**PROBLEM STATEMENT**:

Given two integers n and k, return all possible combinations of k numbers out of 1 2 3 ... n.

Make sure the combinations are **sorted**.

To elaborate,

1. Within every entry, elements should be sorted. [1, 4] is a valid entry while [4, 1] is not.
2. Entries should be sorted within themselves.

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

[

[1,2],

[1,3],

[1,4],

[2,3],

[2,4],

[3,4],

]

**CODE:**

void util(vector<vector<int>>&ans,vector<int>part,int n,int k,int idx)

{

if(k==0)

{

ans.push\_back(part);

return;

}

for(int i=idx;i<=n;i++)

{

part.push\_back(i);

util(ans,part,n,k-1,i+1);

part.pop\_back();

}

}

vector<vector<int> > Solution::combine(int A, int B) {

vector<vector<int>>ans;

vector<int>part;

util(ans,part,A,B,1);

return ans;

}