# 238. Product of Array Except Self

Given an integer array nums, return an array answer such that answer[i] is equal to the product of all the elements of nums except nums[i].

The product of any prefix or suffix of nums is guaranteed to fit in a 32-bit integer.

You must write an algorithm that runs in O(n) time and without using the division operation.

## SOLUTION IN C++

class Solution {

public:

vector<int> productExceptSelf(vector<int>& nums) {

const int n = nums.size();

vector<int> ans(n); // Can also use `nums` as the ans array.

vector<int> prefix(n, 1); // prefix product

vector<int> suffix(n, 1); // suffix product

for (int i = 1; i < n; ++i)

prefix[i] = prefix[i - 1] \* nums[i - 1];

for (int i = n - 2; i >= 0; --i)

suffix[i] = suffix[i + 1] \* nums[i + 1];

for (int i = 0; i < n; ++i)

ans[i] = prefix[i] \* suffix[i];

return ans;

}

};