

Университет ИТМО, факультет программной инженерии и компьютерной техники  
Двухнедельная отчётная работа по «Информатике»: аннотация к статье

Дата прошедшей лекции	Номер прошедшей лекции	Название статьи/главы книги/видеолекции	Дата публикации (не старше 2021 года)	Размер статьи (от 400 слов)	Дата сдачи
11.09.2024	1	TREX: 27-ричная симметричная система счисления	11.06.2021	~997	25.09.2024
25.09.2024	2	Новый алгоритм сжатия LZ4 от TOAST в PostgreSQL 14. Насколько быстрым он может быть?	26.11.2021	~1338	09.10.2024
09.10.2024	3	Regexes Got Good: The History And Future Of Regular Expressions In JavaScript	20.08.2024	~3020	23.10.2024
	4				
	5				
	6				
	7				

Выполнил(а) Соколов А.А., № группы P3108, оценка                       
Фамилия И.О. студента не заполнять

**Прямая полная ссылка на источник или сокращённая ссылка (bit.ly, tr.im и т.п.)**

<https://www.smashingmagazine.com/2024/08/history-future-regular-expressions-javascript/>

**Теги, ключевые слова или словосочетания (минимум три слова)**

Современные регулярные выражения в JavaScript, История регулярных выражений в JavaScript, Улучшения regex в JavaScript

**Перечень фактов, упомянутых в статье (минимум четыре пункта)**

1. Каждая новая версия JavaScript добавляла небольшие улучшения в регулярные выражения
2. ES2018 версия включает в себя поддержку флага s (dotAll), а ES2020 – добавила метод “matchAll”
3. Несмотря на улучшения, JavaScript по-прежнему не поддерживает некоторые важные функции других языков, такие как флаг x для незначительных пробелов и комментариев.
4. История регулярных выражений в JavaScript начинается с ECMAScript 3, который добавил их в JavaScript, вдохновленные Perl
5. ES2024 добавил флаг v, который поддерживается в Node.js 20 и который расширяет возможности работы с Unicode-свойствами, позволяя работать с множествами символов и строк.
6. Named capturing groups используются в регулярных выражениях для более удобного извлечения подстрок из совпадений.

**Позитивные следствия и/или достоинства описанной в статье технологии (минимум три пункта)**

1. Регулярные выражения в JavaScript стали мощным инструментом для обработки текста благодаря новым флагам и улучшенной поддержке Unicode.
2. Введение флага v значительно расширило возможности работы с символами, добавив поддержку пересечения и вычитания множеств.
3. Атомарные группы и жадные квантификаторы помогают избежать проблем, связанных с бэктрекингом (важно для обеспечения стабильной работы приложений и защиты от атак типа ReDoS)

**Негативные следствия и/или недостатки описанной в статье технологии (минимум три пункта)**

1. Отсутствие флага x затрудняет написание читаемых и комментируемых регулярных выражений в JavaScript.
2. Несмотря на улучшения, JavaScript все еще использует движок с обратным отслеживанием, что может приводить к уязвимостям ReDoS (Denial of Service на основе регулярных выражений).
3. Встроенные регулярные выражения в JavaScript остаются сложными для понимания и поддержки при работе с более сложными паттернами, несмотря на улучшения.

**Ваши замечания, пожелания преподавателю или анекдот о программистах<sup>1</sup>**

How to Regex:

Step 1: Open your favorite editor, Step 2: Let your cat play on your keyboard

<sup>1</sup> Наличие этой графы не влияет на оценку