

# Защита лабораторной работы

Лабораторная работа №3 (вариант 10)

---

Сергее Т.С.

09 февраля 2023

Российский университет дружбы народов, Москва, Россия

## Информация

---

- Сергеев Тимофей Сергеевич
- Студент 3 курса группы НФИбд-02-20
- Студенческий билет №1032201669
- Российский университет дружбы народов
- 1032201669@pfur.ru

## Вводная часть

---

- Данная работа нацелена на изучение языков программирования Julia и Modelics, созданных для выполнения математических вычислений и моделирования.

- Консоль компьютера
- Язык программирования Julia
- Язык программирования Modelica

- Построить математическую модель боевых действий двух стран - X и Y. Рассмотреть два случая: с участием партизанских отрядов и без.
- Написать код для первого и второго случая на языке Julia.
- Написать код для первого и второго случая на языке Modelica.
- Составить отчёт на языке Markdown и сконвертировать его в docx и pdf.
- Подготовить презентацию на языке Markdown и защитить её.

## Выполнение работы

---



```
1  model lab31
2    Real x, y;
3    Real a=0.45;
4    Real b=0.86;
5    Real c=0.49;
6    Real h=0.73;
7    Real t=time;
8    initial equation
9    x=21200;
10   y=9800;
11   equation
12   der(x)= -a*x - b*y + sin(t+1);
13   der(y)= -c*x - h*y + cos(t+2);
14   end lab31;
```

Рис. 1: Реализация первого случая на языке Modelica

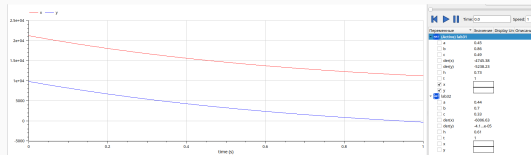


Рис. 2: Результат симуляции для первого случая

```
1 model lab32
2 Real x, y;
3 Real a=0.44;
4 Real b=0.7;
5 Real c=0.33;
6 Real h=0.61;
7 Real t=time;
8 initial equation
9 x=21200;
10 y=9800;
11 equation
12 der(x)= -a*x - b*y + sin(2*t);
13 der(y)= -c*x*y - h*y + cos(t)+1;
14 end lab32;
```

Рис. 3: Реализация второго случая на языке Modelica

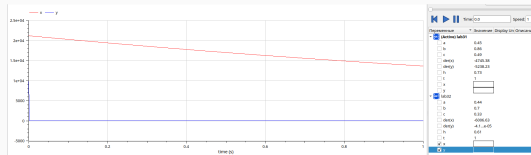


Рис. 4: Результат симуляции для второго случая

```
1 using DifferentialEquations
2 using Plots
3
4 a=0.45
5 b=0.86
6 c=0.49
7 h=0.73
8
9 function F!(du, u, p, t,)
10     du[1] = -a*u[1] - b*u[2] + sin(t+1)
11     du[2] = -c*u[1] - h*u[2] + cos(t+2)
12 end
13
14 begin
15     u0 = [21200, 9800]
16     T = [0.0, 1.0]
17     prob = ODEProblem(F!, u0, T)
18 end
19
```

Рис. 5: Подключаем библиотеки, задаём значение коэффициентов, задаём нашу функцию, решающую дифференциальные уравнения. Затем зададим значения наших войск и времени.

```
sol = solve(prob, dtmax=0.05)

const X = Float64[]
const Y = Float64[]

for u in sol.u
    x, y = u
    push!(X, x)
    push!(Y, y)
end

plt = plot(
    dpi = 300,
    size=(1000,600),
    plot_title="Модель боевых действий без участия партизанских отрядов"
)
```

Рис. 6: Выполним функцию с данными значениями. Создадим два пустых массива, в которые мы передадим полученные значения. Затем с помощью функционала библиотеки Plots создадим поле для вывода результата.

```
plot!(
    plt,
    sol.t,
    X,
    xlabel="Время",
    color=:blue,
    label="Армия страны X"
)

plot!(
    plt,
    sol.t,
    Y,
    ylabel="Армия",
    color=:red,
    label="Армия страны Y"
)

savefig(plt, "lab31.png")
```

Рис. 7: Выведем на экран полученные графы и сохраним результат в формате png

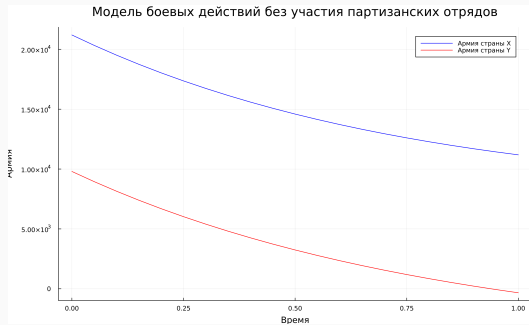


Рис. 8: Результат работы программы для первого случая



```
using DifferentialEquations
using Plots

a=0.44
b=0.7
c=0.33
h=0.61

function F!(du, u, p, t,)
    du[1] = -a*u[1] - b*u[2] + sin(2*t)
    du[2] = -c*u[1]*u[2] - h*u[2] + cos(t)+1
end

begin
    u0 = [21200, 9800]
    T = [0.0, 1.0]
    prob = ODEProblem(F!, u0, T)
end

sol = solve(prob, dtmax=0.05)

const X = Float64[]
const Y = Float64[]
```

```
for u in sol.u
    x, y = u
    push!(X, x)
    push!(Y, y)
end

plt = plot(
    dpi = 300,
    size=(800,600),
    plot_title="Модель боевых действий с участием партизанских отрядов"
)
```

Рис. 10: Выполним функцию с данными значениями. Создадим два пустых массива, в которые мы передадим полученные значения. Затем с помощью функционала библиотеки Plots создадим поле для вывода результата.

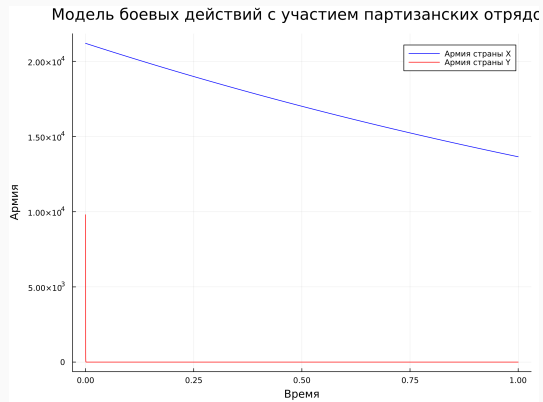


Рис. 11: Результат работы программы для первого случая

## Результаты

---

Выполнив данную лабораторную работу, мы продолжили знакомство с языком программирования Julia, начали изучение языка Modelica. Сравнивая реализацию одной программы на этих двух языках, можно заметить, что реализация на языке Modelica заметно проще и более точно показывает результат, поскольку можно отследить значения переменных с максимальной точностью на любом отрезке графика.

Вовремя выполненная лабораторная работа - хорошая оценка - довольный студент -  
счастливое будущее!