动态规划子数组： 446. 等差数列划分 II - 子序列

代码：

```cpp
class Solution {
public:
    int numberOfArithmeticSlices(vector<int>& nums) {
        int n = nums.size();
        int sum = 0;
        unordered_map<long long,vector<int>> hash;
        for(int i = 0;i < n;i++)
        {
            hash[nums[i]].push_back(i);
        }
        vector<vector<int>> dp(n,vector<int>(n));
        for(int j = 2;j < n;j++)
        {
            for(int i = 1; i < j;i++)
            {
                long long a = (long long)2*nums[i] - nums[j];
                if(hash.count(a))
                {
                    for(auto e : hash[a])
                    {
                        if(e < i)
                        {
                            dp[i][j] += dp[e][i] + 1;
                        }
                    }
                }
                sum += dp[i][j];
            }

        }
        return sum;
    }
};
```

动态规划回文序列：647. 回文子串

代码：

```cpp
class Solution {
public:
    int countSubstrings(string s) {
        int sum = 0;
        int n = s.size();
        vector<vector<int>> dp(n, vector<int>(n));
        for (int i = n-1; i >= 0; i--) {
            for (int j = i; j < n; j++) {
                if (s[i] == s[j]) {
                    if (i == j || i + 1 == j) {
                        dp[i][j] = 1;
                    } else {
                        dp[i][j] = dp[i + 1][j - 1];
                    }
                    if(dp[i][j]) sum++;
                }
            }
        }
        return sum ;
    }
};
```