396. Rotate Function

Difficulty : Medium

https://leetcode.com/problems/rotate-function

You are given an integer array `nums` of length `n`.

Assume $\texttt{arr}_k$ to be an array obtained by rotating `nums` by `k` positions clock-wise. We define the **rotation function** `F` on `nums` as follow:

- $\texttt{F(k) = 0 * arr}_k\texttt{[0] + 1 * arr}_k\texttt{[1] + ... + (n - 1) * arr}_k\texttt{[n - 1]}$.

Return *the maximum value of* `F(0), F(1), ..., F(n-1)`.

The test cases are generated so that the answer fits in a **32-bit** integer.

Example 1:

```
Input: nums = [4,3,2,6]
Output: 26
Explanation:
F(0) = (0 * 4) + (1 * 3) + (2 * 2) + (3 * 6) = 0 + 3 + 4 + 18 = 25
F(1) = (0 * 6) + (1 * 4) + (2 * 3) + (3 * 2) = 0 + 4 + 6 + 6 = 16
F(2) = (0 * 2) + (1 * 6) + (2 * 4) + (3 * 3) = 0 + 6 + 8 + 9 = 23
F(3) = (0 * 3) + (1 * 2) + (2 * 6) + (3 * 4) = 0 + 2 + 12 + 12 = 26
So the maximum value of F(0), F(1), F(2), F(3) is F(3) = 26.
```

Example 2:

```
Input: nums = [100]
Output: 0
```

Constraints:

- `n == nums.length`
- $1 \le n \le 10^5$
- `-100 <= nums[i] <= 100`