

Full Stack Web Development

Exercise

Exercise

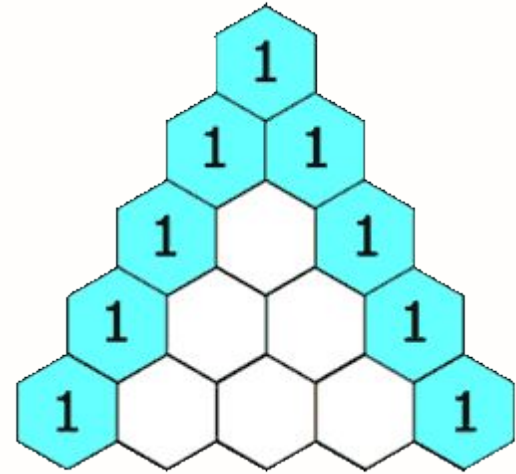
- Given an array `nums` of size `n`, return *the majority element*. The majority element is the element that appears more than $\lfloor n / 2 \rfloor$ times. You may assume that the majority element always exists in the array.
- **Example 1:**
 - **Input:** `nums = [3,2,3]`
 - **Output:** 3
- **Example 2:**
 - **Input:** `nums = [2,2,1,1,1,2,2]`
 - **Output:** 2

Exercise

- Create a function to convert roman numeral to integer.
- **Example 1:**
 - Input: s = "III"
 - Output: 3
 - Explanation: III = 3.
- **Example 2:**
 - Input: s = "LVIII"
 - Output: 58
 - Explanation: L = 50, V = 5, III = 3.
- **Example 3:**
 - Input: s = "MCMXCIV"
 - Output: 1994
 - Explanation: M = 1000, CM = 900, XC = 90 and IV = 4.

Exercise

- Given an integer numRows, return the first numRows of **Pascal's triangle**.
- In **Pascal's triangle**, each number is the sum of the two numbers directly above it as shown →
- **Example 1:**
 - **Input:** numRows = 5
 - **Output:** `[[1],[1,1],[1,2,1],[1,3,3,1],[1,4,6,4,1]]`
- **Example 2:**
 - **Input:** numRows = 1
 - **Output:** `[[1]]`



- You are given an array prices where prices[i] is the price of a given stock on the i^{th} day.
- You want to maximize your profit by choosing a **single day** to buy one stock and choosing a **different day in the future** to sell that stock.
- Return *the maximum profit you can achieve from this transaction*. If you cannot achieve any profit, return 0.
- **Example 1:**
 - Input: prices = [7,1,5,3,6,4]
 - Output: 5
 - Explanation: Buy on day 2 (price = 1) and sell on day 5 (price = 6), profit = 6-1 = 5.
 - Note that buying on day 2 and selling on day 1 is not allowed because you must buy before you sell.
- **Example 2:**
 - Input: prices = [7,6,4,3,1]
 - Output: 0
 - Explanation: In this case, no transactions are done and the max profit = 0.

Thank You!

