

312. Burst Balloons

题目描述: <https://leetcode.com/problems/burst-balloons/>

给定一堆气球，气球上写着价格，当扎破一个气球时，获得 $\text{nums}[\text{left}] * \text{nums}[\text{k}] * \text{nums}[\text{right}]$ 这么多钱，然后k消失。

求把所有气球弄破最多能获得的钱数

解题思路:

$\text{dp}[i][j]$ 代表把从i到j扎破能获得的钱数。

记得nums两端临界点都放一个1进去。

$\text{dp}[i][j] = \max(\text{dp}[i][k-1] + \text{dp}[k+1][j] + \text{nums}[i-1] * \text{nums}[k] * \text{nums}[j+1]) \quad (i \leq k \leq j)$

代码:

```
class Solution {
public:
    int maxCoins(vector<int>& nums) {
        int n = nums.size();
        if(n == 0) return 0;
        nums.insert(nums.begin(), 1);
        nums.push_back(1);
        vector<vector<int>> > f(n+2, vector<int>(n+2, 0));
        for(int len = 1; len <= n; len++) {
            for(int i = 1; i+len-1 <= n; i++) {
                int j = i+len-1;
                for(int k = i; k <= j; k++) {
                    f[i][j] = max(f[i][j], f[i][k-1]+f[k+1][j]+nums[i-1]*nums[k]*nums[j+1]);
                }
            }
        }
        return f[1][n];
    }
};
```