1. Combinations

Medium

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

**Example:**

Input: n = 4, k = 2  
Output:  
[  
 [2,4],  
 [3,4],  
 [2,3],  
 [1,2],  
 [1,3],  
 [1,4],  
]

**Solution**

dfs

class Solution {  
public:  
 vector<vector<int>> combine(int n, int k) {  
 vector<vector<int>>ans;  
 vector<int>tmp;  
 bool flag[n + 1] = {false};  
 dfs(ans, tmp, 0, n, k, flag);  
 return ans;  
 }  
 void dfs(vector<vector<int>>&ans, vector<int>&tmp, int curIndex, int n, int k, bool flag[]){  
 if(tmp.size() == k){  
 ans.push\_back(tmp);  
 return;  
 }  
 for(int i = curIndex + 1; i <= n; ++i){  
 if(flag[i] == true)continue;  
 tmp.push\_back(i);  
 flag[i] = true;  
 dfs(ans, tmp, i, n, k, flag);  
 tmp.pop\_back();  
 flag[i] = false;  
 }  
 }  
};