# В некоторых задачах могут потребоваться простые регулярки. Можете их не делать или загуглите. Или поучите, кто всё освоил.

# 0.Палиндром

Палиндром — слово, предложение или последовательность символов, которая абсолютно одинаково читается как в привычном направлении, так и в обратном. К примеру, “Anna” — это палиндром, а “table” и “John” — нет.

## Постановка

Дана строка; нужно написать функцию, которая позволяет вернуть значение true, если строка является палиндромом, и false — если нет. При этом нужно учитывать пробелы и знаки препинания.

# **1.Анаграмма**

Так называют слово, которое содержит все буквы другого слова в том же количестве, но ином порядке.

## **Постановка**

Нужно написать функцию, которая проверяет, являются ли две строки анаграммами, причем регистр букв не имеет значения. Учитываются лишь символы; пробелы или знаки препинания в расчет не берутся.

# **2.Поиск гласных**

Достаточно простая задача, которая часто попадается на собеседованиях.

## **Постановка**

Нужно написать функцию, принимающую строку в качестве аргумента и возвращающую количество гласных, которые содержатся в строке.  
Гласными являются «a», «e», «i», «o», «u».

# **3.Фибоначчи**

Классическая задача, которую можно встретить на собеседованиях самого разного уровня. Стоит напомнить, что последовательность Фибоначчи — это ряд чисел, где каждое последующее является суммой двух предыдущих. Так, первые десять чисел выглядят следующим образом: 0, 1, 1, 2, 3, 5, 8, 13, 21, 34.

## **Постановка**

Нужно написать функцию, которая возвращает n-ную запись в определенной последовательности, причем n — число, которое передается в качестве аргумента функции.

**Задача.** Дана строка **'aaa bbb ccc'**. Вырежите из нее слово **'bbb'** тремя разными способами (через [substr](http://old.code.mu/javascript/string/substr.html), [substring](http://old.code.mu/javascript/string/substring.html), [slice](http://old.code.mu/javascript/string/slice.html)).

**Задача.** В переменной **date** лежит дата в формате **'2025-12-31'**. Преобразуйте эту дату в формат **'31/12/2025'**.

Дана строка **'js'**. Сделайте из нее строку **'JS'**.

 Дана строка **'JS'**. Сделайте из нее строку **'js'**.

Дана строка **'я учу javascript!'**. Найдите **количество символов** в этой строке.

 Дана строка **'я учу javascript!'**. Вырежите из нее слово **'учу'** и слово **'javascript'** тремя разными способами (через [substr](http://old.code.mu/javascript/string/substr.html), [substring](http://old.code.mu/javascript/string/substring.html), [slice](http://old.code.mu/javascript/string/slice.html)).

 Дана строка **'я учу javascript!'**. Найдите позицию подстроки **'учу'**.

 Дана переменная **str**, в которой хранится какой-либо текст. Реализуйте обрезание длинного текста по следующему принципу: если количество символов этого текста больше заданного в переменной n, то в переменную **result** запишем первые **n** символов строки **str** и добавим в конец троеточие '...'. В противном случае в переменную **result** запишем содержимое переменной **str**.

Дана строка **'я учу javascript!'**. С помощью метода **split** запишите каждое слово этой строки в **отдельный** элемент массива.

 Дана строка **'я учу javascript!'**. С помощью метода **split** запишите каждый символ этой строки в отдельный элемент массива.

 В переменной **date** лежит дата в формате **'2025-12-31'**. Преобразуйте эту дату в формат **'31.12.2025'**.

Дан массив **['я', 'учу', 'javascript', '!']**. С помощью метода **join** преобразуйте массив в строку **'я+учу+javascript+!'**.

Преобразуйте **первую букву** строки в верхний регистр.

 Преобразуйте первую букву **каждого слова** строки в верхний регистр.

 Преобразуйте строку **'var\_test\_text'** в **'varTestText'**. Скрипт, конечно же, должен работать с любыми аналогичными строками.

**Напишите функцию stringToarray(str), которая преобразует строку в массив слов.**

**Напишите функцию delete\_characters(str, length), которая возвращает подстроку, состоящую из указанного количества символов.**

**Напишите функцию insert\_dash(str), которая принимает строку str в качестве аргумента и вставляет тире (-) между словами. При этом все символы строки необходимо перевести в верхний регистр.**

**Напишите функцию, которая принимает строку в качестве аргумента и преобразует регистр первого символа строки из нижнего регистра в верхний**

**Напишите функцию capitalize(str), которая возвращает строку, в которой каждое слово начинается с заглавной буквы.**

**Напишите функцию change\_register(str), которая принимает в качестве аргумента строку и и заменяет регистр каждого символа на противоположный. Например, если вводится «КаЖдЫй ОхОтНиК», то на выходе должно быть «кАжДыЙ оХоТнИк».**

**Напишите функцию remove\_char(str), которая возвращает строку, очищенную от всех не буквенно-цифровых символов.**

**Напишите функцию zeros(num, len), которая дополняет нулями до указаной длины числовое значение с дополнительным знаком «+» или « -« в зависимости от передаваемого аргумента.**

**Напишите функцию insensitive\_search(str1, str2), которая осуществляет поиск подстроки str2 в строкеstr1 без учёта регистра символов.**

**Напишите функцию initSnake(str), которая преобразует стиль написания составных слов строки из CamelCase в snake\_case, при котором несколько слов разделяются символом подчеркивания (\_), причём каждое слово пишется с маленькой буквы.**

**Напишите функцию repeatStr(str, n), которая вовращает строку повторяемую определённое количество раз.**

**Напишите функцию path(pathname), которая вовращает имя файла (подстрока после последнего символа "\" ) из полного пути к файлу.**

var pathname = "/home/user/dir/file.txt";

**Напишите функцию insert(str, substr, pos), которая вставляет подстроку substr** **в указханную позицию pos** **строки str. По умолчанию подстрока вставляется в начало строки.**

**Напишите функцию limitStr(str, n, symb), которая обрезает строку, если она длиннее указанного количества символов n. Усеченная строка должна заканчиваться троеточием «...» (если не задан параметр symb) или заданным символом symb.**

**Напишите функцию cutString(str, n), которая делит строку на подстроки, состоящие из n символов.**

**Напишите функцию count(str, stringsearch), которая возвращает количество символов stringsearch в строке str.**

**Напишите функцию cutTegs(str), которая возвращает строку str, очищенную от всех HTML-тегов.**

**Напишите функцию strip(str), которая удаляет все лишние пробелы из строки str.**

**let** str = " Max is a good boy ";

**Напишите функцию cutString(str, n), которая удаляет лишние слова из строки str, оставив в ней n слов.**

**Напишите функцию unique\_letters(str), которая возвращает строку, оставив в ней только уникальные символы, т.е. встречающиеся в строке один раз.**

**Напишите функцию removeDuplicate(str), которая возвращает строку, очищенную от слов-дупликатов, т.е. каждое слово должно повторяться не более одного раза.**

**Напишите функцию findWord(word, str), которая проверяет, существует ли в строке str слова word.**

**Напишите функцию isUpperCase(str, character), которая определяет в каком регистре написан символ строки в указанной позиции. Если в верхнем регистре выводится – true, если в нижнем – false.**

**Напишите функцию, которая принимает на вход строку, а возвращает эту строку «задом наперед»**