**Find Kth permutation :-**

Medium Accuracy: 68.28% Submissions: 10K+ Points: 4

Given two integers **N**(1<=N<=9) and **K**. Find the kth permutation sequence of first N natural numbers. Return the answer in **string** format.

**Example 1:**

**Input:** N =4, K = 3

**Output:** 1324

**Explanation:**

Permutations of first 4 natural numbers:

1234,1243,1324,1342,1423,1432.....

So the 3rd permutation is 1324.

**Example 2:**

**Input:** N = 3, K = 5

**Output:** 312

**Explanation:**

Permutations of first 3 natural numbers:

123,132,213,231,312,321.

So the 5th permutation is 312.

**Your Task:**  
You don't need to read input or print anything. Your task is to complete the function **kthPermutation()**which takes two integers N and K as input parameters and returns a string denoting the kth permutation.

**Expected Time Complexity:** O(N2)  
**Expected Auxiliary Space:** O(N)

**Constraints:**  
1 <= N <= 9  
1 <= K <= N!

**Code :-**

//{ Driver Code Starts

//Initial Template for C++

#include <bits/stdc++.h>

using namespace std;

// } Driver Code Ends

//User function Template for C++

class Solution

{

public:

string kthPermutation(int n, int k)

{

vector<int> v(n);

for(auto i=0; i<n; i++)

v[i] = i+1;

for(auto i=1; i<k; i++){

next\_permutation(v.begin(), v.end());

}

string ans="";

for(auto i:v){

ans = ans + to\_string(i);

}

return ans;

}

};

//{ Driver Code Starts.

int main()

{

int t;

cin >> t;

while (t--)

{

int n, k;

cin >> n >> k;

Solution ob;

string ans = ob.kthPermutation(n, k);

cout << ans;

cout << "\n";

}

return 0;

}

// } Driver Code Ends

**T.C :- O(n2)**

**S.C :- O(n)**