

# Лабораторная работа №2

## Частный случай задачи о погоне

---

Ким М. А.

18 февраля 2023

Российский университет дружбы народов, Москва, Россия

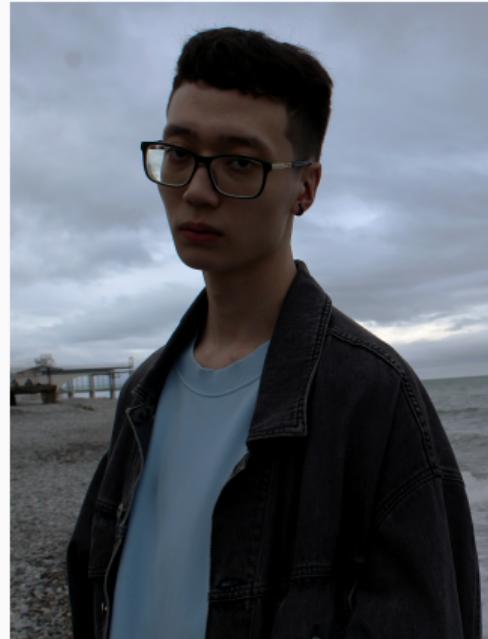
# Информация

---

# Докладчик

---

- Ким Михаил Алексеевич
- студент уч. группы НФИбд-01-20
- Российский университет дружбы народов
- 1032201664@pfur.ru
- <https://github.com/exmanka>



## Вводная часть

---

- Необходимость навыков моделирования реальных математических задач, построение графиков.

## Объект и предмет исследования

---

- Язык программирования Julia
- Язык моделирования Modelica
- Задача о погоне. Заяц-волк.

## Цели и задачи

---

- Ознакомиться с базовым функционалом языков Julia и OpenModelica.
- Смоделировать математическую задачу о погоне с помощью данных языков.

## Материалы и методы

---

- Языки программирования:
  - Julia
  - OpenModelica

## Процесс выполнения работы

---

## Подготовка системы для работы

---

## Установка Julia, REPL, дополнительных библиотек

```
makim@makim:~$ wget https://julialang-s3.julialang.org/bin/linux/x64/1.8/julia-1.8.5-linux-x86_64.tar.gz
```

```
makim@makim:~$ tar julia-1.8.5-linux-x86_64.tar.gz
```

```
makim@makim:~$ sudo cp -r julia-1.8.5 /opt/
```

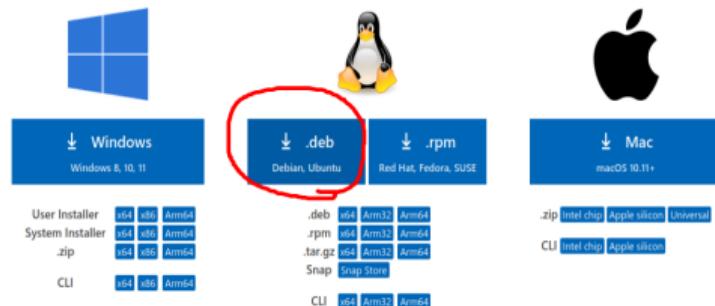
```
makim@makim:~$ sudo ln -s /opt/julia-1.8.5/bin/julia /usr/local/bin/julia
```

(@v1.8) `pkg> add DifferentialEquations` (@v1.8) `pkg>`

## Установка Visual Studio Code

## Download Visual Studio Code

Free and built on open source. Integrated Git, debugging and extensions.

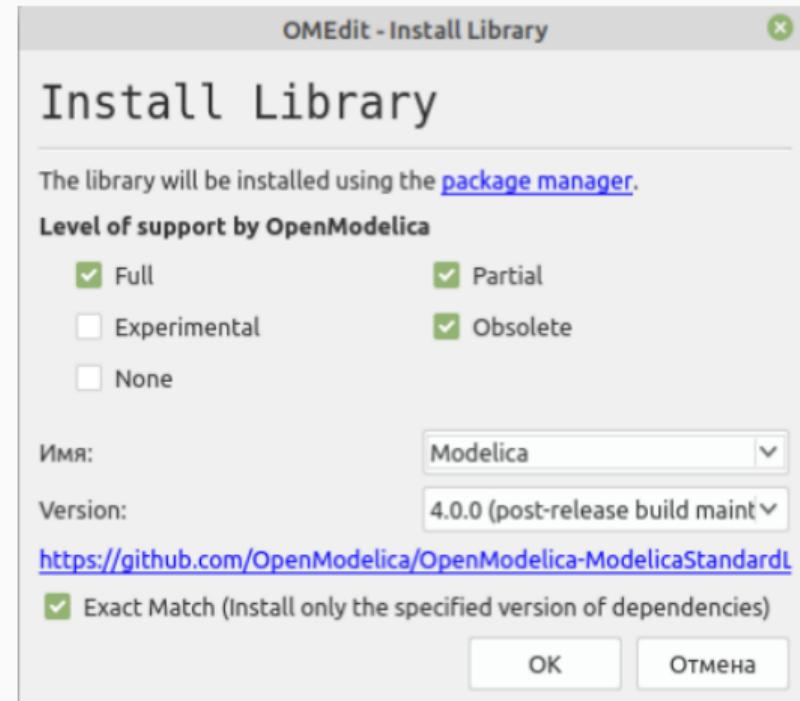


```
makim@makim:~/media/st/_lab_current$ cd ~/Загрузки
makim@makim:~/Загрузки$ sudo apt install ./code_1.75.1-1675893397_amd64.deb
[sudo] пароль для makim:
Чтение списков пакетов... Готово
Построение дерева зависимостей
Чтение информации о состоянии... Готово
Заметьте, вместо «./code_1.75.1-1675893397_amd64.deb» выбирается «code»
Следующие пакеты устанавливались автоматически и больше не требуются:
  linux-headers-5.4.0-89 linux-headers-5.4.0-89-generic linux-image-5.4.0-89
  linux-modules-extra-5.4.0-89-generic
Для их удаления используйте «sudo apt autoremove».
Следующие НОВЫЕ пакеты будут установлены:
  code
```

$$2\theta - \alpha + \text{Tab} = \alpha$$

# Установка Modelica, OpenModelica, дополнительных библиотек

```
makim@makim:~$ for deb in deb deb-src; do echo "$deb http://build.openmodelica.org/apt `lsb_release -cs` release"; done | sudo tee /etc/apt/sources.list.d/openmodelica.list
[sudo] пароль для makim:
deb http://build.openmodelica.org/apt uma release
deb-src http://build.openmodelica.org/apt uma release
makim@makim:~$ wget -q http://build.openmodelica.org/apt/openmodelica.asc -O- | sudo apt-key add -
OK
makim@makim:~$ sudo apt update
Сущ:1 http://mirror.docker.ru/ubuntu focal InRelease
Сущ:2 http://mirror.docker.ru/ubuntu focal-updates InRelease
Сущ:3 http://mirror.docker.ru/ubuntu focal-backports InRelease
makim@makim:~$ sudo apt install openmodelica
Чтение списков пакетов... Готово
Построение дерева зависимостей
Чтение информации о состоянии... Готово
Следующие пакеты устанавливались автоматически и больше
 linux-headers-5.4.0-89 linux-headers-5.4.0-89-generic
 linux-image-5.4.0-89-generic linux-modules-5.4.0-89-9
 linux-modules-extra-5.4.0-89-generic
Для их удаления используйте «sudo apt autoremove».
Будут установлены следующие дополнительные пакеты:
 aglfn binfmt-support clang clang-10 drcontrol drmodel
makim@makim:~$ /usr/bin/omc --version
OpenModelica 1.20.0
```



## Формулировка задания

---

## Формулировка задания

*На море в тумане катер береговой охраны преследует лодку браконьеров. Через определенный промежуток времени туман рассеивается, и лодка обнаруживается на расстоянии 6,4 км от катера. Затем лодка снова скрывается в тумане и уходит прямолинейно в неизвестном направлении. Известно, что скорость катера в 2,5 раза больше скорости браконьерской лодки:  $V_O = 2.5V_B$ .*

Браконьеры и охотники видят друг друга на расстоянии  $a = 6.2$  до наступления тумана.

Браконьеры двигаются по прямой в то время, как охотники движутся по спирали, чтобы нагнать браконьеров в любом случае, поскольку вторые движутся в неизвестном направлении.

## Математические вычисления

---

# Математические вычисления

$$t = \frac{r_0}{\sqrt{3}} = \frac{a - r_0}{\sqrt{3}r_0} = \frac{a - r_0}{2,5\sqrt{3}}$$

$$2,5r_0 = a - r_0$$

$$3,5r_0 = a$$

$$r_0 = \frac{a}{3,5}$$

$$r_0 = \frac{6,2}{2,5}$$

$$\sqrt{r_0^2} = \sqrt{r_1^2} + \sqrt{r_2^2}$$

$$(2,5\sqrt{3})^2 = \sqrt{r_3^2} + \sqrt{r_4^2}$$

$$\sqrt{r_1^2} = 6,25\sqrt{3}^2 - \sqrt{r_3^2}$$

$$\sqrt{r_2^2} = 5,25\sqrt{3}^2$$

$$\sqrt{r_3^2} = \sqrt{5,25}\sqrt{3}$$

$$\begin{cases} \frac{dr}{dt} = \sqrt{3} \\ r \frac{d\theta}{dt} = \sqrt{5,25}\sqrt{3} \end{cases} \quad \begin{cases} dt = \frac{dr}{\sqrt{3}} \\ dt = \frac{r d\theta}{\sqrt{5,25}\sqrt{3}} \end{cases}$$

$$\frac{dr}{\sqrt{3}} = \frac{r d\theta}{\sqrt{5,25}\sqrt{3}} \Rightarrow \frac{dr}{d\theta} = \frac{r}{\sqrt{5,25}}$$

$$t = \frac{r_0}{\sqrt{3}} = \frac{r_1 - a}{\sqrt{3}r_0} = \frac{r_1 - a}{2,5\sqrt{3}} \quad \frac{x - r}{k - a}$$

$$2,5r_1 = r_1 - a$$

$$r_1 = \frac{r_1 - a}{2,5}$$

$$\frac{x}{x} = \frac{k - x}{2,5x} \quad \frac{x}{x} = \frac{x + k}{2,5x}$$

$$2,5x = k - x \quad 2,5x = x + k$$

$$3,5x = k \quad 1,5x = k$$

$$3,5r_1 = a \quad 1,5r_1 = a$$

$$r_0 = \frac{a}{3,5} \approx 1,77 \quad r_1 = \frac{a}{1,5} \approx 4,13$$

## Написание программ

---

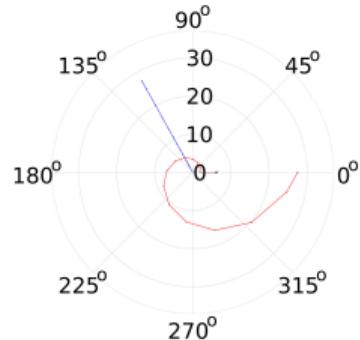
# Код на Julia

```
1 # Дополнительная информация
2 # dr(r) = r / sqrt(3)          # Короткая запись функции, sqrt(3) ~ \sqrt{3} + TAB 3
3 # 0 = range(0, 2\pi, 1000)      # 1000 значений от 0 до 2\pi
4 # @show \phi                  # Макрос, позволяющий выводить не только значение, но и имя переменной.
5 # rfloat = rand(0.0:0.0001:2\pi) # Случайное Float64 значение от 0.0 до 2\pi с шагом 0.0001
6 #
7
8 # Подключение библиотек
9 using Plots
10 using DifferentialEquations
11
12
13 const n = 2.5
14
15
16 "Правая часть ОДУ: u - переменная (скаляр или массив), r - параметры (кортеж, tuple), t - аргумент (скалар, время)."
17 function F(u, p, t)
18     # аргументы p, t нужны для работы метода
19     return u / \sqrt{n * n - 1)
20 end
21
22 "Функция решения задачи, a - начальное расстояние катера от лодки, n - отношение скорости катера к лодки"
23 function the_chase_curve(a, n, flag)
24     "Расстояние, на котором катер начнет описывать спираль"
25     r = 0.0
26     "Интервал спирал"
27     T = (0.0, 0.0)
28
29     if flag == 0
30         r = a / (n + 1)
31         T = (0, 2\pi)
32     elseif flag == 1
33         r = -(a / (n - 1))
34         T = (-n, n)
35     else
36         println("Неправильно выбран флаг")
37     end
38
39     # Задача
40     prob = ODEProblem(F, abs(r), T)
41
42     # Решение задачи
43     sol = solve(
44         prob,
45         dtmax=0.5
46     )
47
48
```

```
49     # Задание пустого пространства
50     plt = plot(
51         proj=:polar,
52         ylims=(0.0, round(wbs(sol.u[size(sol.u[1])])) + 10.0),
53         aspect_ratio=:equal,          # Размер одного деления по осям всегда одинаков
54         dpi=600,
55         legend=true,                # Отображение легенды
56         # bg=:black
57     )
58
59     @show sol.t
60     @show sol.u
61
62     r_idx = rand(1:size(sol.t)[1])
63     r_angle = sol.t[r_idx]
64     # r_angle_list = [sol.t[n] for n in 1:size(sol.t)[1]]
65
66     plot!(plt, xlabel="r", ylabel="r(t)", title="Кривая погоне. Частный случай", legend=:outerbottom)
67     savefig(plt, "lab2_" * string(flag) * ".png")
68
69     plot!(plt, [0.0, 0.0], [a, r], label="Начальное катера движение по прямой", color=:brown, lw=0.4)
70     scatter!(plt, [0.0], [a], label=" ", mc=:brown, ms=0.4)
71     plot!(plt, [r_angle, r_angle], [0.0, sol.u[1]], label="траектория лодки", color=:blue, lw=0.4)
72     scatter!(plt, [r_angle], [sol.u[1]], label=" ", mc=:blue, ms=0.01)
73     plot!(plt, [sol.t[1], sol.t[1]], [sol.u[1], sol.u[1]], label="траектория катера", color=:red, lw=0.4)
74     savefig(plt, "lab2_" * string(flag) * ".png")
75
76     for i in 2:size(sol.t)[1]
77         # Добавление параметров в пространство
78         plot!(plt, [r_angle, r_angle], [sol.u[i-1], sol.u[i]], label=" ", color=:blue, lw=0.4)
79         scatter!(plt, [r_angle], [sol.u[i]], label=" ", mc=:blue, ms=0.01)
80
81         plot!(plt, [sol.t[i-1], sol.t[i]], [sol.u[i-1], sol.u[i]], label=" ", color=:red, lw=0.4)
82         scatter!(plt, [sol.t[i]], [sol.u[i]], label=" ", mc=:red, ms=0.01)
83
84         savefig(plt, "lab2_" * string(flag) * "_" * string(i))
85     end
86
87     return 0
88 end
89
90
91 the_chase_curve(6.2, n, 1)
92
```

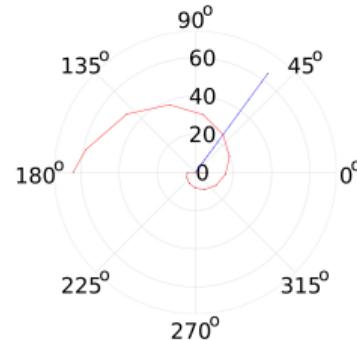
# Результаты выполнения программы на Julia

Кривая погони. Частный случай



- Начальное движение по прямой
- Траектория лодки
- Траектория катера

Кривая погони. Частный случай



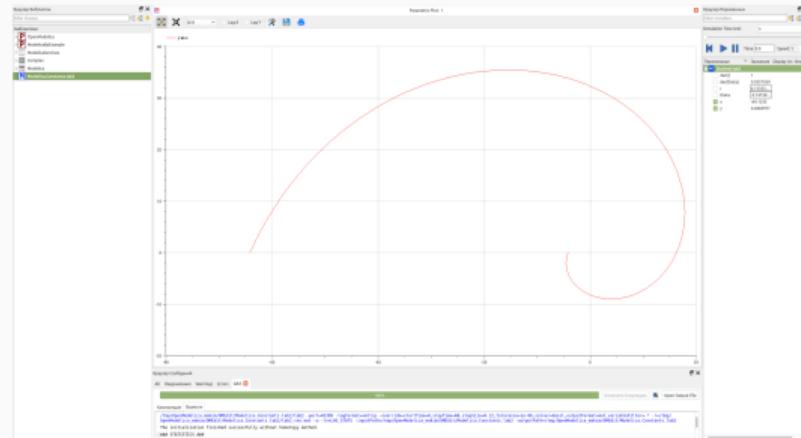
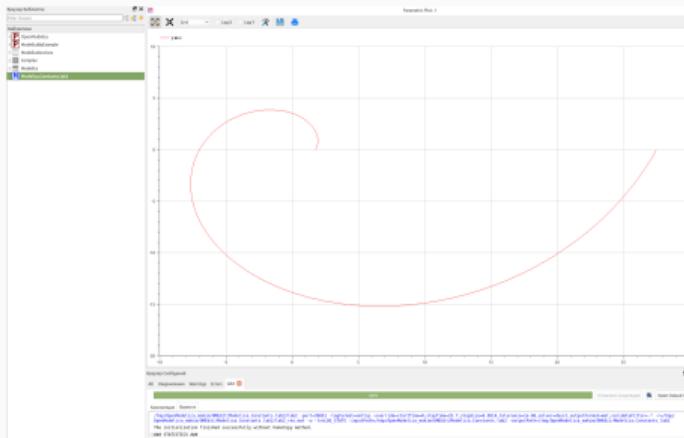
- Начальное движение по прямой
- Траектория лодки
- Траектория катера

# Код на Modelica

```
1 within Modelica.Constants;
2 model lab2
3  constant Real n = 2.5;
4  constant Real a = 6.2;
5  constant Real r1 = a / (n + 1);
6  constant Real r2 = a / (n - 1);
7  Real theta(start=0);
8  Real r(start=r1);
9  Real x;
10 Real y;
11 equation
12  der(r) = 1;
13  der(r)/der(theta) = r / sqrt(5.25);
14  x = r * cos(theta);
15  y = r * sin(theta);
16 annotation(experiment(StartTime=0,StopTime=25.7));
17 //annotation(experiment(StartTime=0,StopTime=60));
18 end lab2;
```

```
1 within Modelica.Constants;
2 model lab2
3  constant Real n = 2.5;
4  constant Real a = 6.2;
5  constant Real r1 = a / (n + 1);
6  constant Real r2 = a / (n - 1);
7  Real theta(start=-pi);
8  Real r(start=r2);
9  Real x;
10 Real y;
11 equation
12  der(r) = 1;
13  der(r)/der(theta) = r / sqrt(5.25);
14  x = r * cos(theta);
15  y = r * sin(theta);
16 //annotation(experiment(StartTime=0,StopTime=25.7));
17 annotation(experiment(StartTime=0,StopTime=60));
18 end lab2;
```

# Результаты выполнения программы на Modelica



## Результаты

---

## Результаты

---

- Написаны программы на языках Julia и Modelica, моделирующие задачу о погоне Заяц-Волк в двух вариантах
- Построены графики движения в соответствии с поставленными задачами

## Вывод

Ознакомился с базовым функционалом языка программирования Julia и языка моделирования Modelica, а также с функционалом программного обеспечения OpenModelica. Используя эти средства, построил математическую модель, представляющую собой частный случай задачи о погоне.