Requirements

- You can use c、c++、Java or Python
- For **questions 1 to 5**, you need to write in **functions form**, taking input parameters and returning the output, without having to use any input function like `scanf` to receive input from the keyboard. For example, you should write as following for question 4

```c
// c code
int romanToInt(char * s){
    // write you code here
}
```

```java
// java code
class Solution {
    public int romanToInt(String s) {
        // write you code here
    }
}
```

```python
# python3 code
class Solution:
    def romanToInt(self, s: str) -> int:
        # write you code here
```

- For **question 6** and the **Iris classification** in *OPTIONAL* , you only need to finish one of them. Please complete the Iris classification as a **priority**. If this is too hard for you, please complete question 6 then.

Compulsory

question 1

Given an array `nums` of size n, return the majority element.

The majority element is the element that appears more than `⌊n / 2⌋` times. You may assume that the majority element always exists in the array.

example

```
Input: nums = [2,2,1,1,1,2,2]
Output: 2
```

constraints

- n == nums.length
- 1 <= n <= 5*10^4
- -10^9<= nums[i] <= 10^9

follow-up(optional)

Could you solve the problem in linear time and in `O(1)` space?

question 2

A phrase is a palindrome if, after converting all uppercase letters into lowercase letters and removing all non-alphanumeric characters, it reads the same forward and backward. Alphanumeric characters include letters and numbers.

Given a string `s`, return true if it is a palindrome, or false otherwise.

example 1

```
Input: s = "A man, a plan, a canal: Panama"
Output: true
Explanation: "amanaplanacanalpanama" is a palindrome.
```

example 2

```
Input: s = "race a car"
Output: false
Explanation: "raceacar" is not a palindrome.
```

constraints

- 1 <= s.length <= 2 * 10^5
- `s` consists only of printable ASCII characters

question 3

You are climbing a staircase. It takes `n` steps to reach the top.

Each time you can either climb `1` or `2` steps. In how many distinct ways can you climb to the top?

example 1

```
Input: n = 2
Output: 2
Explanation: There are two ways to climb to the top.
1. 1 step + 1 step
2. 2 steps
```

example 2

```
Input: n = 3
Output: 3
Explanation: There are three ways to climb to the top.
1. 1 step + 1 step + 1 step
2. 1 step + 2 steps
3. 2 steps + 1 step
```

question 4

Roman numerals are represented by seven different symbols: `I, V, X, L, C, D` and `M`.

```
Symbol       Value
I             1
V             5
X             10
L             50
C             100
D             500
M             1000
```

For example, 2 is written as `II` in Roman numeral, just two ones added together. 12 is written as `XII`, which is simply `X` + `II`. The number 27 is written as `XXVII`, which is `XX` + `V` + `II`.

Roman numerals are usually written largest to smallest from left to right. However, the numeral for four is not `IIII`. Instead, the number four is written as `IV`. Because the one is before the five we subtract it making four. The same principle applies to the number nine, which is written as `IX`. There are six instances where subtraction is used:

- `I` can be placed before `V` (5) and `X` (10) to make 4 and 9.
- `X` can be placed before `L` (50) and `C` (100) to make 40 and 90.
- `C` can be placed before `D` (500) and `M` (1000) to make 400 and 900.

Given a roman numeral, convert it to an integer

example 1

```
Input: s = "LVIII"
Output: 58
Explanation: L = 50, V= 5, III = 3.
```

example 2

```
Input: s = "MCMXCIV"
Output: 1994
Explanation: M = 1000, CM = 900, XC = 90 and IV = 4.
```

Optional

You only need to finish one of followings. Please finish the Iris classification as a **priority**. If this is too hard for you, please complete question 6 then.

question 5

Given an integer array `nums`, find the subarray with the largest sum, and return its sum.

attention

please solve the problem in $O(n)$ time!

example 1

```
Input: nums = [-2,1,-3,4,-1,2,1,-5,4]
Output: 6
Explanation: The subarray [4,-1,2,1] has the largest sum 6.
```

example 2

```
Input: nums = [1]
Output: 1
Explanation: The subarray [1] has the largest sum 1.
```

constraints

- 1 <= nums.length <= 10^5
- -10^4 <= nums[i] <= 10^4

iris classification

Use a **linear regression** model to complete the iris dataset classification, you can choose to use `sklearn` or any other libraries or write a linear model from scratch.

Please divide the dataset into 3 parts, where **training set: validation set: testing set=7: 1: 2**