Лабораторная работа №1

Julia. Установка и настройка. Основные принципы.

Беличева Дарья Михайловна

Содержание

# 1 Цель работы

Основная цель работы — подготовить рабочее пространство и инструментарий для работы с языком программирования Julia, на простейших примерах познакомиться с основами синтаксиса Julia.

# 2 Задание

1. Установите под свою операционную систему Julia, Jupyter.
2. Используя Jupyter Lab, повторите примеры из раздела лабораторной работы.
3. Выполните задания для самостоятельной работы.

# 3 Теоретическое введение

Julia – высокоуровневый свободный язык программирования с динамической типизацией, созданный для математических вычислений [1]. Эффективен также и для написания программ общего назначения. Синтаксис языка схож с синтаксисом других математических языков, однако имеет некоторые существенные отличия.

Для выполнения заданий была использована официальная документация Julia [2].

# 4 Выполнение лабораторной работы

У меня уже были установлены Julia и Jupyter (рис. 1).

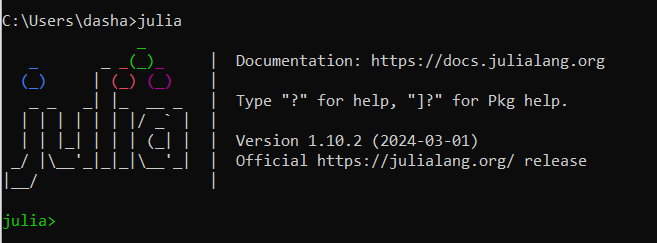


Рис. 1: Запуск Julia

Теперь повторим простейшие примеры для знакомства с синтаксисом Julia (рис. 1-4).



Рис. 2: Выполнение примеров из лабораторной

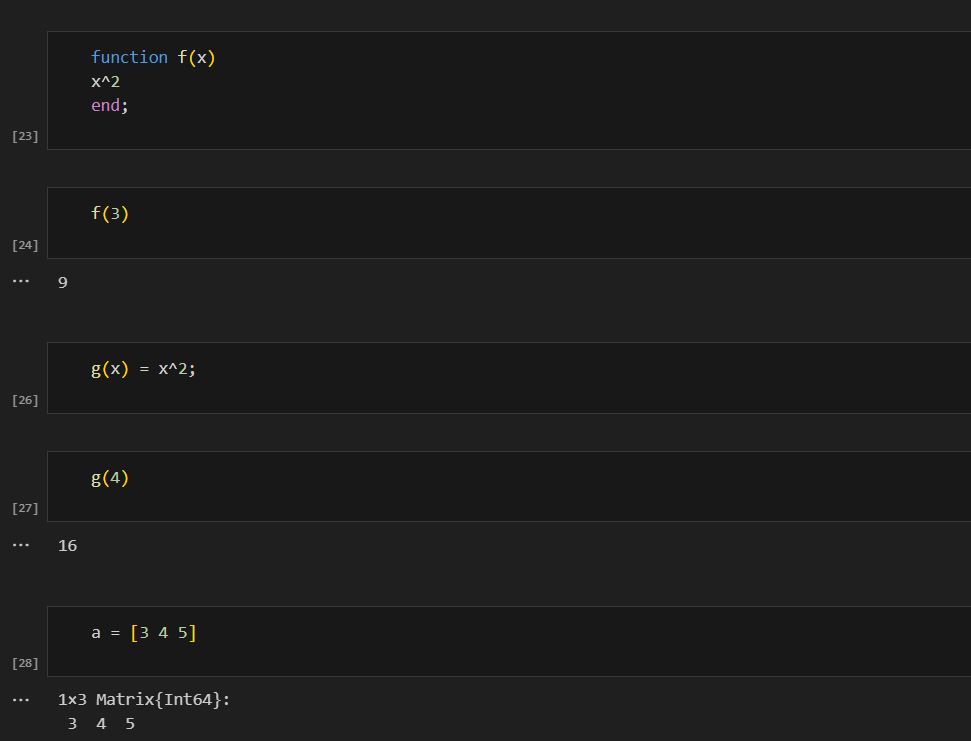


Рис. 3: Выполнение примеров из лабораторной

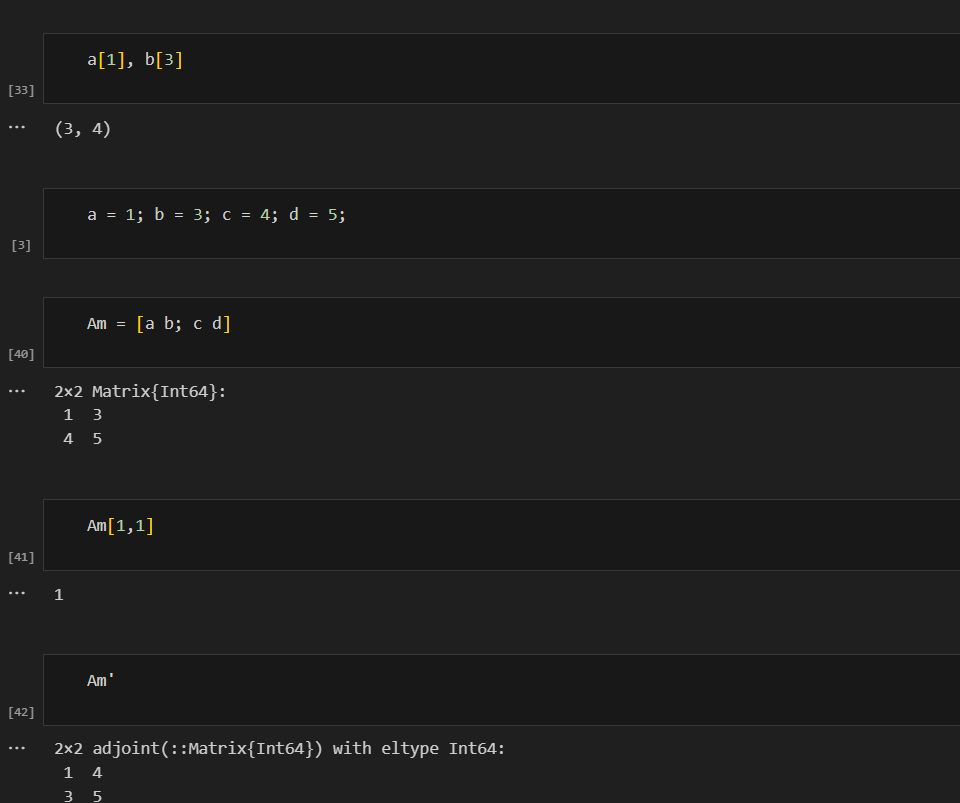


Рис. 4: Выполнение примеров из лабораторной

Теперь перейдем к выполнению заданий.

**Задание №1**

Изучим документацию по основным функциям Julia для чтения / записи / вывода информации на экран: read(), readline(), readlines(), readdlm(), print(), println(), show(), write(). Приведем свои примеры их использования, поясняя особенности их применения.

Для того, чтобы ознакомиться с документацией достаточно поставить знак ? перед интересующей функцией. Например, ?print.

Создадим текстовый файл с любым содержанием в папке, где мы работаем. Откроем его на чтение и прочитаем с помощью команды read(). Текст вывелся в одну строку с разделителями \r\n. Также прочитаем текст используя функцию readline() - выведется только первая строка. Чтобы прочитать все строки в файле используем команду readlines() (рис. 5).



Рис. 5: Чтение файла

Далее посмотрим, как работает команда println(), print() и show() (рис. 6-7).

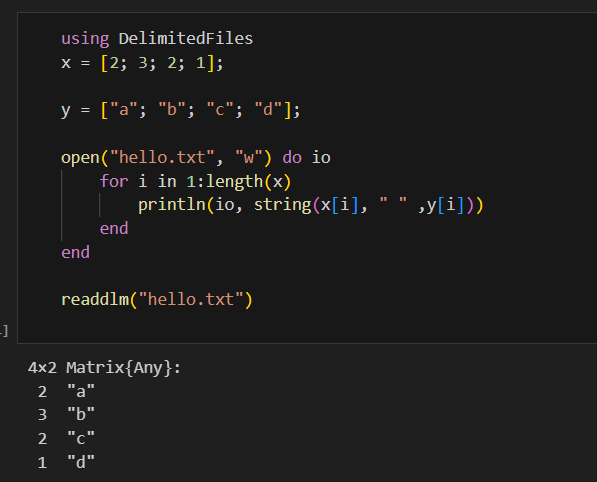


Рис. 6: Вывод на печать



Рис. 7: Вывод на печать

Посмотрим на работу функции write() (рис. 8).

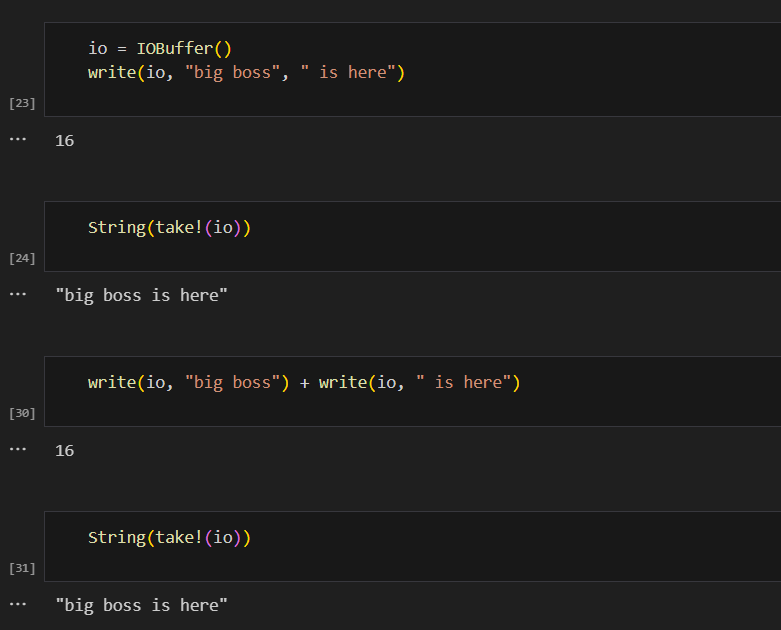


Рис. 8: Команда записи

**Задание №2**

Изучим документацию по функции parse() (рис. 9). Приведем свои примеры её использования, поясняя особенности её применения (рис. 10).

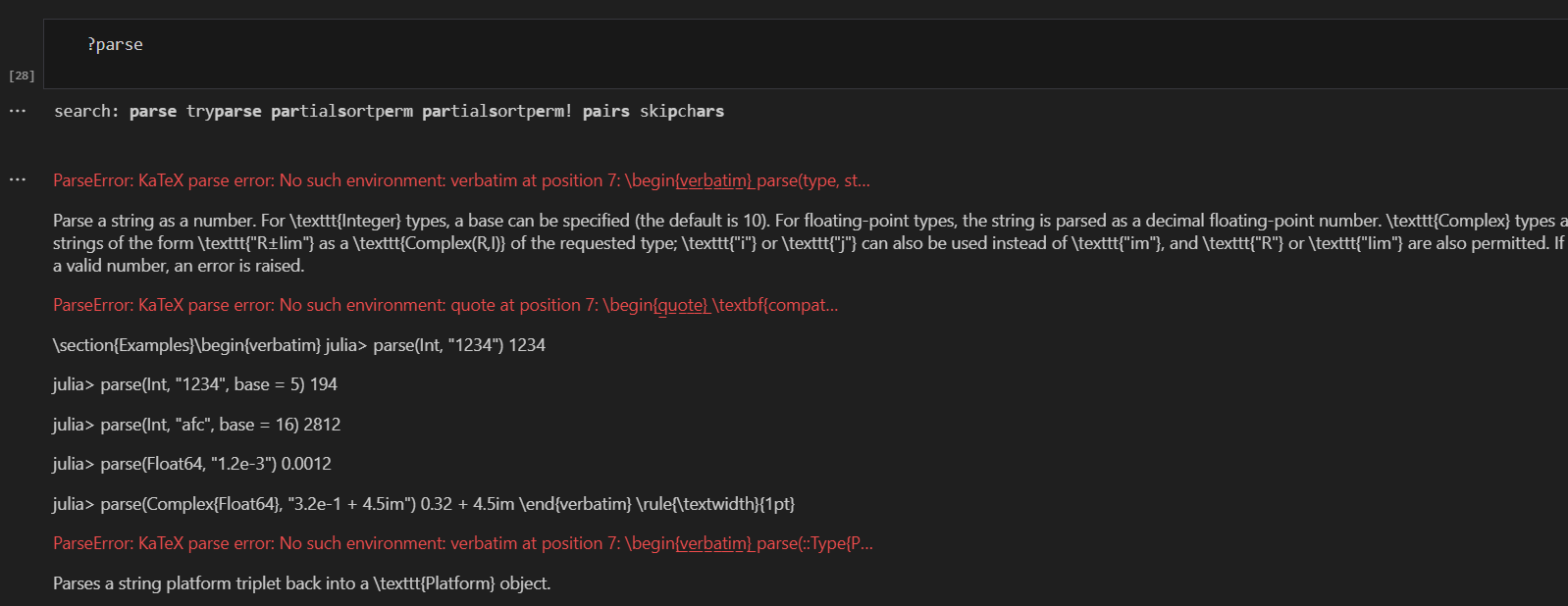


Рис. 9: Документация по функции parse()



Рис. 10: Примеры использования функции parse()

**Задание №3**

Изучим синтаксис Julia для базовых математических операций с разным типом переменных: сложение, вычитание, умножение, деление, возведение в степень, извлечение корня, сравнение, логические операции. Приведем примеры с пояснениями по особенностям их применения (рис. 11-12).



Рис. 11: Примеры базовых математических операций

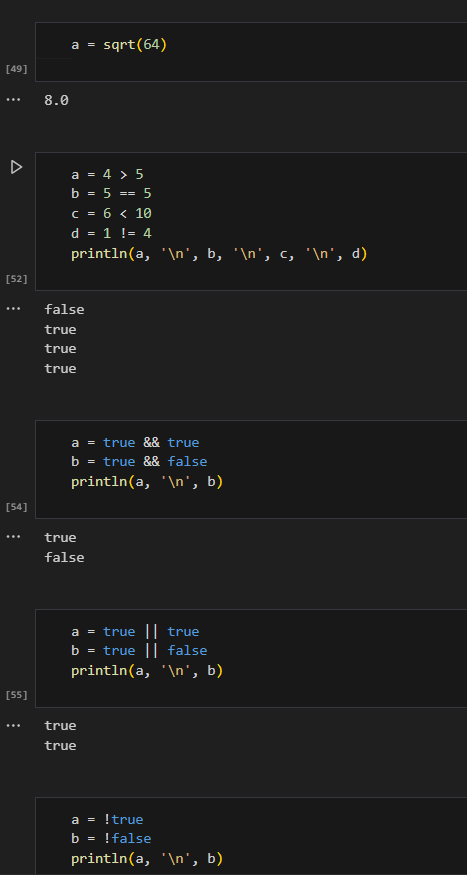


Рис. 12: Примеры базовых математических операций

**Задание №4**

Приведем несколько примеров с пояснениями с операциями над матрицами и векторами: сложение, вычитание, скалярное произведение, транспонирование, умножение на скаляр (рис. 13-14).



Рис. 13: примеры операций над матрицами

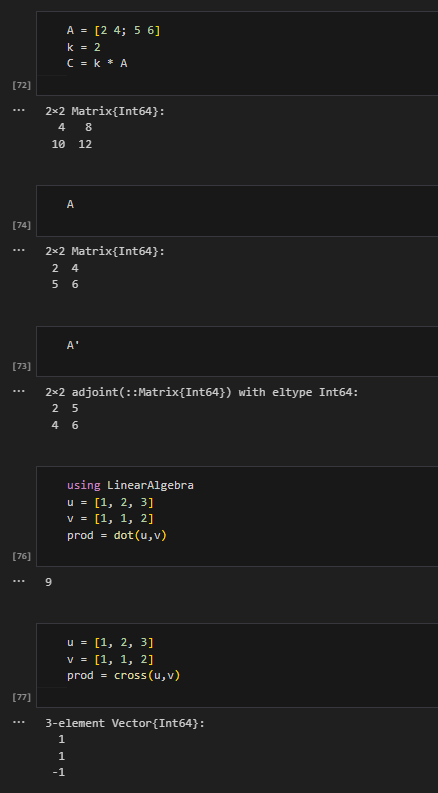


Рис. 14: примеры операций над векторами

# 5 Выводы

В результате выполнения данной лабораторной работы я подготовила рабочее пространство и инструментарий для работы с языком программирования Julia, на простейших примерах познакомилась с основами синтаксиса Julia.

# Список литературы

1. JuliaLang [Электронный ресурс]. 2024 JuliaLang.org contributors. URL: <https://julialang.org/> (дата обращения: 11.10.2024).

2. Julia 1.11 Documentation [Электронный ресурс]. 2024 JuliaLang.org contributors. URL: <https://docs.julialang.org/en/v1/> (дата обращения: 11.10.2024).