6.16 SUBSET GENERATION WITH CONSTRAINT

Question:

Write a program to implement the concept of subset generation. Given a set of unique integers and a specific integer 3, generate all subsets that contain the element 3. Return a list of lists where each inner list is a subset containing the element 3.

Example: E = [2, 3, 4, 5], x = 3
The subsets containing 3: [3], [2, 3], [3, 4], [3, 5], [2, 3, 4], [2, 3, 5], [3, 4, 5], [2, 3, 4, 5]

Additionally, 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]]

Example 2:
Input: nums = [0]
Output: [[],[0]]

AIM

To implement a Python program that generates all subsets of a given set and additionally filters subsets containing a specific element.

ALGORITHM

1. Sort the input set to maintain lexicographical order.
2. Use backtracking to generate all subsets:
   - At each step, decide whether to include the current element in the subset.
   - Recurse until all elements are considered.
3. Store each subset generated.
4. To generate subsets containing a specific element (e.g., 3), filter the generated subsets to include only those containing the element.
5. Return the complete power set or the filtered set of subsets as required.

PROGRAM

```
def subsets_with_element(E, x):
    result = []
    def backtrack(start, path):
        if x in path:
            result.append(path[:])
        for i in range(start, len(E)):
            path.append(E[i])
            backtrack(i + 1, path)
            path.pop()
    backtrack(0, [])
    return result

E = list(map(int, input("Enter set elements: ").split()))
x = int(input("Enter required element: "))
print("Subsets containing", x, ":")
for subset in subsets_with_element(E, x):
    print(subset)
```

Input:

E = [2, 3, 4, 5], x = 3
nums = [1, 2, 3]

Output:

```
Enter set elements: 2 3 4 5
Enter required element: 3
Subsets containing 3 :
[2, 3]
[2, 3, 4]
[2, 3, 4, 5]
[2, 3, 5]
[3]
[3, 4]
[3, 4, 5]
[3, 5]
>>>
```

RESULT:

Thus, the program is successfully executed and verified to generate all subsets of a set and filter subsets containing a specified element.

PERFORMANCE ANALYSIS:

Time Complexity: O(2^n) where n is the number of elements, since each element can either be included or excluded.

Space Complexity: O(n) for recursion depth and temporary path storage.