Coin Change Permutations

Friday, 30 April 2021 7:01 PM

coins → {2, 3, 5, 6} → infinites

target → 7

Smart calls

Base case → (target == 0) print the ans

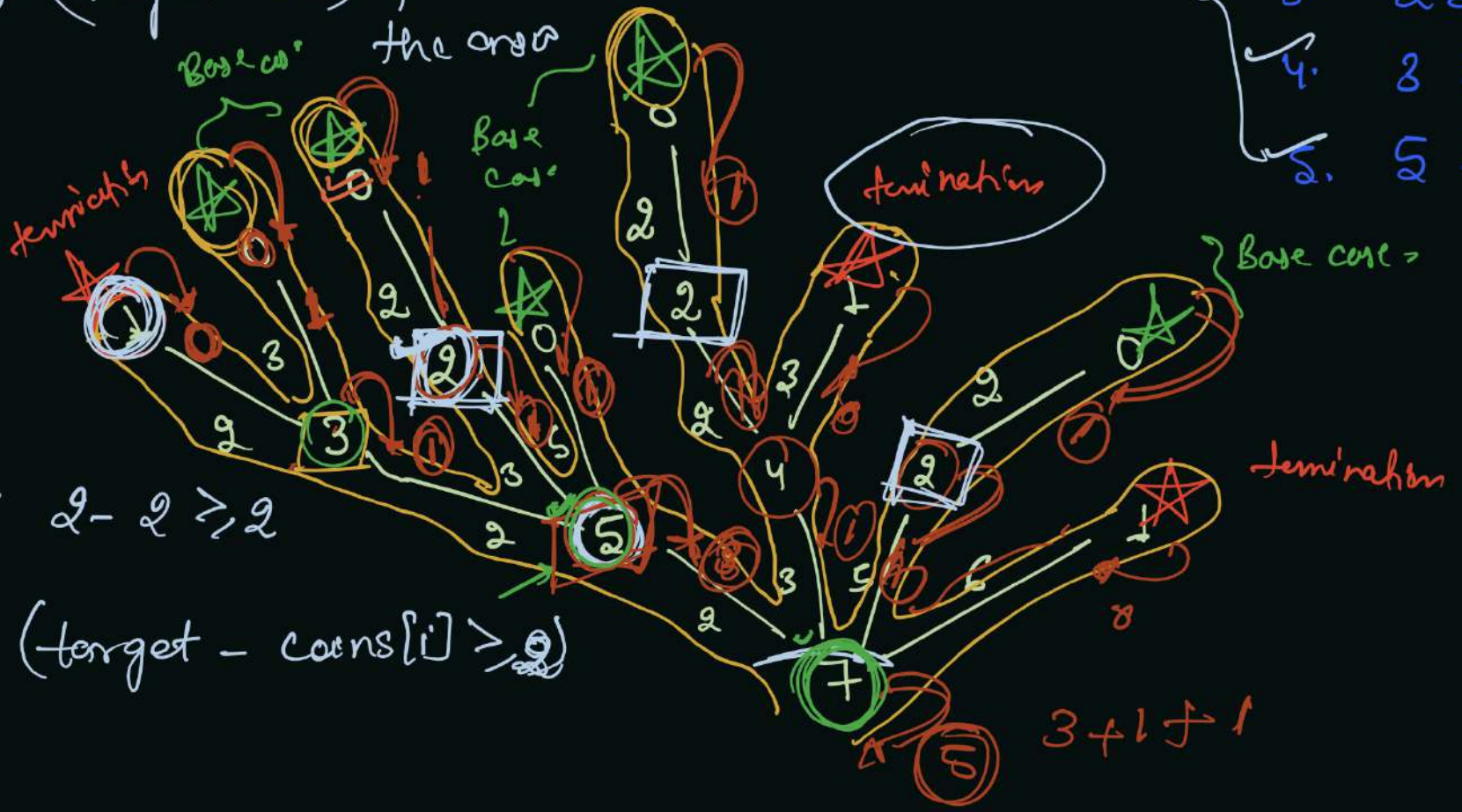
termination case →

Depend on smart calls: 2 - 2 >= 2

Call → (target - coins[i] >= 0)

10 - 5 >= 0

1. 2 2 3
2. 2 3 2
3. 2 5
4. 3 2 2
5. 5 2



Memoisation using DP DP = new int[target+1];

0	1	2	3	4	5	6	7
1	0	1	1	1	3		5

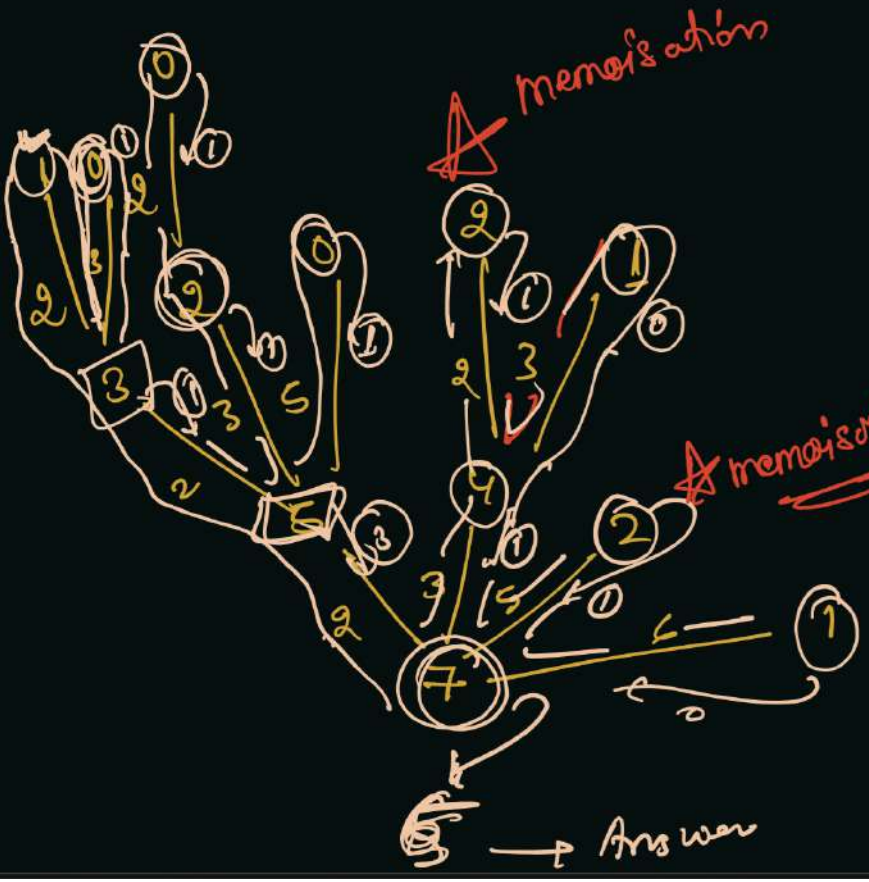
```
public static int coinChangePerm_memo(int[] coins, int target, int[] dp) {
    if(target == 0) {
        return dp[target] = 1;
    }

    if(dp[target] != 0) {
        return dp[target];
    }

    int count = 0;
    for(int i = 0; i < coins.length; i++) {
        int coin = coins[i];
        if(target - coin >= 0) {
            count += coinChangePerm_memo(coins, target - coin, dp);
        }
    }
    return dp[target] = count;
}
```

target = 5

2, 3, 5



coins → {2, 3, 5, 6}

dp[7-3] + dp[4]

target +1

Coins → 2, 3, 5, 6

No. of ways to prepare target 4

target →	0	1	2	3	4	5	6	7
	1	0	1	1	1	3	3	5
	-	-	2	3	22	32 23 5	222 33 6	322 232 62 223 25

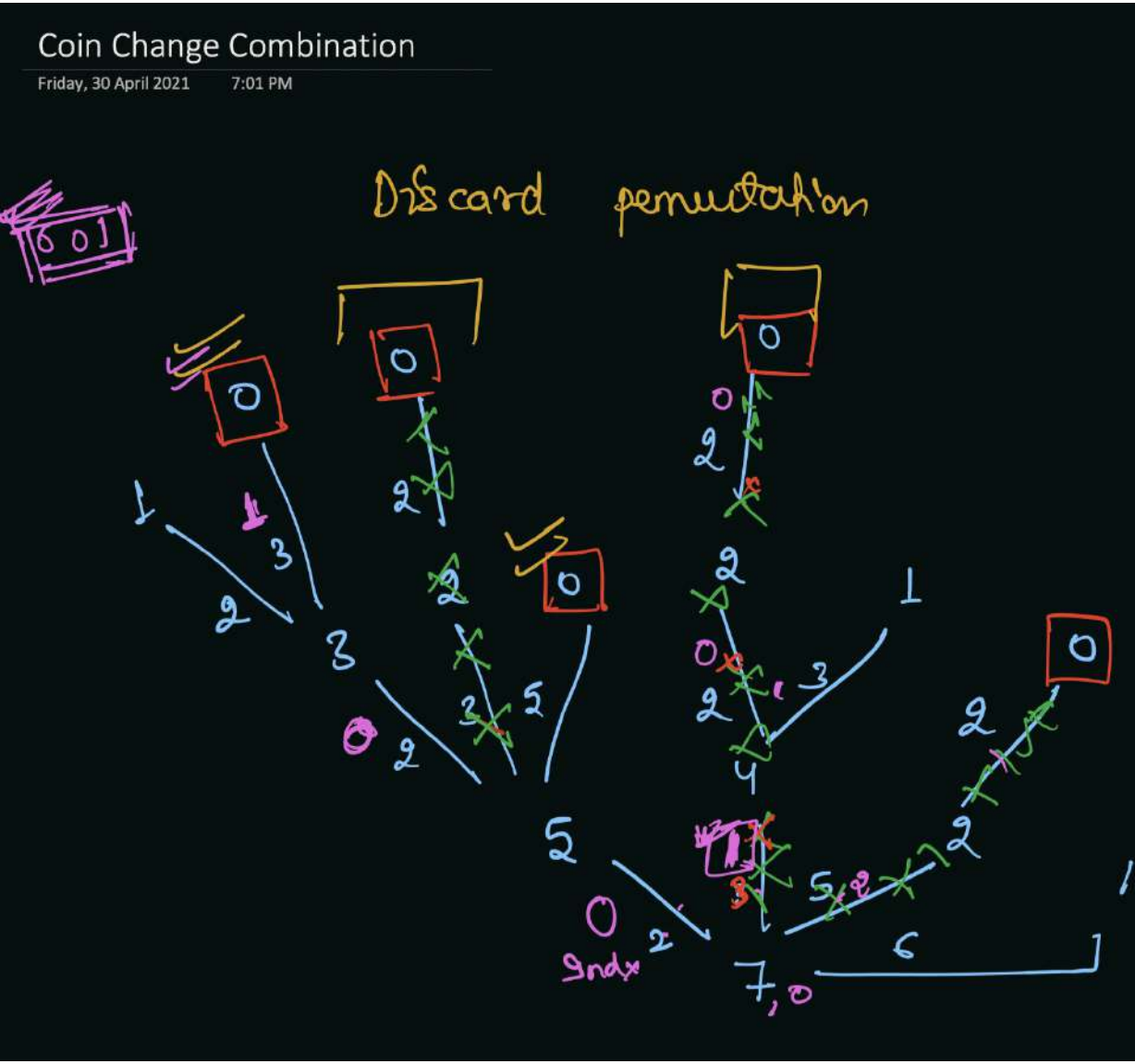
Storage = target +1

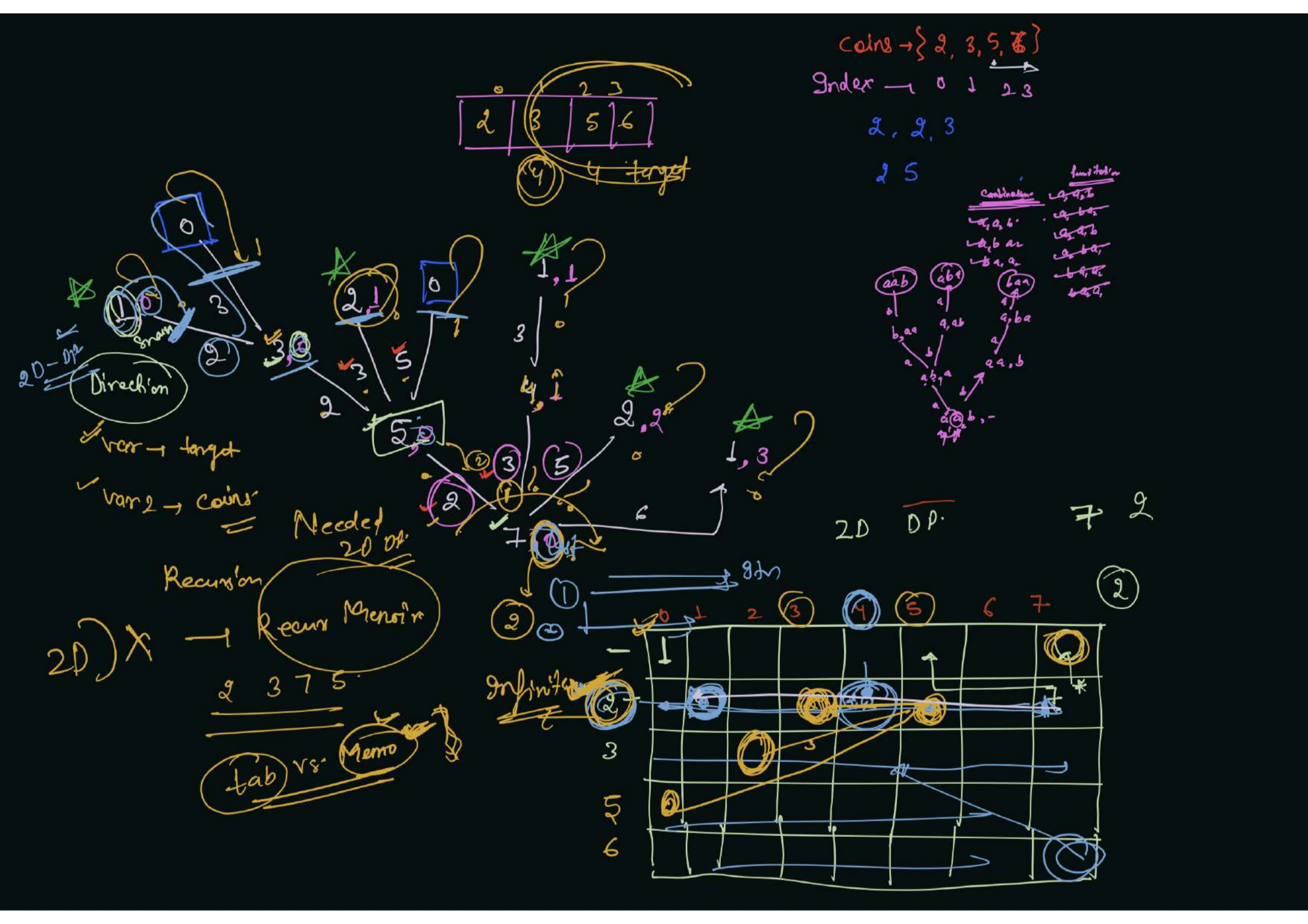
Meaning assign =

Direction analyse

Coin Change Combination

Friday, 30 April 2021 7:01 PM




coins → {2, 3, 5, 6}
Index → 0 1 2 3
2, 2, 3
2 5
2 3 5 6
4 target
2D - DP
Direction
var → target
var2 → coins
Needed 2D DP.
Recursion
Recur Memoir
2D DP.
7 2
infinite
tab Vs. Memo
2 3 7 5
combination
permutation

coins → 2 3 5 6

	0	1	2	3	4	5	6	7
coin 2 →	1	0	1	1	1	1	3	2
coin 3 →								
coin 5 →								
coin 6 →								

Steps - (1) Prepare DP of coin change permutation ccc and coin change combination.

print all possible permutation of coin change permutation.

(2) Reverse thinking

Target Sum Subset

Friday, 30 April 2021 6:45 PM