Университет ИТМО, факультет программной инженерии и компьютерной техники

Двухнедельная отчётная работа по «Информатике»: аннотация к статье

Дата прошедшей лекции: 11.10.2023 Номер прошедшей лекции: 3 Дата сдачи: 25.10.2023

Выполнил(а) Гаврилин О. С. , № группы *P3130* , оценка

Фамилия И.О. студента не заполнять

|  |  |  |
| --- | --- | --- |
| **Название статьи/главы книги/видеолекции**  Регулярные выражения. Всё проще, чем кажется | | |
| **ФИО автора статьи (или e-mail)**  [NikitaPozdeyev](https://habr.com/ru/users/NikitaPozdeyev/) | **Дата публикации**  **(не старше 2020 года)**  "11" июля 2021г. | **Размер статьи**  **(от 400 слов)**  \_\_\_\_\_\_\_\_ |
| **Прямая полная ссылка на источник или сокращённая ссылка (bit.ly, tr.im и т.п.)**  https://habr.com/ru/articles/567106/ | | |
| **Теги, ключевые слова или словосочетания**  Регулярки, регулярные выражения | | |
| **Перечень фактов, упомянутых в статье (минимум три пункта)**  **Что такое регулярка и с чем ее едят?**  Если по простому, регулярка- это некий шаблон, по которому фильтруется текст. Мы можем написать нужный нам шаблон (регулярку) и таким образом искать в тексте необходимые нам символы, слова и т.д. Также их используют, например, при заполнении поля E-mail на различных сайтах, т.е. создают шаблон по типу: someEmail@gmail.com. Это я взял как пример, не более. Теперь, разобравшись, что это, приступим к изучению. Обещаю, скучно не будет)  **Квантификаторы**  С помощью квантификаторов мы можем указывать сколько раз должен повторяться тот или иной символ (ну или группа символов). Ниже приведу список квантификаторов с пояснением, а дальше попрактикуемся с ними.  **{n}** - символ повторяется ровно n раз  **{m,n}** - символ повторяется в диапазоне от m до n раз  **{m,}** - символ повторяется минимум m раз (от m и более)  **Lookahead и lookbehind (опережающая и ретроспективная проверки)**  **Lookahead** или же **опережающая проверка** позволяет выбрать символ или группу символов, если после него идет идет какой-либо символ или группа символов. **Lookbehind** или же **ретроспективная проверка** позволяет выбрать символ или группу символов, если до них идет какой-то символ или группа символов.  **lookahead** - опережающая проверка - **X(?=Y)** - найти Х, при условии, что после него идет Y  **негативная опрережающая проверка** - **Х(?!Y)**  **lookbehind** - ретроспективная проверка - **(?<=Y)X** - найти Х, при условии, что до него идет Y  **негативная ретроспективная проверка** - **(?<!Y)Xo** | | |
| **Позитивные следствия и/или достоинства описанной в статье технологии (минимум три пункта)**   1. Обеспечивают быструю обработку и фильтрацию текста 2. Упрощают работу со строками 3. Поддерживаются в большинстве ЯП | | |
| **Негативные следствия и/или недостатки описанной в статье технологии (минимум три пункта)**   1. Являются сложными к прочтению 2. Наличие некоторого числа вариаций часто вызывает путаницу 3. В некоторых случаях могут замедлять работу программы | | |
| **Анекдот:**  - Что у программистов быстрее всего изнашивается?  - Отпечатки пальцев! | | |