Valid Palindrome

Given a string, determine if it is a palindrome, considering only alphanumeric characters and ignoring cases.

For example,
`"A man, a plan, a canal: Panama"` is a palindrome.
`"race a car"` is *not* a palindrome.

Note:
Have you consider that the string might be empty? This is a good question to ask during an interview.

For the purpose of this problem, we define empty string as valid palindrome.

Solution 1

```
bool isPalindrome(string s) {
    for (int i = 0, j = s.size() - 1; i < j; i++, j--) { // Move 2 pointers from
each end until they collide
        while (isalnum(s[i]) == false && i < j) i++; // Increment left pointer if
not alphanumeric
        while (isalnum(s[j]) == false && i < j) j--; // Decrement right pointer i
f no alphanumeric
        if (toupper(s[i]) != toupper(s[j])) return false; // Exit and return erro
r if not match
    }

    return true;
}
```

written by satyakam original link here

Solution 2

```
public class Solution {
    public boolean isPalindrome(String s) {
        if (s.isEmpty()) {
            return true;
        }
        int head = 0, tail = s.length() - 1;
        char cHead, cTail;
        while(head <= tail) {
            cHead = s.charAt(head);
            cTail = s.charAt(tail);
            if (!Character.isLetterOrDigit(cHead)) {
                head++;
            } else if(!Character.isLetterOrDigit(cTail)) {
                tail--;
            } else {
                if (Character.toLowerCase(cHead) != Character.toLowerCase(cTail))
{
                    return false;
                }
                head++;
                tail--;
            }
        }

        return true;
    }
}
```

written by aleksey.danilevsky.5 original link here

Solution 3

```
bool isPalindrome(string s) {
    int start=0, end=s.length()-1;
    while(start<end) {
        if (!isalnum(s[start])) start++;
        else if (!isalnum(s[end])) end--;
        else {
            if (tolower(s[start++])!=tolower(s[end--])) return false;
        }
    }
    return true;
}
```

written by lester_zhang original link here