```
A man, a plan, a canal: Panama
```

A man a plan a canal Panama → removed non alphanumeric chars

a man a plan a Canal panama → converted to lowercase

↓

amanaplanacanalpanama

Yes.

→ Character.isLetterOrDigit();

```java
public static boolean isPalindrome(String str){

    int left = 0;
    int right = str.length() - 1;

    while(left < right){
        while(left < right && !Character.isLetterOrDigit(str.charAt(left))){
            left++;
        }
        while(left < right && !Character.isLetterOrDigit(str.charAt(right))){
            right--;
        }

        if(Character.toLowerCase(str.charAt(left)) != Character.toLowerCase(str.charAt(right))){
            return false;
        }

        left++;
        right--;
    }
    return true;
}
```

```java
public static int countNumbers(String str){
    int count = 0;
    boolean isNumber = false;

    for(int i = 0; i < str.length(); i++){
        char ch = str.charAt(i);
        if(Character.isDigit(ch)){
            if(!isNumber){
                count++;
                isNumber = true;
            }
        }
        else{
            isNumber = false;
        }
    }
    return count;
}
```

→ 132 ab 2 cd 45 ef 6

0 1 2 3 4 5 6 7 8 9 10 11 12

Count = 0 1 2 3 4

isN = f

i = 0, '1'

isDigit(1) → T

if(!isN) → T

isN = t

132 a

i = 1, '3'

isDigit(3) T

i = 2, '2'

isDigit 2)

if(!—)

isDigit('a') → F

isN = f

TC → O(n)

SC → O(1)

```java
public static char nonrepeating(String str){

    int freq[]= new int[26];

    for(){
        freq[ch - 'a']++;
    }

    for(){  // freq array traverse
        if(freq[ch-'a'] == 1){
            return ch;
        }
    }
    retun '-';
}
```

TC → O(n)

SC → O(n) → fixed of size 26.