Министерство образования Республики Беларусь

Учреждение высшего образования

БЕЛОРУССКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ

ИНФОРМАТИКИ И РАДИОЭЛЕКТРОНИКИ

Факультет компьютерных систем и сетей

Кафедра электронных вычислительных машин

ОТЧЕТ

по лабораторной работе №3

на тему

**Работа со строками**

|  |  |
| --- | --- |
| Студенты группы 150501  Проверила | Михалович Т.В.  Петров В.В.  Герман Ю. О. |

Минск 2023

**Цель:** изучить технику работы со строками в Scala.

**Краткие теоретические сведения:**

Регулярное выражение (или регулярное выражение, regex) в Scala представляет собой шаблон, описывающий набор символов. Оно используется для поиска, сопоставления и манипуляции текстом в строках.

В Scala, работа с регулярными выражениями осуществляется с помощью класса scala.util.matching.Regex, который предоставляет методы для работы с текстом с использованием регулярных выражений.

несколько примеров шаблонов для регулярных выражений:

1. **Поиск цифр:**
   * Шаблон: **\d+**
   * Описание: Этот шаблон ищет одну или более цифр подряд.
2. **Поиск электронной почты:**
   * Шаблон: **[\w\.-]+@[a-zA-Z\d\.-]+\.[a-zA-Z]{2,}**
   * Описание: Этот шаблон соответствует стандартному формату электронной почты.
3. **Поиск URL:**
   * Шаблон: **(http|https)://[^\s]+**
   * Описание: Этот шаблон ищет ссылки, начинающиеся с "http://" или "https://".
4. **Поиск слов, начинающихся с определенной буквы:**
   * Шаблон: **\b[Aa]\w\***
   * Описание: Этот шаблон находит слова, начинающиеся с буквы "A" или "a".
5. **Поиск даты в формате MM/DD/YYYY:**
   * Шаблон: **^(0[1-9]|1[0-2])/(0[1-9]|[12][0-9]|3[01])/\d{4}$**
   * Описание: Этот шаблон ищет даты в формате "месяц/день/год".

Класс StringBuilder в Scala представляет собой изменяемую последовательность символов (mutable sequence of characters), которая предназначена для построения строк. Он эффективен при частых операциях конкатенации или изменения строк в циклах, так как избегает накладных расходов, связанных с неизменяемыми строками.

Вот пример использования класса StringBuilder в Scala:

val stringBuilder = new StringBuilder

// Добавление текста в StringBuilder

stringBuilder.append("Hello")

stringBuilder.append(" ")

stringBuilder.append("World")

// Преобразование в строку

val resultString: String = stringBuilder.toString()

// Вывод результата

println(resultString)

val myList = List(1, 2, 3, 4, 5)

**Выполнение работы:**

Перед выполнением основного задания лабораторной работы мы выполнили рекомендованные к изучению примеры.

В первом примере мы познакомились с функцией для работы со строками replaceAll в языке программирования Scala. Результатом исполнения программы является нижеприведенный скриншот:

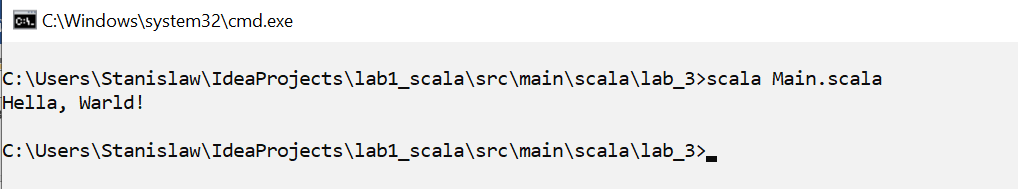


Рисунок 1 — Результат выполнения первого примера

Во втором примере мы познакомились с функцией для работы со строками split в языке программирования Scala. Результатом исполнения программы является нижеприведенный скриншот:

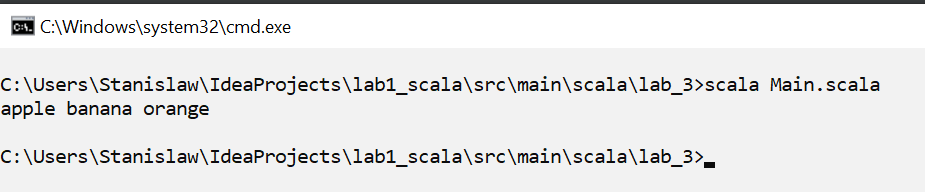


Рисунок 2 — Результат выполнения второго примера

В третьем примере мы познакомились с функцией для работы со строками startsWith, endsWith в языке программирования Scala. Результатом исполнения программы является нижеприведенный скриншот:

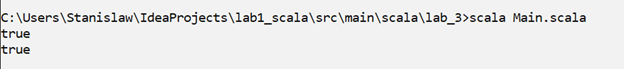


Рисунок 3 — Результат выполнения третьего примера

В четвертом примере мы познакомились с функцией для работы со строками substring в языке программирования Scala. Результатом исполнения программы является нижеприведенный скриншот:

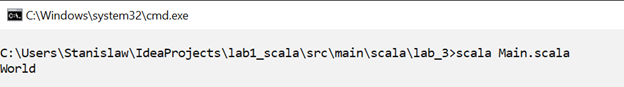


Рисунок 4 — Результат выполнения четвертого примера

В пятом примере мы познакомились с функцией для работы со строками toCharArray в языке программирования Scala. Результатом исполнения программы является нижеприведенный скриншот:

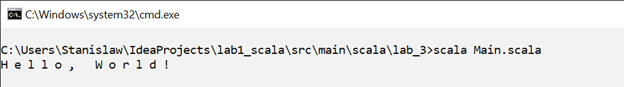


Рисунок 5 — Результат выполнения пятого примера

В ходе выполнения лабораторной работы был произведен анализ 8-ми заданий.

Задание 1: Дан текст: ‘Hello to everybody’. C помощью техники регулярных выражений заменить латинские буквы на русские (или на цифры, если русский шрифт не поддерживается)

Листинг кода программы:

val text = "Hello to everybody"

val immutableMap = Map('H' -> 'Н',

'e' -> 'е',

'l' -> 'л',

'o' -> 'о',

't' -> 'т',

'v' -> 'в',

'r' -> 'р',

'y' -> 'у',

'b' -> 'б',

'd' -> 'д'

)

val replaced = text.map(char => immutableMap.getOrElse(char.toLower, char))

println(replaced)

Результаты выполнения задания представлен скриншотам ниже:

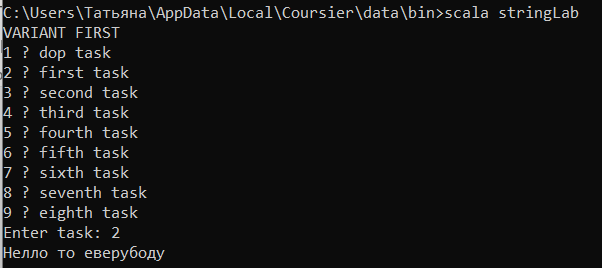


Рисунок 6 – Результат выполнения 1 задания

Задание 2: Найти в тексте “When executing the exercise extract all extra words” все слова, начинающиеся на ext.

Листинг кода программы:

val text = "When executing the exercise extract all extra words"

val regex = "\\bext\\w\*".r

val matches = regex.findAllIn(text).toList

matches.foreach(println)

Результаты выполнения задания представлен скриншотам ниже:

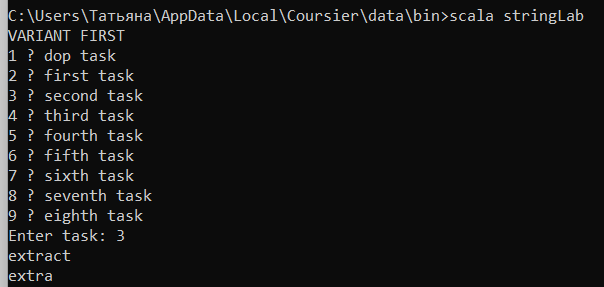


Рисунок 7 – Результат выполнения 2 задания

Задание 3: В тексте ‘A big round ball fall to the ground’ заменить артикль the на a.

Листинг кода программы:

val text = "A big round ball fall to the ground"

val words = text.split("\\s+")

def replaceTheWithA(word: String): String = {

if (word.toLowerCase == "the") {

"a"

} else {

word

}

}

val replacedText = words.map(replaceTheWithA).mkString(" ")

println(replacedText)

Результаты выполнения задания представлен скриншотам ниже:

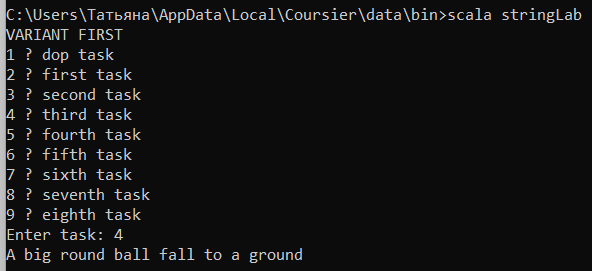


Рисунок 8 – Результат выполнения 3 задания

Задание 4: Записать все слова в тексте в обратном порядке.

Листинг кода программы:

val text = "A big round ball falls to the ground"

val words = text.split("\\s+")

val reversedWords = words.reverse

val reversedText = reversedWords.mkString(" ")

println(reversedText)

Результаты выполнения задания представлен скриншотам ниже:

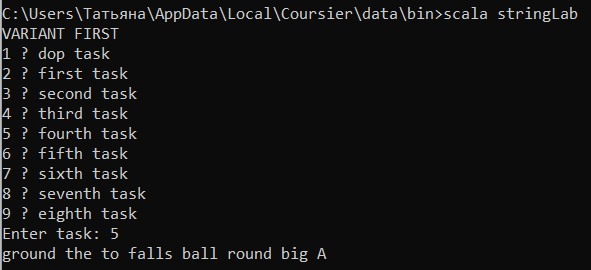


Рисунок 9 – Результат выполнения 4 задания

Задание 5: Дан текст: ‘Hello to everybody’. Выбросить все гласные.

Листинг кода программы:

val text = "Hello to everybody"

def removeVowels(input: String): String = {

input.replaceAll("[aeiouAEIOU]", "")

}

val textWithoutVowels = removeVowels(text)

println(textWithoutVowels)

Результаты выполнения задания представлен скриншотам ниже:

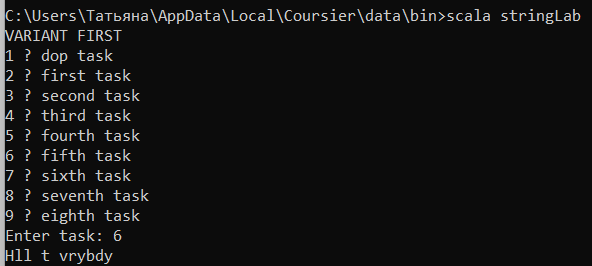


Рисунок 10 – Результат выполнения 5 задания

Задание 6: Дан текст: ‘Hello to everybody’. Удвоить каждую букву в слове

Листинг кода программы:

val text = "Hello to everybody"

def doubleLettersInWord(word: String): String = {

word.flatMap(char => char.toString \* 2)

}

val words = text.split("\\s+")

val textWithDoubledLetters = words.map(doubleLettersInWord).mkString(" ")

println(textWithDoubledLetters)

Результаты выполнения задания представлен скриншотам ниже:

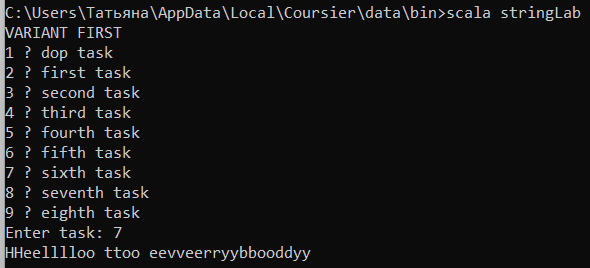


Рисунок 11 – Результат выполнения 6 задания

Задание 7: Дан текст: ‘Hello to everybody’. Удалить все вхождения буквы y

Листинг кода программы:

val text = "Hello to everybody"

def removeLetterY(input: String): String = {

input.replaceAll("y", "")

}

val textWithoutLetterY = removeLetterY(text)

println(textWithoutLetterY)

Результаты выполнения задания представлен скриншотам ниже:

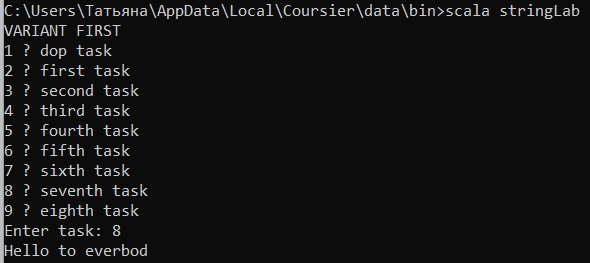


Рисунок 12 – Результат выполнения 7 задания

Задание 8: Дан текст: ‘Hello to everybody’. Вставить слова with heartness чтобы получить

Hello with heartness to everybody

Листинг кода программы:

val text = "Hello to everybody"

val wordToInsert = "with heartness"

val words = text.split("\\s+")

val positionToInsert = 1

val modifiedWords = words.patch(positionToInsert, Seq(wordToInsert), 0)

val modifiedText = modifiedWords.mkString(" ")

println(modifiedText)

}

Результаты выполнения задания представлен скриншотам ниже:

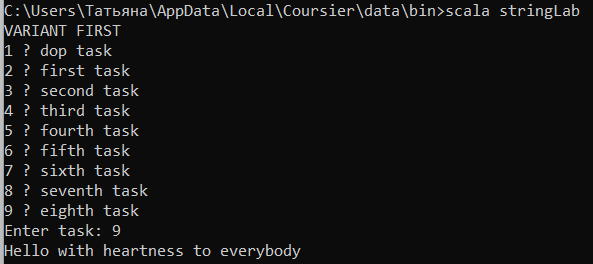


Рисунок 13 – Результат выполнения 8 задания

Индивидуальное задание: Вывести суммарное число всех гласных в собственном тексте.

Листинг кода программы:

val text = scala.io.StdIn.readLine()

val vowels = Set('а', 'е', 'ё', 'и', 'о', 'у', 'ы', 'э', 'ю', 'я', 'А', 'Е', 'Ё', 'И', 'О', 'У', 'Ы', 'Э', 'Ю', 'Я',

'a', 'e', 'i', 'o', 'y', 'u', 'A', 'E', 'I', 'O', 'Y', 'U')

val vowelCount = text.count(char => vowels.contains(char))

println(s"Суммарное число гласных в тексте: $vowelCount")

Результаты выполнения задания представлен скриншотам ниже:

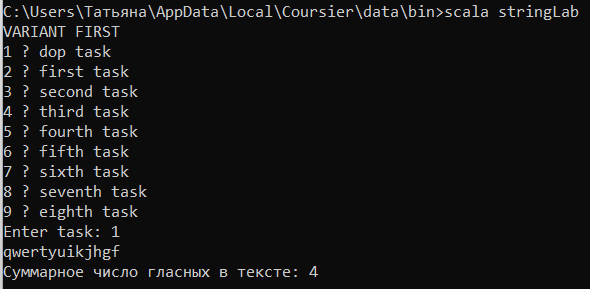


Рисунок 14 – Результат выполнения индивидуального задания

Вывод: В ходе выполнения данной лабораторной работы нами были проанализированы и изучены техники работы и использования функций для строк в Scala.