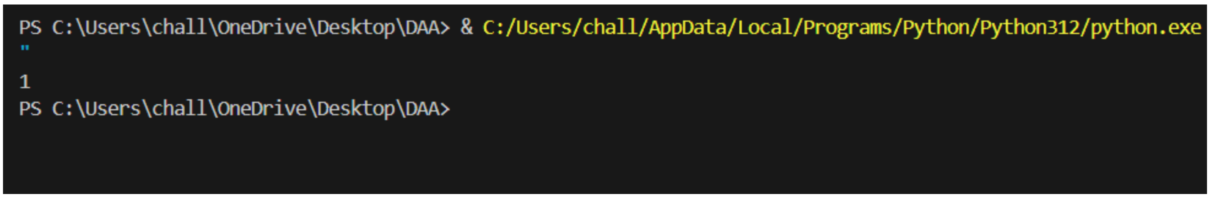
39. Number of Ways to Wear Different Hats to Each Other There are n people and 40 types of hats labeled from 1 to 40. Given a 2D integer array hats, where hats[i] is a list of all hats preferred by the ith person. Return the number of ways that the n people wear different hats to each other. Since the answer may be too large, return it modulo 109 + 7. Example 1: Input: hats = [[3,4], [4,5], [5]] Output: 1 Explanation: There is only one way to choose hats given the conditions. First person choose hat 3, Second person choose hat 4 and last one hat 5. Example 2: Input: hats = [[3,5,1], [3,5]] Output: 4 Explanation: There are 4 ways to choose hats: (3,5), (5,3), (1,3) and (1,5)

PROGRAM:

```
def countWaysToWearHats(hats):
    MOD = 10**9 + 7
    n = len(hats)
    dp = [0] * (1 << n)
    dp[0] = 1
    hat_to_people = [[] for _ in range(41)]
    for i in range(n):
        for hat in hats[i]:
            hat_to_people[hat].append(i)
    for hat in range(1, 41):
        for state in range((1 << n) - 1, -1, -1):
            for person in hat_to_people[hat]:
                if state & (1 << person):
                    continue
                dp[state | (1 << person)] += dp[state]
                dp[state | (1 << person)] %= MOD
    return dp[(1 << n) - 1]
```

```
hats1 = [[3, 4], [4, 5], [5]]
print(countWaysToWearHats(hats1))
```

OUTPUT:



```
PS C:\Users\chall\OneDrive\Desktop\DAA> & C:/Users/chall/AppData/Local/Programs/Python/Python312/python.exe
"
1
PS C:\Users\chall\OneDrive\Desktop\DAA>
```

TIME COMPLEXITY:

Time complexity for the above code is

F(n)=O(2n+n)