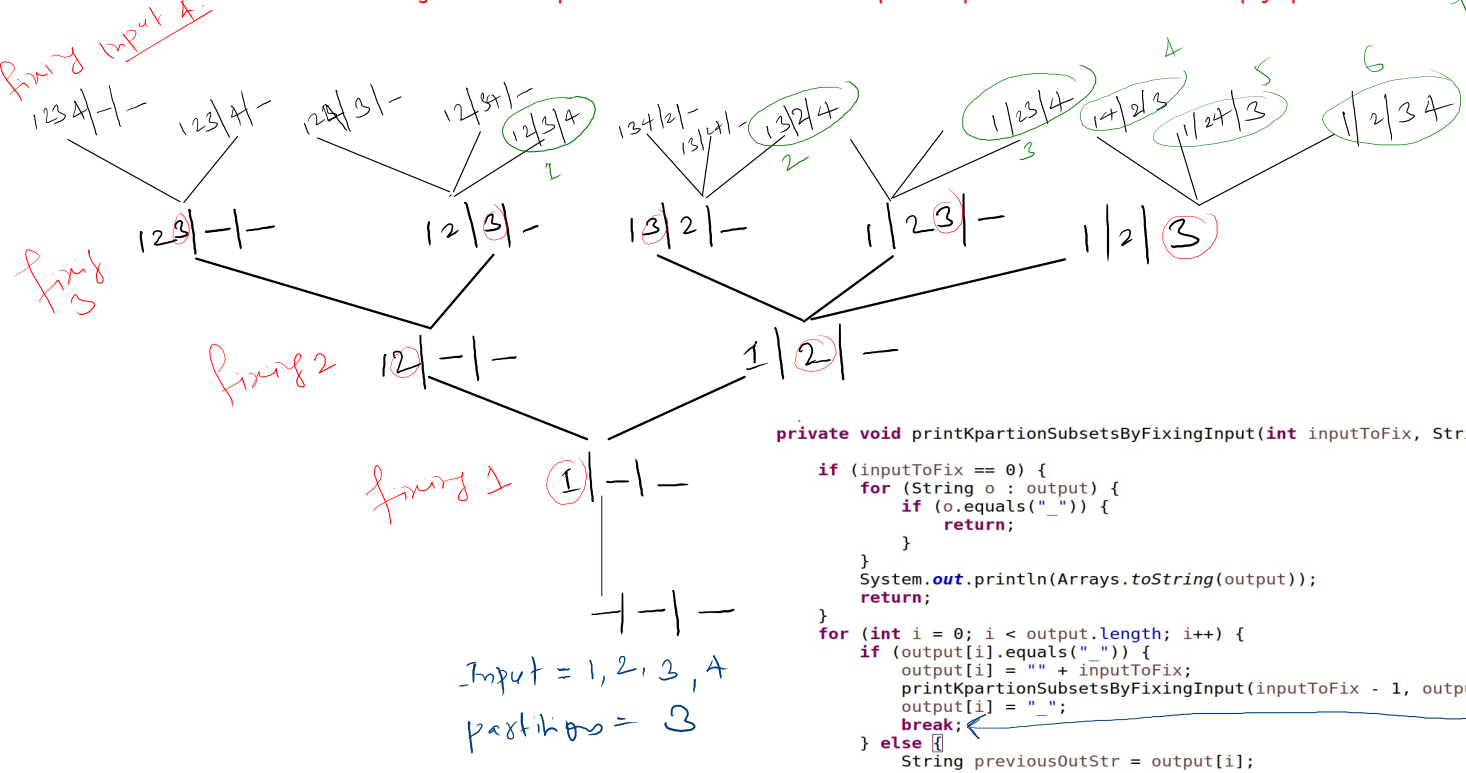
1. Input is getting fixed at each level of tree.
2. Partitions are tried as option at each level.

Note:

1. While fixing a given_input we should treat all the empty partitions as same partition to avoid the permutation generation. So we should not spawn new branch for each empty partition rather just spawn single branch against first empty partition.

if we use first empty partition then all the empty partitions will come at end

2. If we wish to generate the permutations then we should spawn separate branch for each empty space.




```
private void printKpartionSubsetsByFixingInput(int inputToFix, String[] output) {

    if (inputToFix == 0) {
        for (String o : output) {
            if (o.equals("_")) {
                return;
            }
        }
        System.out.println(Arrays.toString(output));
        return;
    }
    for (int i = 0; i < output.length; i++) {
        if (output[i].equals("_")) {
            output[i] = "" + inputToFix;
            printKpartionSubsetsByFixingInput(inputToFix - 1, output);
            output[i] = "_";
            break;
        } else {
            String previousOutStr = output[i];
            output[i] += inputToFix;
            printKpartionSubsetsByFixingInput(inputToFix - 1, output);
            output[i] = previousOutStr;
        }
    }
}
```