

# 017. Letter Combinations of a Phone Number

## 017 Letter Combinations of a Phone Number

- BackTracking

### Description

Given a digit string, return all possible letter combinations that the number could represent.

A mapping of digit to letters (just like on the telephone buttons) is given below.



**Input:** Digit string "23"

**Output:** ["ad", "ae", "af", "bd", "be", "bf", "cd", "ce", "cf"].

### Note:

Although the above answer is in lexicographical order, your answer could be in any order you want.

### 1. Thought line

### 2. BackTracking

```
1 class Solution {
2 public:
3     vector<string> letterCombinations(string digits) {
4         vector<string> result;
5         if(digits.empty()) return vector<string>();
6         static const vector<string> v = {"", "", "abc", "def", "ghi", "jkl", "mno",
7             "pqrs", "tuv", "wxyz"};
8
9         result.push_back(""); // add a seed for the initial case
10        for(int i = 0 ; i < digits.size(); ++i) {
11            int num = digits[i] - '0';
12            if(num < 0 || num > 9) break;
13
14            const string& candidate = v[num];
15            if(candidate.empty()) continue;
16
17            vector<string> tmp;
18            for(int j = 0 ; j < candidate.size() ; ++j) {
19                for(int k = 0 ; k < result.size() ; ++k) {
20                    tmp.push_back(result[k] + candidate[j]);
21                }
22            }
23            result.swap(tmp);
24        }
25        return result;
26    }
```

