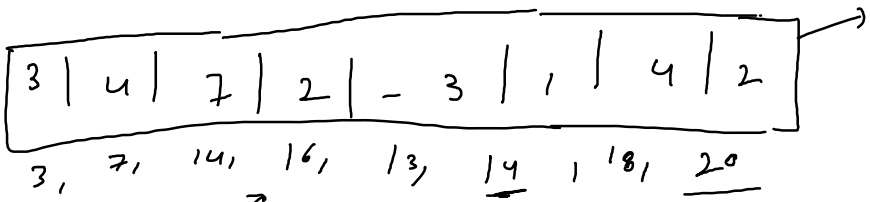
Day 05

Sunday, 3 January 2021 7:21 PM



Homework Assignments :

https://leetcode.com/problems/valid-palindrome/
https://leetcode.com/problems/valid-palindrome-ii/
https://leetcode.com/problems/string-to-integer-atoi/
https://leetcode.com/problems/reverse-string/
https://leetcode.com/problems/reverse-words-in-a-string/
https://leetcode.com/problems/expressive-words/

Given a string, determine if it is a palindrome, considering only alphanumeric characters and ignor
cases.
Note: For the purpose of this problem, we define empty string as valid palindrome.
Example 1:
Input: "A man, a plan, a canal: Panama"
Output: true

//Character.IsLetterOrDigit(s.charAt(i))

Implement atoi which converts a string to an integer.
The function first discards as many whitespace characters as necessary until the first non-whitesp character is found. Then, starting from this character takes an optional initial plus or minus sign fo as many numerical digits as possible, and interprets them as a numerical value.
The string can contain additional characters after those that form the integral number, which are ig and have no effect on the behavior of this function.
If the first sequence of non-whitespace characters in str is not a valid integral number, or if no suc sequence exists because either str is empty or it contains only whitespace characters, no convers performed.
If no valid conversion could be performed, a zero value is returned.
Note:

- Only the space character ' ' is considered a whitespace character.
- Assume we are dealing with an environment that could only store integers within the 32-bit sig integer range: [–231, 231 – 1]. If the numerical value is out of the range of representable value 1 or –231 is returned.

Input: str = "42"
Output: 42

Input: str = " -42"
Output : -42

Input: str = "4193 with words"
Output : 4193

Input: str = "words and 987"
Output : 0

```
// Discard Whitespace's  in the beginning

// Check if optional if it exists


// str.charAt(i) >='0'
// str.charAt(i) <='9'

Int result

While( I < =str.length && str.charAt(i) >='0' && str.charAt(i) <='9')

{
        If( result  > Integer.MAX_VALUE / 10 II result == integer.MAX_VALUE /10)
        {
            Return Integer.Max_Value
        }
        result = result  * 10 + ( str.charAt(i) - '0') ;
    }



}
```

Write a function that reverses a string. The input string is given as an array of characters char[].
Do not allocate extra space for another array, you must do this by **modifying the input array in-place** with O(1) extra memory.
You may assume all the characters consist of printable ascii characters.

Example 1:
Input: ["h","e","l","l","o"]
Output: ["o","l","l","e","h"]

https://leetcode.com/problems/reverse-words-in-a-string/

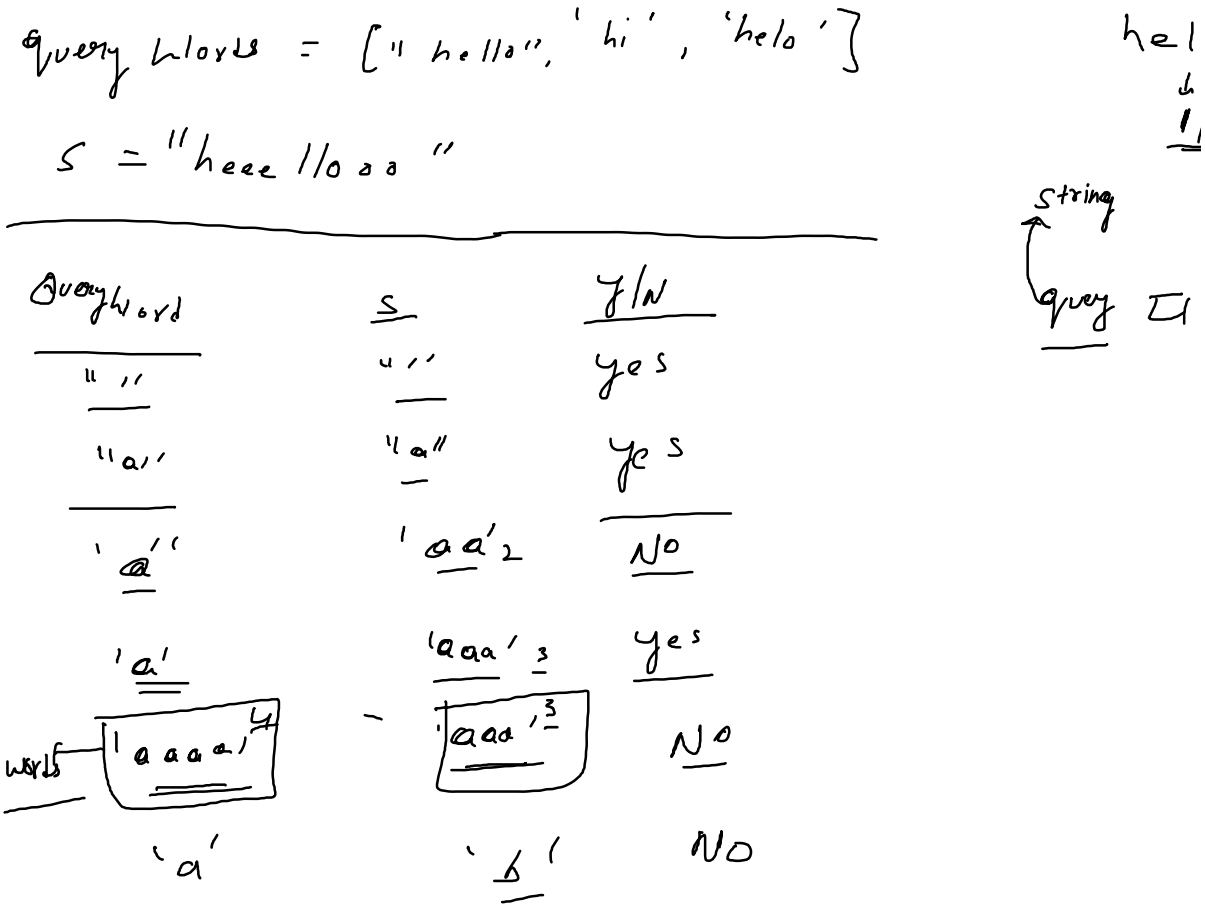
Given an input string s, reverse the order of the **words**.
A **word** is defined as a sequence of non-space characters. The **words** in s will be separated by at least one space.
Return *a string of the words in reverse order concatenated by a single space.*
Note that s may contain leading or trailing spaces or multiple spaces between two words. The returned string should only have a single space separating the words. Do extra spaces.

Input: s = "the sky is blue"
Output: "blue is sky the"

Sometimes people repeat letters to represent extra feeling, such as "hello" -> "heeellooo", "hi" -> "
these strings like "heeellooo", we have *groups* of adjacent letters that are all the same: "h", "eee", "ooo".

For some given string S, a query word is *stretchy* if it can be made to be equal to S by any number of applications of the following *extension* operation: choose a group consisting of characters c, and some number of characters c to the group so that the size of the group is 3 or more.

For example, starting with "hello", we could do an extension on the group "o" to get "hellooo", but cannot get "helloo" since the group "oo" has size less than 3. Also, we could do another extension > "llllll" to get "hellllllooo". If S = "hellllllooo", then the query word "hello" would be stretchy because two extension operations: query = "hello" -> "hellooo" -> "hellllllooo" = S.
Given a list of query words, return the number of words that are stretchy.



Example:
Input:
S = "heeellooo"
words = ["hello", "hi", "helo"]
Output: 1
Explanation:
We can extend "e" and "o" in the word "hello" to get "heeellooo".
We can't extend "helo" to get "heeellooo" because the group "ll" is not size 3 or more.

```
↳ function isStretchy(s, w) {
    var i = 0; j = 0;
    while (i < s.length && w.length)
    {
        char c1 = s[i];
        char c2 = w[j];
        if(c1 != c2) {
            return false;
        }
        int e1 = i, e2 = j;
        while (e1 < s.length && s[e1] == c1)
            e1++;
        while (e2 < w.length && w[e2] == c2)
            e2++;
        int n1 = e1 - i;
        int n2 = e2 - j;
        if (n1 < n2 || (n1 < 3 && n1 != n
            return false;
        i = e1;
        j = e2;
    }
    if (i == s.length && j == w.length)
        return true;
    else
        return false;
}
words.forEach(
    count =
```

BW ES

heello

heelloo

aeqa aaa

hello hello helo

heeello

hello