Задача №1. Написать функцию «encode», которая преобразует указанную входную последовательность в указанную выходную последовательность. Пример входных данных: AABBBGGHKKK, функция «encode» возвращает: 2A3B2GH3K.

Задача №2. Разместите в переменную какой-либо длинный текст (несколько предложений). И определите средствами Javascript есть ли в этом тексте какое-либо слово.

Задача №3. У нас есть строка "Привет, мир". Сделайте каждое слово с заглавной буквы, сделайте все буквы заглавными.

Задача №4. Посчитайте, сколько букв в строке "Привет, мир".

Задача №5.

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Задача №6. Есть строка "Нужно:убрать:двоеточие". Замените в этой строке символ ":" на символ пробела.

Задача №7.

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

Задача №1. Написать функцию «encode», которая преобразует указанную входную последовательность в указанную выходную последовательность. Пример входных данных: AABBBGGHKKK, функция «encode» возвращает: 2A3B2GH3K.

Решение.

Изображение выглядит как текст, снимок экрана, Шрифт, документ

Автоматически созданное описание

Сложность алгоритма: **O(N)**

Вывод:

1. строка уже по умолчанию массив, по которому можно итерироваться циклом for of;
2. у строк есть метод «charAt(index)», который позволяет получить символ по индексу из строки.

Задача №2. Разместите в переменную какой-либо длинный текст (несколько предложений). И определите средствами Javascript есть ли в этом тексте какое-либо слово.

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Сложность алгоритма: O(1) или O(N).

Вывод:

1. для поиска по большим строкам можно воспользоваться String.prototype.search([regexp]), который на вход ожидает регулярное выражение. **Возвращает** индекс вхождения, иначе -1;
2. динамическое регулярное выражение можно создать через new RegExp(‘something’, [flags]);
3. статическое регулярное выражение можно записать через «/something/[flags]».

Задача №3. У нас есть строка "Привет, мир". Сделайте каждое слово с заглавной буквы, сделайте все буквы заглавными.

Изображение выглядит как текст, электроника, снимок экрана, программное обеспечение

Автоматически созданное описание Изображение выглядит как текст, Шрифт, программное обеспечение, снимок экрана

Автоматически созданное описание

Сложность алгоритма: **O(N)**

Вывод:

• длину строки можно получить использую String.prototype.length;

• строку можно разбить на массив, используя String.prototype.**split**(separator);

• удалить лишние пробелы в начале / конце строки – String.prototype.**trimLeft**() / String.prototype.**trimRight**();

• получить фрагмент строки – String.prototype.**slice**(from, to); **!примечание**: [from, to);

• Изменить регистр строки – String.prototype.**toLowerCase()** / String.prototype.**toUpperCase**().

Задача №4. Посчитайте, сколько букв в строке "Привет, мир".

Изображение выглядит как текст, снимок экрана, дисплей, программное обеспечение

Автоматически созданное описание

Сложность алгоритма: **O(N)**

Задача №5.

Изображение выглядит как текст, снимок экрана, Шрифт, число

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, Шрифт, программное обеспечение

Автоматически созданное описание

Задача №6. Есть строка "Нужно:убрать:двоеточие". Замените в этой строке символ ":" на символ пробела.

Изображение выглядит как текст, Шрифт, программное обеспечение, снимок экрана

Автоматически созданное описание

Задача №7.

Изображение выглядит как текст, снимок экрана, Шрифт, линия

Автоматически созданное описание

Изображение выглядит как текст, снимок экрана, Шрифт, программное обеспечение

Автоматически созданное описание