You are given an integer array nums. You must perform **exactly one** operation where you can **replace** one element nums[i] with nums[i] \* nums[i].

Return *the****maximum****possible subarray sum after****exactly one****operation*. The subarray must be non-empty.

**Example 1:**

**Input:** nums = [2,-1,-4,-3]

**Output:** 17

**Explanation:** You can perform the operation on index 2 (0-indexed) to make nums = [2,-1,**16**,-3]. Now, the maximum subarray sum is 2 + -1 + 16 = 17.

**Example 2:**

**Input:** nums = [1,-1,1,1,-1,-1,1]

**Output:** 4

**Explanation:** You can perform the operation on index 1 (0-indexed) to make nums = [1,**1**,1,1,-1,-1,1]. Now, the maximum subarray sum is 1 + 1 + 1 + 1 = 4.

**Constraints:**

* 1 <= nums.length <= 105
* -104 <= nums[i] <= 104