

518 Coin Change 2

You are given coins of different denominations and a total amount of money. Write a function to compute the number of combinations that make up that amount. You may assume that you have infinite number of each kind of coin.

Example 1:

```
Input: amount = 5, coins = [1, 2, 5]
Output: 4
Explanation: there are four ways to make up the amount:
5=5
5=2+2+1
5=2+1+1+1
5=1+1+1+1+1
```

Example 2:

```
Input: amount = 3, coins = [2]
Output: 0
Explanation: the amount of 3 cannot be made up just with coins of 2.
```

Example 3:

```
Input: amount = 10, coins = [10]
Output: 1
```

Intution and Solution::

I recommend this video:

<https://www.youtube.com/watch?v=DJ4a7cmjZY0>

Code :

```
int change(int amount, vector<int>& coins) {
    if(amount == 0) return 1;
    if(coins.size() == 0)
        return 0;
    int n = coins.size();
    int dp[n+1][amount+1];
    for(int i=0;i<=n;i++)
        dp[i][0] = 1;
    for(int i=1;i<=amount;i++)
        dp[0][i] = 0;
    for(int i=1;i<=n;i++){
        for(int j=1;j<=amount;j++){
            if(j >= coins[i-1])
            {
                dp[i][j] = dp[i-1][j] + dp[i][j-coins[i-1]];
            }
            else
                dp[i][j] = dp[i-1][j];
        }
    }
    return dp[n][amount];
}
```