# **077.** Combinations

# **077 Combinations**

• Backtracking

## **Description**

Given two integers n and k, return all possible combinations of k numbers out of 1 ... n.

For example,

If n = 4 and k = 2, a solution is:

```
[
[2,4],
[3,4],
[2,3],
[1,2],
[1,3],
[1,4],
]
```

### 1. Thought line

#### 2. Backtracking

```
class Solution {
private:
    void backTrackingCombine(int st, int& ed, int& k, vector<int>& temp, vector<vector<int>>& result){
        if (temp.size()==k){
            result.push_back(temp);
            return;
        }
        if (st>ed || temp.size()>k) return;

        for (int i = st; i==ed; ++i){
            temp.push_back(i);
            backTrackingCombine(i+1, ed, k, temp, result);
            temp.pop_back();
        }
    }

public:
    vector<vector<int>> combine(int n, int k) {
        vector<vector<int>> combine(int n, int k) {
            vector<vector<int>> result(0);
            backTrackingCombine(1, n, k, temp, result);
            return result;
    }
};
```