

---

## Front matter

title: "Лабораторная работа №2"  
subtitle: "Задача о погоне"  
author: "Камкина Арина Леонидовна"

## Generic options

lang: ru-RU  
toc-title: "Содержание"

## Bibliography

bibliography: bib/cite.bib  
csl: pandoc/csl/gost-r-7-0-5-2008-numeric.csl

## Pdf output format

toc: true # Table of contents  
toc-depth: 2  
lof: true # List of figures  
lot: false # List of tables  
fontsize: 12pt  
linestretch: 1.5  
papersize: a4  
documentclass: scrreprt

## I18n polyglossia

polyglossia-lang:  
name: russian  
options:  
- spelling=modern  
- babelshorthands=true  
polyglossia-otherlangs:  
name: english

## I18n babel

babel-lang: russian  
babel-otherlangs: english

## Fonts

mainfont: PT Serif  
romanfont: PT Serif  
sansfont: PT Sans  
monofont: PT Mono

mainfontoptions: Ligatures=TeX  
romanfontoptions: Ligatures=TeX  
sansfontoptions: Ligatures=TeX,Scale=MatchLowercase  
monofontoptions: Scale=MatchLowercase,Scale=0.9

## Biblatex

biblatex: true  
biblio-style: "gost-numeric"  
biblatexoptions:

- parenttracker=true
- backend=biber
- hyperref=auto
- language=auto
- autolang=other\*
- citestyle=gost-numeric

## Pandoc-crossref LaTeX customization

figureTitle: "Рис."  
tableTitle: "Таблица"  
listingTitle: "Листинг"  
lofTitle: "Список иллюстраций"  
lotTitle: "Список таблиц"  
lolTitle: "Листинги"

## Misc options

indent: true  
header-includes:

- \usepackage{indentfirst}
- \usepackage{float} # keep figures where there are in the text
- \floatplacement{figure}{H} # keep figures where there are in the text

---

## Цель работы

**Приобрести практические навыки работы с языком Julia и OpenModelica, научиться строить графики и решать математические задачи.**

## Теоретическое введение

Julia — высокоуровневый свободный язык программирования с динамической типизацией, созданный для математических вычислений. Эффективен также и для написания программ общего назначения. Синтаксис языка схож с синтаксисом других

математических языков (например, MATLAB и Octave), однако имеет некоторые существенные отличия. Julia написан на Си, C++ и Scheme. Имеет встроенную поддержку многопоточности и распределённых вычислений, реализованные в том числе в стандартных конструкциях. [wiki:bash].

OpenModelica — свободное открытое программное обеспечение для моделирования, симуляции, оптимизации и анализа сложных динамических систем. Основано на языке Modelica. Активно развивается Open Source Modelica Consortium, некоммерческой неправительственной организацией. Open Source Modelica Consortium является совместным проектом RISE SICS East AB и Линчёпингского университета. По своим возможностям приближается к таким вычислительным средам как Matlab Simulink, Scilab xCos, имея при этом значительно более удобное представление системы уравнений исследуемого блока.

OpenModelica не поддерживает построение графиков в полярных координатах, поэтому невозможно написать лабораторную на этом языке. [wiki:bash]

---

## Выполнение лабораторной работы

### Подготовка

Так как задание выполняется по вариантам, сначала нужно рассчитать свой - мой вариант 27.

1. Установила язык Julia
2. Произвела математические вычисления:
  1. Введем полярные координаты.
  2. Траектория катера должна быть такой, чтобы и катер, и лодка все время были на одном расстоянии от полюса, только в этом случае траектория катера пересечется с траекторией лодки.
  3. Пусть через время  $t$  катер и лодка окажутся на одном расстоянии  $x$  от полюса. За это время лодка пройдет  $x$ , а катер  $k \cdot x$  (или  $k + x$ , в зависимости от начального положения катера относительно полюса). Отсюда у нас выходит два случая:

$$r_{0\_1} = 11.7 / (3.7 + 1) \text{ и } r_{0\_2} = 11.7 / (3.7 - 1)$$

4. После того, как катер береговой охраны окажется на одном расстоянии от полюса, что и лодка, он должен сменить прямолинейную траекторию и начать двигаться вокруг полюса удаляясь от него со скоростью лодки  $v$ .

2.5. Решение исходной задачи сводится к решению системы из двух дифференциальных уравнений, из которой выходит, что  $r = \exp(1.0)^{(t / \sqrt{12.69})}$

### Создание проекта (код на Julia)

```

using Plots
using DifferentialEquations

n = parse(Float64, "3.7")
s = parse(Float64, "11.7")

r0_1 = s/(n+1)
r0_2 = s/(n-1)

tspan = (0, 2*pi)

function f(u, p, t)
    u/exp(1.0)^(teta/sqrt(12.69))
end
prob = ODEProblem(f, r0_1, tspan)
sol = solve(prob, Tsit5(), reltol = 1e-8, abstol = 1e-8)
dxR = rand(1:size(sol.t)[1])
rAngles = [sol.t[dxR] for i in 1:size(sol.t)[1]]

plt1 = plot(proj=:polar, aspect_ratio=:equal, dpi = 1000, legend=true,
bg=:black)

plot!(plt1, xlabel="", ylabel="", title="Случай 1", legend=:outerbottom)
plot!(plt1, [rAngles[1], rAngles[2]], [0.0, sol.u[size(sol.u)[1]]],
label="Траектория браконьеров", color=:blue, lw=1)
scatter!(plt1, rAngles, sol.u, label="", mc=:blue, ms=0.0005)
plot!(plt1, sol.t, sol.u, xlabel="", ylabel="", label="Траектория охраны",
color=:pink, lw=1)
scatter!(plt1, sol.t, sol.u, label="", mc=:pink, ms=0.0005)

savefig(plt1, "try1.png")

prob = ODEProblem(f, r0_2, tspan)
sol = solve(prob, Tsit5(), reltol = 1e-8, abstol = 1e-8)
dxR = rand(1:size(sol.t)[1])
rAngles = [sol.t[dxR] for i in 1:size(sol.t)[1]]

plt1 = plot(proj=:polar, aspect_ratio=:equal, dpi = 1000, legend=true,
bg=:black)

plot!(plt1, xlabel="", ylabel="", title="Случай 2", legend=:outerbottom)
plot!(plt1, [rAngles[1], rAngles[2]], [0.0, sol.u[size(sol.u)[1]]],
label="Траектория браконьеров", color=:blue, lw=1)
scatter!(plt1, rAngles, sol.u, label="", mc=:blue, ms=0.0005)
plot!(plt1, sol.t, sol.u, xlabel="", ylabel="", label="Траектория охраны",
color=:pink, lw=1)
scatter!(plt1, sol.t, sol.u, label="", mc=:pink, ms=0.0005)

savefig(plt1, "try2.png")

```

Полученные графики(рис. @fig:001, @fig:002).



{ #fig:001 width=70%}

{ #fig:001 width=70%}

---

## Анализ результатов

Были построены два графика, на которых видно точки перенесения наших "кораблей".

## Вывод

В процессе выполнения данной лабораторной работы я приобрела практические навыки работы с языком Julia.

## Список литературы

[1] Документация по Julia: [https://ru.wikipedia.org/wiki/Julia\\_\(язык\\_программирования\)](https://ru.wikipedia.org/wiki/Julia_(язык_программирования))

[2] Документация по OpenModelica: <https://ru.wikipedia.org/wiki/OpenModelica>