



Experiment3.2

NAME – RAJDEEP JAISWAL

UID 20BCS2761

BRANCH-BTECH CSE

SUB- CC LAB

1. **Aim:** ⇒ To implement the concept of backtracking.

2. **Objective:**

- ⇒ The objective is to build problem solving capability and to learn the basic concepts of data structures.
- ⇒ The implementation of Combination which shows and brushes up the concept of greedy.

3. **Leetcode code and output:**

□ **COMBINATION CODE=**

```
class Solution { private: void combine(int n, int k, vector<vector<int>>
&output, vector<int> &temp, int start){ if(temp.size() == k){
output.push_back(temp); return;
} for(int i=start;
i<=n; i++){
temp.push_back(i); combine(n, k,
output, temp, i+1); temp.pop_back();
}
} public:
vector<vector<int>> combine(int n, int k) {
vector<vector<int>> output; vector<int>
temp;
combine(n, k, output, temp, 1);
return output;
} };
```



DEPARTMENT OF

COMPUTER SCIENCE & ENGINEERING

OUTPUT=

LeetCode Problem 77: Combinations

Description: Given two integers n and k , return all possible combinations of k numbers chosen from the range $[1, n]$. You may return the answer in any order.

Example 1: $n = 4, k = 2$

```
class Solution {
private:
void combine(int n, int k, vector<vector<int>> &output, vector<int> &temp, int start){
if(temp.size() == k){
output.push_back(temp);
return;
}
}
```

Testcase Result: Accepted (Runtime: 0 ms)

LeetCode Submission Details

Accepted

Next question

More challenges: 39. Combination Sum, 46. Permutations

Accepted a few seconds ago (C++)

User: HARSHIDA SHAILY (9410, 201167)

Runtime: 26 ms, Memory: 8.9 MB

Beats: 73.14% (Runtime), 90.34% (Memory)

□ SUBETS



CODE

```
class Solution { public:  
    void pS(int ind,vector<int> &nums,vector<int> &v,vector<vector<int>> &ans){  
        if(ind==nums.size()){  
            ans.push_back(v);  
            return;  
        }  
        v.push_back(nums[ind]);  
        pS(ind+1,nums,v,ans);  
        v.pop_back();  
        pS(ind+1,nums,v,ans);  
    }  
    vector<vector<int>>  
    subsets(vector<int>& nums) {  
        vector<vector<int>> ans;        vector<int> v;  
        pS(0,nums,v,ans);        return ans;  
    } };
```

OUTPUT=

The screenshot shows the LeetCode interface for problem 78. Subsets. The problem description states: "Given an integer array nums of unique elements, return all possible subsets (the power set). The solution set must not contain duplicate subsets. Return the solution in any order." Example 1: Input: nums = [1,2,3], Output: [[], [1], [2], [1,2], [3], [1,3], [2,3], [1,2,3]]. The code editor shows the C++ solution provided above. The test case result is "Accepted" with a runtime of 4 ms. The input field contains "[1,2,3]" and the console is empty.



DEPARTMENT OF

Discover. Learn. Empower.

COMPUTER SCIENCE & ENGINEERING

The screenshot shows a LeetCode submission page for the problem "Letter Case Permutation" (ID 784). The submission is marked as "Accepted" and was made by user "HARSHIDA SHAILY" using C++.

Submission Details:

- Status: Accepted
- Language: C++
- Runtime: 0 ms
- Memory: 6.9 MB
- Beats: 100%

The page also shows a distribution chart for the "Beats" metric, indicating that the submission is faster than 100% of other submissions. The submission was made "a few seconds ago".