322. Coin Change

Medium,
Dynamic Programming.

You are given coins of different denominations and a total amount of money amount. Write a function to compute the fewest number of coins that you need to make up that amount. If that amount of money cannot be made up by any combination of the coins, return -1.

Example 1:

```
Input: coins = [1, 2, 5], amount = 11
Output: 3
Explanation: 11 = 5 + 5 + 1
```

Example 2:

```
Input: coins = [2], amount = 3
Output: -1
```

解法一

Java

```
class Solution {
    public int coinChange(int[] coins, int amount) {
        int[] dp = new int[amount+1];
        for(int i=0; i<amount+1;i++){
            if(i==0) dp[0]=0;
            else dp[i]=amount+1;
        }
        for(int i=1; i<=amount; i++){
            for(int j=0; j<coins.length; j++){
                if(i>=coins[j]){
                    dp[i]=Math.min(dp[i],dp[i-coins[j]]+1);
                }
            }
        }
        if(dp[amount]>amount) return -1;
        return dp[amount];
    }
}
```