

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

Дата прошёлшей лекции	Номер прошёлшей лекции	Название статьи/главы книги/видеолекции	Дата публикации (не старше 2022 года)	Размер статьи (от 400 слов)	Дата сдачи
10.09.2025	1	Представление числовой информации.	05.09.2022	~6200	24.09.2025
24.09.2025	2	ОБРАТИМЫЙ АЛГОРИТМ СОКРЫТИЯ ДАННЫХ В ЗАШИФРОВАННЫХ ИЗОБРАЖЕНИЯХ, ИСПОЛЬЗУЮЩИЙ КОД ХЭММИНГА (7, 4) И MSB-ПРОГНОЗИРОВАНИЕ	2022	~4500	08.10.2025
08.10.2025	3	Стратегии работы с регулярными выражениями в Python	2025	~4200	22.10.2025
	4				
	5				
	6				
	7				

Выполнил(а) _____ Шелестюк М.О. _____, № группы Р3117, оценка _____
Фамилия И.О. студента _____ не заполнять

Прямая полная ссылка на источник или сокращённая ссылка (bit.ly, tr.im и т.п.)
<https://innova-science.ru/wp-content/uploads/2025/07/sbornik-nauchnyh-trudov-12.07.2025-noi-8.pdf#page=6>

Теги, ключевые слова или словосочетания

Регулярные выражения, формат данных

Перечень фактов, упомянутых в статье (минимум четыре пункта)

- Классы символов, такие как \d, \w, \s используются для обозначения категорий знаков — цифр, буквенно-цифровых символов и пробелов.
- Квантификаторы {n}, {n, }, {n, m} управляют количеством совпадений элемента или группы в выражении.
- Для тестирования и отладки регулярных выражений применяются специализированные онлайн-инструменты, например Regex101, Regexr, RegExPal и RegexPlanet.
- Несмотря на универсальность синтаксиса, различные языки имеют собственные способы задания флагов и вызова функций.

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

- Использование регулярных выражений в Python значительно ускоряет обработку текстовых данных.
- Поддержка группировок и квантификаторов позволяет извлекать конкретные подстроки и выполнять контекстные проверки внутри выражений.
- Наличие интерактивных инструментов тестирования (Regex101, Regexr) облегчает обучение и снижает вероятность синтаксических ошибок.

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

- Сложный синтаксис регулярных выражений делает их трудночитаемыми и малопонятными.
- Регулярные выражения имеют высокую вычислительную стоимость при анализе длинных строк с большим количеством возможных совпадений.
- Различия в синтаксисе затрудняют переносимость решений между языками программирования.

Ваши замечания, пожелания преподавателю или анекдот о программистах¹

```
text = "Старшекурсники говорят: прохождение сессии жуть"  
pattern = r"(\w)\w*\s+(\w)\w*\s+(\w)\w*\.\?$_"  
result = re.search(pattern, text)  
print(result.group(1) + result.group(2) + result.group(3))  
#псж :)
```

¹ Наличие этой графы не влияет на оценку

