

# Отчёт по лабораторной работе №6

---

Быстров Г. А.

18 марта 2023

Российский университет дружбы народов, Москва, Россия

- узнать как работать с математическими моделями;
- решить возникающие трудности и проблемы;
- практически получить полезный результат.

В данной лабораторной работе мне будет необходимо изучить построение математических моделей и рассмотреть простейшую модель эпидемии.

1. Сделаем программную реализацию на языке OpenModelica для первого случая (рис. 1).

```
1 model Lab6_1var
2   constant Integer N = 10060;
3   constant Integer I_0 = 61;
4   constant Integer R_0 = 23;
5   constant Integer S_0 = N-I_0-R_0;
6   constant Real alpha = 0.01;
7   constant Real beta = 0.02;
8   Real i(start=I_0);
9   Real r(start=R_0);
10  Real s(start=S_0);
11  Real t = time;
12  equation
13    der(i) = -beta*i;
14    der(r) = beta*i;
15    der(s) = 0;
16    annotation(experiment(StartTime = 0, StopTime = 90.0));
17 end Lab6_1var;
```

Рис. 1: Код программы на OpenModelica для первого случая

2. График изменения числа людей в каждой из трех групп для первого случая (рис. 2).

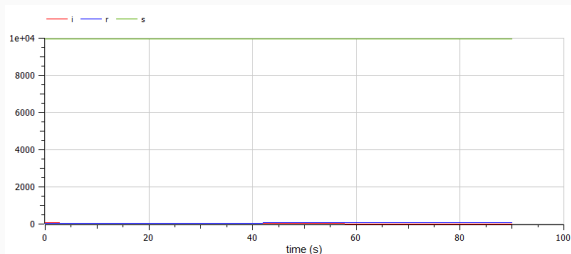


Рис. 2: График для первого случая (OpenModelica)

## 3. Сделаем программную реализацию на языке Julia (рис. 3).

```
1 using DifferentialEquations
2 using Plots
3
4 function F!(du, u, p, t)
5     du[1] = - beta * u[2]
6     du[2] = beta * u[1]
7     du[3] = 0
8 end
9
10 N = 10000
11 I_0 = 61
12 R_0 = 23
13 S_0 = N - I_0 - R_0
14 alpha = 0.01
15 beta = 0.02
16 T = (0.0, 90.0)
17 u_0 = [I_0, R_0, S_0]
18
19 prob = ODEProblem(F!, u_0, T)
20
21 sol = solve(
22     prob,
23     saveat = 0.1
24 )
25
26 const I = Float64[]
27 const R = Float64[]
28 const S = Float64[]
29
30 @show sol.u
31 @show sol.t
32
```

Рис. 3: Код программы на Julia

4. Сделаем программную реализацию на языке Julia (рис. 4).

```
33 for u in sol.u
34     I, r, s = u
35     push!(I, I)
36     push!(r, r)
37     push!(s, s)
38 end
39
40 plt = plot(
41     size = (700, 500),
42     dpi = 500,
43     legend = true,
44     title = "График для первого случая"
45 )
46
47 plot!(
48     plt,
49     sol.t,
50     I,
51     color=:green,
52     xlabel="Время",
53     ylabel="Численность",
54     label="Инфицированные"
55 )
56
57 plot!(
58     plt,
59     sol.t,
60     R,
61     color=:blue,
62     xlabel="Время",
63     ylabel="Численность",
64     label="Восприимчивые"
```

Рис. 4: Код программы на Julia

5. Сделаем программную реализацию на языке Julia (рис. 5).

```
65     }  
66  
67     plot(  
68         plt,  
69         sol.t,  
70         S,  
71         color=:pink,  
72         xlabel="Время",  
73         ylabel="Численность",  
74         label="Восприимчивость"  
75     )  
76  
77     savefig(plt, "Labo_ver1_Bystrovgleb")
```

Рис. 5: Код программы на Julia



6. График изменения числа людей в каждой из трех групп для первого случая (рис. 6).

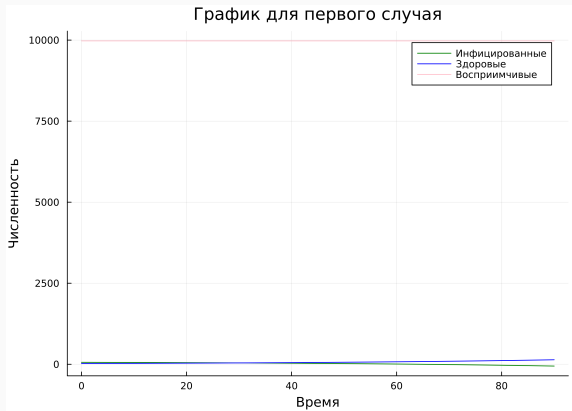


Рис. 6: График для первого случая (Julia)

7. Сделаем программную реализацию на языке OpenModelica для второго случая (рис. 7).

```
1  model Lab6_2var
2      constant Integer N = 10060;
3      constant Integer I_0 = 61;
4      constant Integer R_0 = 23;
5      constant Integer S_0 = N-I_0-R_0;
6      constant Real alpha = 0.01;
7      constant Real beta = 0.02;
8      Real i(start=I_0);
9      Real r(start=R_0);
10     Real s(start=S_0);
11     Real t = time;
12 equation
13     der(i)= alpha*s-beta*i;
14     der(r)= beta*i;
15     der(s)= -alpha*s;
16     annotation(experiment(StartTime = 0, StopTime = 90.0));
17 end Lab6_2var;
```

Рис. 7: Код программы на OpenModelica для второго случая

8. График изменения числа людей в каждой из трех групп для второго случая (рис. 8).

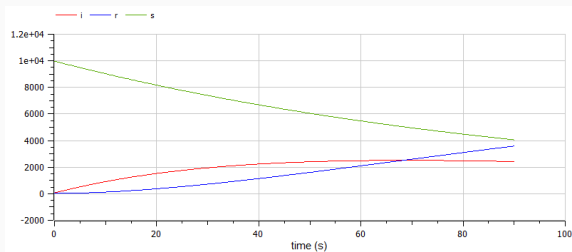


Рис. 8: График для второго случая (OpenModelica)

9. Сделаем программную реализацию на языке Julia (рис. 9).

```
1 using DifferentialEquations
2 using Plots
3
4 function f!(du, u, p, t)
5     du[1] = alpha * u[1] - beta * u[2]
6     du[2] = beta * u[2]
7     du[3] = - alpha * u[1]
8 end
9
10 N = 10000
11 I_0 = 0.1
12 R_0 = 23
13 S_0 = N - I_0 - R_0
14 alpha = 0.01
15 beta = 0.02
16 T = (0.0, 90.0)
17 u_0 = [I_0, R_0, S_0]
18
19 prob = ODEProblem(f!, u_0, T)
20
21 sol = solve(
22     prob,
23     saveat = 0.1
24 )
25
26 const I = Float64[]
27 const R = Float64[]
28 const S = Float64[]
29
30 @show sol.u
31 @show sol.t
32
```

Рис. 9: Код программы на Julia

10. Сделаем программную реализацию на языке Julia (рис. 10).

```
32 for u in sol.u
33     l, r, s = u
34     push!(l, l)
35     push!(R, r)
36     push!(s, s)
37 end
38
39
40 plt = plot(
41     size = (700, 500),
42     dpi = 500,
43     legend = true,
44     title = "График для второго случая"
45 )
46
47 plot!(
48     plt,
49     sol.t,
50     l,
51     color=:green,
52     xlabel="Время",
53     ylabel="Численность",
54     label="Инфицированные"
55 )
56
57 plot!(
58     plt,
59     sol.t,
60     R,
61     color=:blue,
62     xlabel="Время",
63     ylabel="Численность",
64     label="Вакцины"
```

Рис. 10: Код программы на Julia

11. Сделаем программную реализацию на языке Julia (рис. 11).

```
65     )  
66  
67     plot(  
68         plt,  
69         sol.t,  
70         S,  
71         color=:pink,  
72         xlabel="Время",  
73         ylabel="Масса",  
74         label="Восприимчивость"  
75     )  
76  
77     savefig(plt, "Lab0_ver2_BystrovGleb")  
78
```

Рис. 11: Код программы на Julia

12. График изменения числа людей в каждой из трех групп для второго случая (рис. 12).

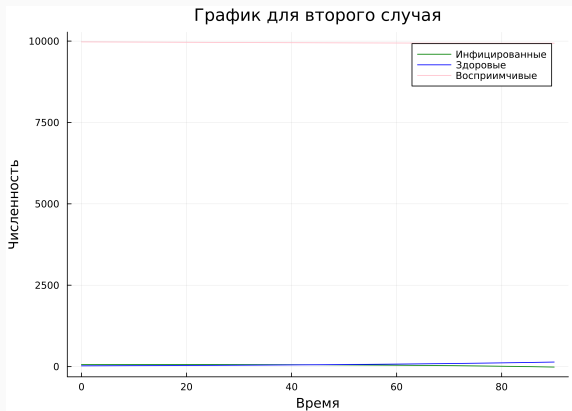


Рис. 12: График для второго случая (Julia)

## 13. Загрузил работу на GitHub (рис. 13).

A screenshot of a GitHub repository's file list. The interface is dark-themed. It shows a list of files and folders with their respective commit messages. The files include configuration files, a course structure, and various README files.

config	Initial commit
labs	feat(main): make course structure
template	Initial commit
.gitattributes	Initial commit
.gitignore	Initial commit
.gitmodules	Initial commit
CHANGELOG.md	Initial commit
COURSE	Initial commit
LICENSE	Initial commit
Makefile	Initial commit
README.en.md	Initial commit
README.git-flow.md	Initial commit
README.md	Initial commit

Рис. 13: Страница репозитория на сайте



- узнал как работать с математическими моделями;
- создал математическую модель согласно заданию.