

238. Product of Array Except Self

题目描述: <https://leetcode.com/problems/product-of-array-except-self/>

给定一个数组[1,2,3,4],要求返回一个对应长度的数组,每个位置的值都是前数组除了本位置的其他值的乘积。则返回[24,12,8,6] **注意: 不能使用除法**

解题思路:

遍历一遍数组,记录好每个位置对应的前面值得乘积,和后面数字的乘积,则返回两者相乘就好。

代码 1:

```
class Solution {
public:
    vector<int> productExceptSelf(vector<int>& nums) {
        vector<int> first(nums.size(), 1);
        vector<int> last(nums.size(), 1);

        for(int i = 1; i < nums.size(); i++){
            first[i] = first[i-1]*nums[i-1];
        }
        for(int i = nums.size() - 2; i >= 0; i--){
            last[i] = last[i+1]*nums[i+1];
        }
        for(int i = 0; i <  nums.size(); i++){
            nums[i] = first[i]*last[i];
        }
        return nums;
    }
};
```

代码 2:

```
class Solution {
public:
    vector<int> productExceptSelf(vector<int>& nums) {
        vector<int> first(nums.size(), 1);

        for(int i = 1; i < nums.size(); i++){
            first[i] = first[i-1]*nums[i-1];
        }
        int last = 1;
        for(int i = nums.size()-1; i >= 0; i--){
            int t = last *first[i];
            last *= nums[i];
            nums[i] = t;
        }
        return nums;
    }
};
```