3073. Maximum Increasing Triplet Value

Description

Given an array `nums`, return *the* ***maximum value*** *of a triplet* `(i, j, k)` *such that* `i < j < k` *and* `nums[i] < nums[j] < nums[k]`.

The **value** of a triplet `(i, j, k)` is `nums[i] - nums[j] + nums[k]`.

Example 1:

Input: `nums = [5,6,9]`

Output: `8`

Explanation: We only have one choice for an increasing triplet and that is choosing all three elements. The value of this triplet would be `5 - 6 + 9 = 8`.

Example 2:

Input: `nums = [1,5,3,6]`

Output: `4`

Explanation: There are only two increasing triplets:

`(0, 1, 3)`: The value of this triplet is `nums[0] - nums[1] + nums[3] = 1 - 5 + 6 = 2`.

`(0, 2, 3)`: The value of this triplet is `nums[0] - nums[2] + nums[3] = 1 - 3 + 6 = 4`.

Thus the answer would be `4`.

Constraints:

- `3 <= nums.length <= 10^5`
- `1 <= nums[i] <= 10^9`
- The input is generated such that at least one triplet meets the given condition.