МИНИСТЕРСТВО ЦИФРОВОГО РАЗВИТИЯ, СВЯЗИ И МАССОВЫХ

КОММУНИКАЦИЙ РОССИЙСКОЙ ФЕДЕРАЦИИ

Ордена Трудового Красного Знамени федеральное государственное бюджетное образовательное учреждение высшего образования

**«Московский технический университет связи и информатики»**

Кафедра «Математическая кибернетика и информационные технологии»

Лабораторная работа №5

**по дисциплине**

**«Информационные технологии и программирование»**

Выполнил: студент гр. БВТ2201

Аблязов И.В.

Проверил:

Москва, 2023 г.

# ОСНОВНАЯ ЧАСТЬ

## Задание 1

В первом задании необходимо написать программу, которая будет искать все числа в заданном тексте и выводить их на экран. При этом программа должна использовать регулярные выражения для поиска чисел и обрабатывать возможные ошибки.

В данной программе нам необходимо искать подстроки, которые могут начинаться с символа «-», должны состоять только из чисел и где числа могут быть разделены точкой. Код данной программы показан на рисунке 1.

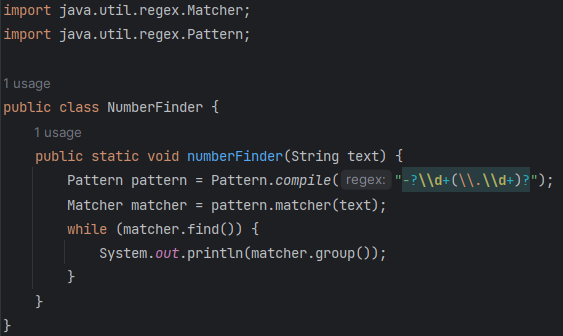


Рисунок 1 Первое задание

Проведем тестирование программы, как показано на рисунке 2.

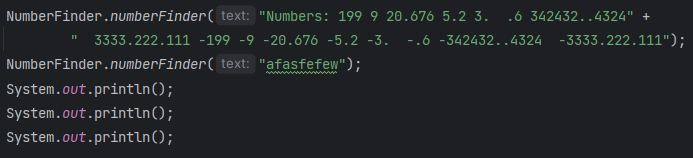


Рисунок 2 Тест 1 задания

Результат выполнения программы показан на рисунке 3.

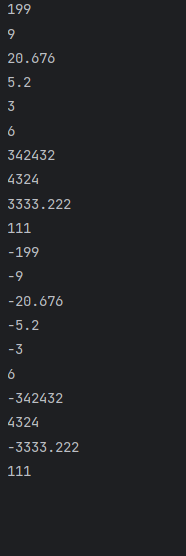


Рисунок 3 Результат первой программы

## Задание 2

Во втором задании необходимо написать программу, которая будет проверять корректность ввода пароля. Пароль должен состоять из латинских букв и цифр, быть длинной от 8 до 16 символов и содержать хотя бы одну заглавную букву и цифру. При этом программа должна использовать регулярные выражения для проверки пароля и обрабатывать возможные ошибки.

Для начала укажем начало и конец строки символами ^ и $. Далее мы указываем с помощью «заглядывания вперед», что у нас должно присутствовать хотя бы 1 цифра. Тоже самое проделываем и для заглавных латинских букв. В конце указываем, что наш пароль должен состоять из любых букв латинского алфавита и цифр в количестве от 8 до 16. Код данной программы показан на рисунке 4.

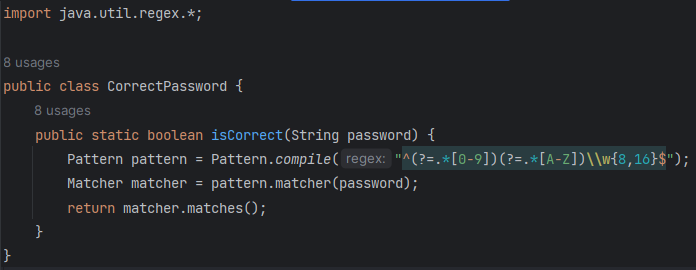


Рисунок 4 Второе задание

Протестируем программу как показано на рисунке 5.

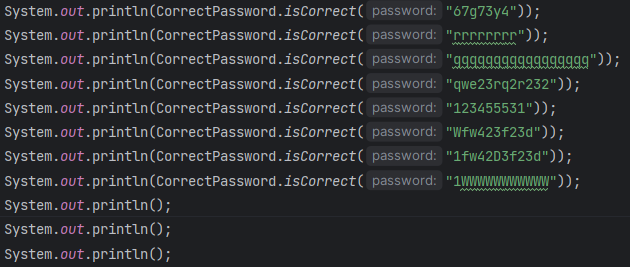


Рисунок 5 Тест 2 задания

Результат выполнения программы показан на рисунке 6.

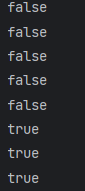


Рисунок 6 Результат работы второй программы

## 3 задание

В третьем задании необходимо написать программу, которая будет заменять все ссылки в заданном тексте на гиперссылки. При этом программа должна использовать регулярные выражения для поиска ссылок и замены и обрабатывать возможные ошибки.

В данной программе нам необходимо исключать такие подстроки, которые уже начинаются на http:// или https://. После этого мы находим строки типа <символы>.<символы>. После чего к каждому найденному значению дописываем в начале http://. Код данной программы показан на рисунке 7

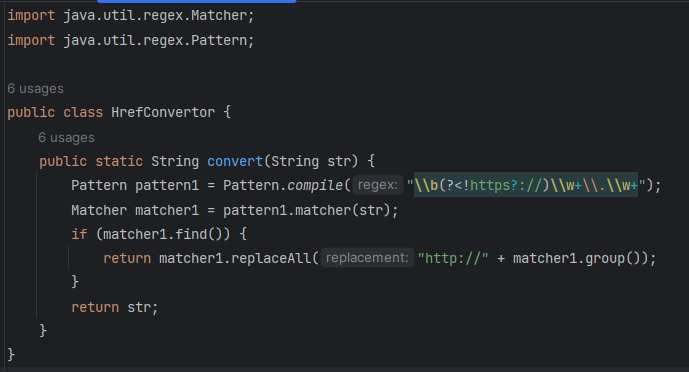


Рисунок 7 Третье задание

Протестируем программу как показано на рисунке 8.

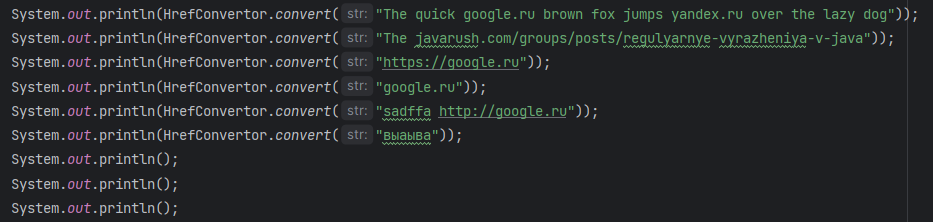


Рисунок 8 Тест 3 задания

Результат работы данной программы показан на рисунке 9.



Рисунок 9 Результат работы третьей программы

## 4 задание

В четвертом задании необходимо написать программу, которая будет проверять корректность ввода IP-адреса. IP-адрес должен состоять из 4 чисел, разделенных точками, и каждое число должно быть в диапазоне от 0 до 255. При этом программа должна использовать регулярные выражения для проверки IP-адреса и обрабатывать возможные ошибки.

Для начала укажем начало и конец строки символами ^ и $. Далее указываем, что после 0 и 1 может быть 1 или 2 любые цифры, после 2 должна стоять цифра от 0 до 4 и еще 1 любая, а после 25 только цифры 0-5. После данная комбинация повторяется с точкой в начале еще 3 раза. Код данной программы показан на рисунке 10.

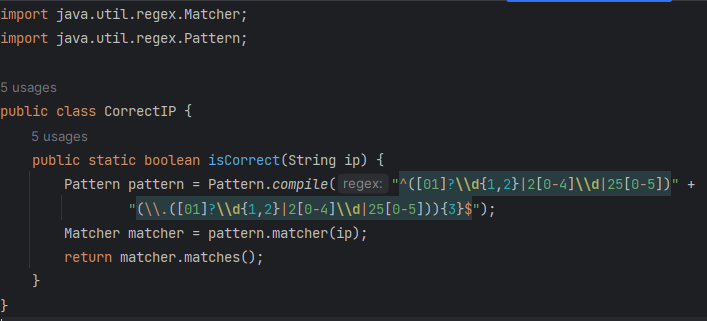


Рисунок 10 Четвертое задание

Протестируем программу как показано на рисунке 11.

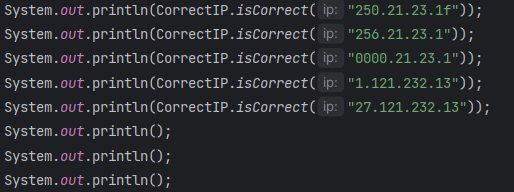


Рисунок 11 Тест 4 задания

Результат работы данной программы показан на рисунке 12.

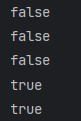


Рисунок 12 Результат работы четвертой программы

## 5 задание

В пятом задании нужно написать программу, которая будет искать все слова в заданном тексте, начинающиеся с заданной буквы, и выводить их на экран. При этом программа должна использовать регулярные выражения для поиска слов и обрабатывать возможные ошибки.

Для начала укажем начало и конец слова символами [\\b](file:///\\b). После чего должен следовать наш символ. После необходимого символа следует любое количество букв. Pattern.CASE\_INSENSITIVE указывает, что нам не важен регистр при поиске. Код данной программы показан на рисунке 13.

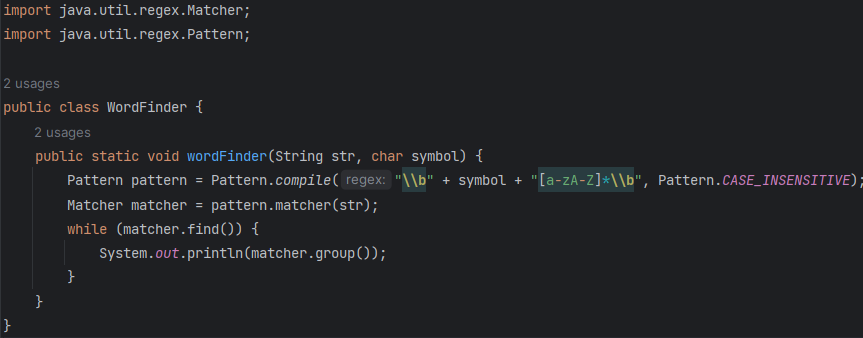


Рисунок 13 Пятое задание

Протестируем программу как показано на рисунке 14.



Рисунок 14 Тест 5 задания

Результат работы данной программы показан на рисунке 15.

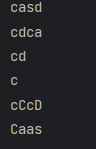


Рисунок 15 Результат работы 5 программы

# ВЫВОД

Данная лабораторная работа помогла лучше освоить язык Java. Была проделана работа с регулярными выражениями. Также были освоены классы Pattern и Matcher. Все знания были закреплены на практике и сохранены на GitHub: <https://github.com/HungryM1NT/ITIP>