Reverse Each Word

Problem Description: Given a string S, reverse each word of a string individually. For eg. if a string is "abc def", reversed string should be "cba fed".

Sample Input:

```
Welcome to Coding Ninjas
```

Sample Output:

```
emocleW ot gnidoC sajniN
```

How to approach?

The first thing to notice in this problem is that a “word” is defined as a continuous sequence of non-space characters. Also, there are no leading or trailing spaces in the string and two words are separated by only a single space.

Now that we know this, we know that every ‘space’ denotes the end of a word and the beginning of the next word.

So, we could iterate through the input string and look out for spaces. If we encounter a space we could reverse the string from the previous space encountered to the space we just encountered. This would give us an O(n) approach to solving this problem.

Let’s have a more detailed discussion about this approach. We’ll maintain a `prev` variable that will be initialised with -1 (not 0, we’ll see why). Then we’ll iterate through the string. If we encounter a non-space character, we just move on to the next iteration. However. if we encounter a space, then we’ll do the following steps:

1. Let the index where we encountered the space be `i`. Then we will reverse the string between `prev+1` (the first character after the previous space) to `i-1` (the first character just before the current space) and update the value of `prev` to `i`.

Now, can you see why we initialized the value of `prev` to -1. It's because `prev` is storing the index of spaces only, not other characters. And since index 0 of the string will always have a non-space character, we can look at -1 as an 'index' containing a space.

2. Now the next time we encounter a space, we'll repeat step 1. And we'll keep iterating till we reach the end of the string.

NOTE: Now you may be thinking that we're done with the question. But wait, there's more. There's a special case that we have not dealt with. Right now, our algorithm runs on the assumption that every word will have a space in front of it. But, that is not the case for the last word in the string. That word won't have any space after it. So, if we end our program now, the last word won't be reversed! This has an easy fix though. After ending the loop we can call our reverse function from indices `prev+1` to `n-1`, where n is the length of string.

The pseudo-code for this approach is shown on the next page.

```
//Function to reverse each word
function reverseEachWord(str):

  n <- str.length
  prev <- -1
  i <- 0

  while(i < n):

    if(str[i] is a space):
      reverseString(str,prev+1,i-1)
      prev <- i

    i <- i + 1

  reverseString(str, prev+1, n-1)

  //Print or return the string as per requirement



//Function to reverse a string between two indices
function reverseString(str, start, end):
  mid <- (start + end)/2
  i <- start
  while(i < mid):
    swap(str[i],str[end])
    i <- i + 1
    end <- end - 1

  return;
```