A **logistics company** is managing a fleet of delivery vehicles. Each vehicle is tasked with transporting packages that are categorized into **three priority levels**:

- **0:** High-priority package (Must be delivered first)
- **1:** Medium-priority package (Should be delivered after high-priority packages but before low-priority)
- **2:** Low-priority package (Can be delivered last without urgency)

The company's delivery system must efficiently **arrange** the packages based on priority levels so that:

- All **high-priority packages (0)** are processed first.
- **Medium-priority packages (1)** come second.
- All **low-priority packages (2)** are processed last.

🚦 Input:

- An integer array `packages` where:
  - **`packages[i] ∈ {0, 1, 2}`.**
  - **`1 <= packages.length <= 10^5`.**

🎯 Output:

- The sorted list of packages, modified **in-place**.

Could you come up with a one-pass algorithm using only constant extra space?