

# **Лабораторная работа 1**

**Имитационное моделирование**

Богданюк Анна

# **Содержание**

<b>1 Цель работы</b>	<b>5</b>
<b>2 Задание</b>	<b>6</b>
<b>3 Теоретическое введение</b>	<b>7</b>
<b>4 Выполнение лабораторной работы</b>	<b>9</b>
<b>5 Выводы</b>	<b>13</b>
<b>Список литературы</b>	<b>14</b>

# **Список иллюстраций**

4.1	Создаём репозиторий по шаблоны . . . . .	9
4.2	Репозиторий на github.com . . . . .	9
4.3	Результат работы test_setup.jl . . . . .	10
4.4	Результат программы 01_exponential_growth.jp . . . . .	10
4.5	Результат программы 02_exponential_growth.jl . . . . .	11
4.6	Результат программы 02_exponential_growth.jl . . . . .	11
4.7	Результат программы 02_exponential_growth.jl . . . . .	12
4.8	Результат программы 02_exponential_growth.jl . . . . .	12

# **Список таблиц**

# **1 Цель работы**

Целью данной лабораторной работы является создание рабочего пространства для выполнения лабораторных работ по курсу «Математическое моделирование», настройка инструментов программной инженерии (Git, Git-flow, семантическое версионирование, общепринятые коммиты), а также освоение методологии литературного программирования на примере языка Julia и пакета DrWatson.

## **2 Задание**

1. Создать рабочий каталог для всего курса.
2. Создать рабочее пространство для программ в рамках лабораторной работы.
3. Выполнить все задания по тексту лабораторной работы.
4. Установить необходимые пакеты.
5. Выполнить предложенный код.
6. Преобразовать код в литературный стиль.
7. Сгенерировать из литературного кода:
  8. чистый код;
  9. jupyter notebook;
10. документацию в формате Quarto.
11. Выполнить код из jupyter notebook.
12. Интегрировать документацию в формате Quarto в отчёт.
13. Добавить в код в литературном стиле вычисление для набора параметров.
14. Сгенерировать из литературного кода с параметрами:
  15. чистый код;
  16. jupyter notebook;
17. документацию в формате Quarto.
18. Выполнить код из jupyter notebook с параметрами.
19. Интегрировать документацию с параметрами в формате Quarto в отчёт.

## 3 Теоретическое введение

В ходе работы были изучены и применены следующие концепции и инструменты программной инженерии:

- Семантическое версионирование (SemVer): Стандарт версионирования программного обеспечения, использующий формат МАЖОРНАЯ.МИНОРНАЯ.ПАТЧ. Увеличение номера версии сигнализирует о характере изменений в API.
- Общепринятые коммиты (Conventional Commits): Спецификация для написания сообщений коммитов. Она определяет набор правил для создания понятной истории изменений.
- Git и Git-flow: Распределенная система контроля версий Git и модель ветвления Git-flow, которая предполагает использование двух основных веток (master и develop), а также вспомогательных (feature, release, hotfix) для организации процесса разработки.
- Верификация коммитов: Использование GPG-ключей для подписи коммитов, что позволяет подтвердить их подлинность на хостингах (GitHub/GitVerse).
- Литературное программирование: Подход, предложенный Дональдом Кнутом, при котором программа пишется как литературное эссе, где код является лишь частью повествования. В Julia для этого используется пакет Literate.jl.

- DrWatson.jl: Фреймворк для управления научными проектами на Julia. Он обеспечивает стандартную структуру каталогов, упрощает сохранение и загрузку результатов, а также способствует воспроизводимости исследований.

## 4 Выполнение лабораторной работы

Для начала создаём рабочий каталог для курса, используя шаблон (рис. 4.1).

```
avbogdanyuk@COMPUTER:~/work/study/2026-1/2026-1==study--mathmod/2026-1==study--mathmod$ mkdir -p ~/work/study/2026-1==study--mathmod
avbogdanyuk@COMPUTER:~/work/study/2026-1==study--mathmod/2026-1==study--mathmod$ cd ~/work/study/2026-1/2026-1==study--simulation-modeling
avbogdanyuk@COMPUTER:~/work/study/2026-1/2026-1==study--simulation-modeling$ gh repo create 2026-1==study--mathmod --template=yanadharma/course-directory-student-template --public
avbogdanyuk@COMPUTER:~/work/study/2026-1/2026-1==study--simulation-modeling$ gh repo create 2026-1==study--simulation-modeling --template=yanadharma/course-directory-student-template --public
✓ Created repository avbogdanyuk/2026-1==study--simulation-modeling on GitHub
https://github.com/avbogdanyuk/2026-1==study--simulation-modeling
```

Рисунок 4.1: Создаём репозиторий по шаблону

Готовый репозиторий для курса на [github.com](https://github.com) (рис. 4.2).

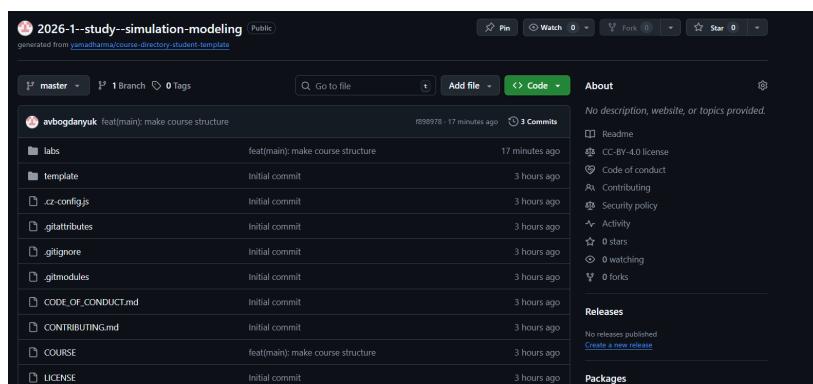


Рисунок 4.2: Репозиторий на [github.com](https://github.com)

Создаём рабочее пространство для программ в рамках лабораторной работы.

Установить необходимые пакеты. На скриншоте показан результат работы программы на языке julia, чтобы проверить скачанные материалы(рис. 4.3).

```

    / Plots
    / DataFrames
    / CSV
    / JLD2
    / Literate
    / IJulia
    / BenchmarkTools
    / Quarto

Структура проекта:
Корень: /home/avbogdanyuk/work/study/2026-1/2026-1==study--simulation-modeling/2026-1--study--simul
ab01/project
    Данные: /home/avbogdanyuk/work/study/2026-1/2026-1==study--simulation-modeling/2026-1--study--simul
ab01/project/data
    Скрипты: /home/avbogdanyuk/work/study/2026-1/2026-1==study--simulation-modeling/2026-1--study--simu
lab01/project/src
    Графики: /home/avbogdanyuk/work/study/2026-1/2026-1==study--simulation-modeling/2026-1--study--simu
lab01/project/plots
avbogdanyuk@COMPUTER:~/work/study/2026-1/2026-1==study--simulation-modeling/2026-1--study--simulat
/project$ ls
Manifest.toml Project.toml add_packages.jl scripts
avbogdanyuk@COMPUTER:~/work/study/2026-1/2026-1==study--simulation-modeling/2026-1--study--simulat
/project$ |

```

Рисунок 4.3: Результат работы test\_setup.jl

Копируем код программы на языке julia из методички по лабораторной работе №1. Компилируем, получим результат в виде графика, который будет сохранён в папке plots (рис. 4.4).

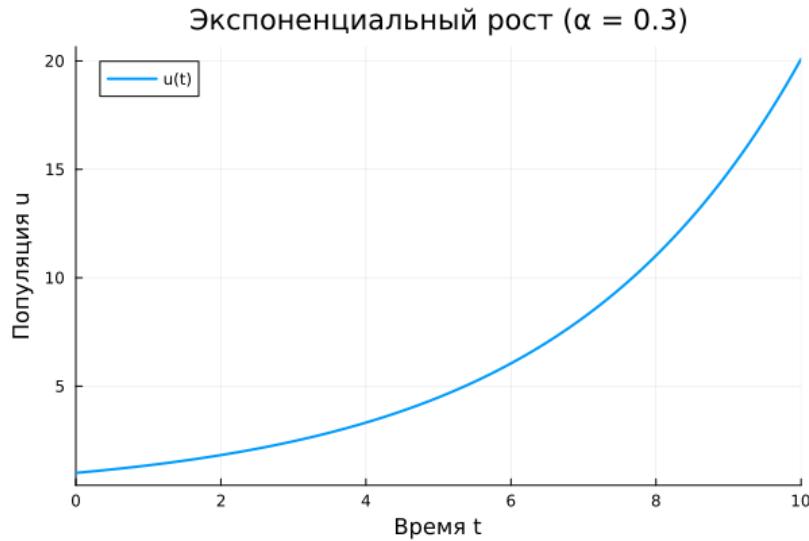


Рисунок 4.4: Результат программы 01\_exponential\_growth.jl

Теперь создаём код для генерации нескольких видов графиков экспоненциального распределения scripts/02\_exponential\_growth.jl. Для начала базовый эксперимент (рис. 4.5).

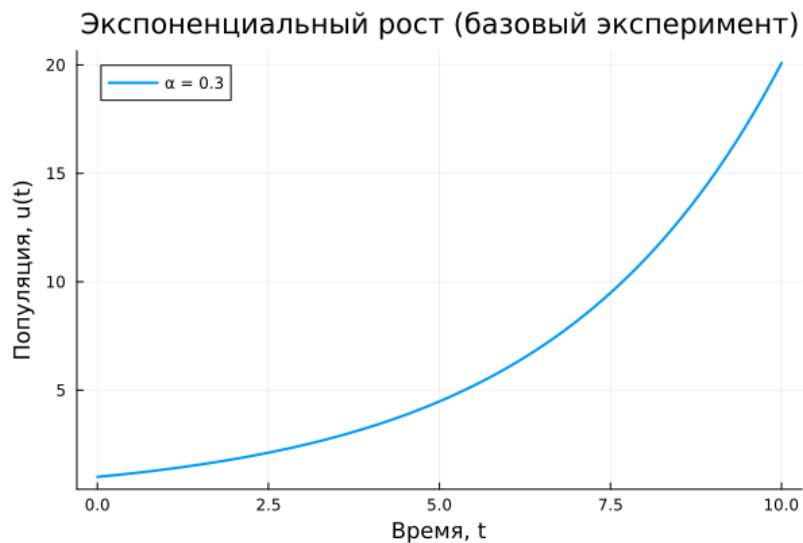


Рисунок 4.5: Результат программы 02\_exponential\_growth.jl

Сравнительный анализ всех экспериментов (рис. 4.6).

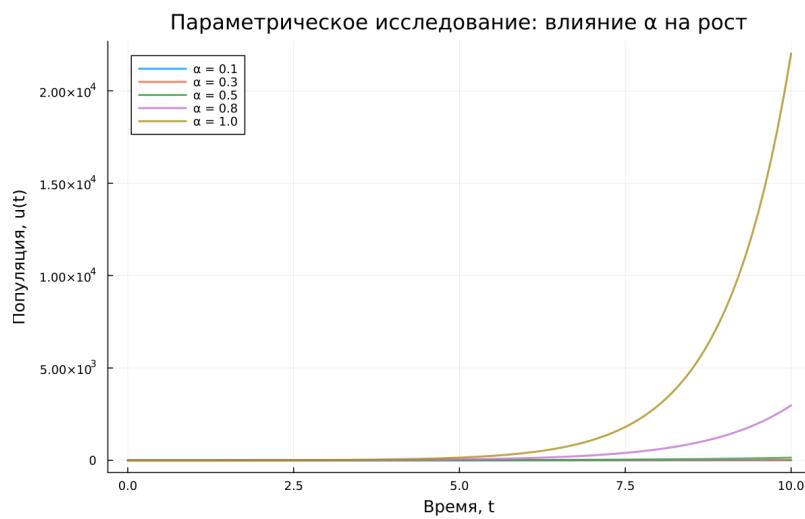


Рисунок 4.6: Результат программы 02\_exponential\_growth.jl

График зависимости времени удвоения от  $\alpha$  (рис. 4.7).

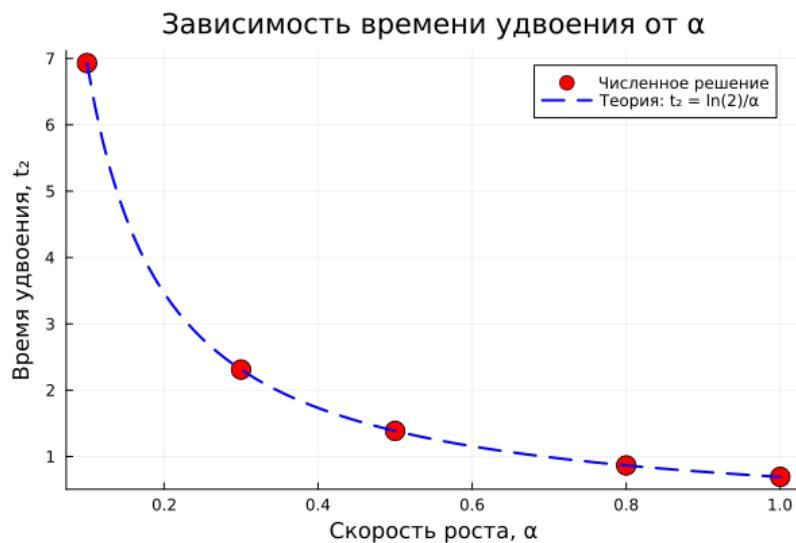


Рисунок 4.7: Результат программы 02\_exponential\_growth.jl

График зависимости времени вычисления от  $\alpha$  (рис. 4.8).

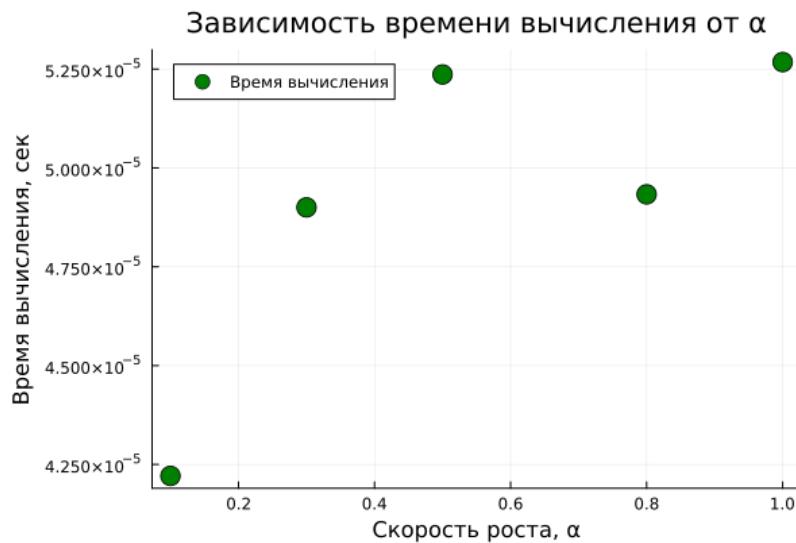


Рисунок 4.8: Результат программы 02\_exponential\_growth.jl

## **5 Выводы**

В ходе выполнения лабораторной работы было создано структурированное рабочее пространство для курса «Математическое моделирование». Освоены базовые инструменты и практики программной инженерии: система контроля версий Git с моделью ветвления Git-flow, семантическое версионирование и стандарт оформления коммитов Conventional Commits. Настроена безопасная работа с удаленными репозиториями с помощью SSH и PGP ключей для верификации коммитов.

## **Список литературы**