Graph and Vertices

Easy Accuracy: 81.04% Submissions: 2K+ Points: 2

Given an integer **n** representing number of vertices. Find out how many **undirected graphs** (not necessarily connected) can be constructed out of a given n number of vertices.

Example 1:

Input: 2
Output: 2

Example 2:

Input: 5
Output: 1024

C++ (g++ 5.4) Average Time: 15m Start Timer

```
// } Driver Code Ends
// User function Template for C++
class Solution {
  public:
    long long count(int n) {
        // your code here
        int x = (n*(n-1))/2;
        long long res = pow(2,x);
        return res;
    }
};
// } Driver Code Ends
```

Custom Input Compile & Run Submit

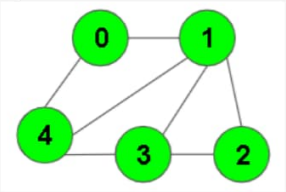
Print adjacency list

Easy Accuracy: 43.42% Submissions: 76K+ Points: 2

Given the adjacency list of a bidirectional graph. Your task is to copy/clone the adjacency list for each vertex and return a new list.

Example 1:

Input:



Output:

C++ (g++ 5.4) Start Timer

```
// } Driver Code Ends
class Solution {
  public:
    // Function to return the adjacency list for each vertex.
    vector<vector<int>> printGraph(int V, vector<int> adj[]) {
        // Code here
       vector<vector<int>>res(V);
        for(int i =0;i<V;i++){
            res[i].push_back(i);
            for(int j =0;j<adj[i].size();j++){
                res[i].push_back(adj[i][j]);
            }
        }
        return res;

    }
};
// } Driver Code Ends
```

Custom Input | Compile & Run | Submit