

# Лабораторная работа №6

Задача об эпидемии

---

Шубнякова Дарья НКНбд-01-22

1. Вводная часть

2. Основная часть

3. Результаты

# **1. Вводная часть**

---

## 1.1 Цели и задачи

Ознакомиться с задачей об эпидемии. Реализовать данную модель на языке Julia, а так же в среде OMEdit.

Построить два графика с динамикой эпидемии и относительными долями населения.

## 2. Основная часть

---

## 2.1 Выполнение лабораторной работы

### 2.1.1 Пишем код на языке Julia для реализации данной модели.

```
[2]: using DifferentialEquations, Plots

# Параметры модели для варианта 13
N = 19000.0      # Общая численность населения
beta = 0.3       # Коэффициент заражения
gamma = 0.1      # Коэффициент выздоровления

# Начальные условия
S0 = N - 119 - 19 # Восприимчивые: 19000 - 119 - 19 = 18862
I0 = 119.0        # Инфицированные
R0 = 19.0         # Выздоровевшие с иммунитетом

u0 = [S0, I0, R0] # Начальные условия [S, I, R]
tspan = (0.0, 200.0) # Временной интервал

# Система уравнений SIR модели
function sir_model!(du, u, p, t)
    S, I, R = u
    beta, gamma, N = p

    du[1] = -beta * S * I / N      # dS/dt
    du[2] = beta * S * I / N - gamma * I # dI/dt
    du[3] = gamma * I              # dR/dt
end

# Параметры модели
p = (beta, gamma, N)

# Решение системы
prob = ODEProblem(sir_model!, u0, tspan, p)
sol = solve(prob, Tsit5(), reltol=1e-6, abstol=1e-6)
```

## 2.2 Продолжение кода.

```
# График 1: Изменение численности групп
plot!(plt[1], sol.t, sol[1,:], label="Восприимчивые (S)", linewidth=2, color=:blue)
plot!(plt[1], sol.t, sol[2,:], label="Инфицированные (I)", linewidth=2, color=:red)
plot!(plt[1], sol.t, sol[3,:], label="Выздоровевшие (R)", linewidth=2, color=:green)
plot!(plt[1], title="Динамика эпидемии (SIR модель)", xlabel="Время", ylabel="Численно

# График 2: Относительные доли
plot!(plt[2], sol.t, sol[1,:]/N, label="S/N", linewidth=2, color=:blue)
plot!(plt[2], sol.t, sol[2,:]/N, label="I/N", linewidth=2, color=:red)
plot!(plt[2], sol.t, sol[3,:]/N, label="R/N", linewidth=2, color=:green)
plot!(plt[2], title="Относительные доли населения", xlabel="Время", ylabel="Доля", leg

# Анализ порогового значения
R0_effective = beta / gamma # Базовое репродуктивное число
I_critical = N * (gamma / beta) # Критическое число инфицированных

println("Анализ эпидемии:")
println("Общая численность: N = ", N)
println("Начальные условия: S₀ = ", S0, ", I₀ = ", I0, ", R₀ = ", R0)
println("Базовое репродуктивное число: R₀ = ", round(R0_effective, digits=2))
println("Критическое число инфицированных: I* = ", round(I_critical, digits=2))

# Проверка условий
if I0 <= I_critical
    println("Случай 1: I₀ ≤ I* – эпидемия будет слабой")
else
    println("Случай 2: I₀ > I* – ожидается сильная эпидемия")
end

# Отображение графиков
display(plt)

# Сохранение результатов
savefig("epidemic_dynamics.png")
println("\nГрафик сохранен в файл: epidemic_dynamics.png")
```

## 2.3 Получаем два графика, необходимых для выполнения задания.

Анализ эпидемии:

Общая численность:  $N = 19000.0$

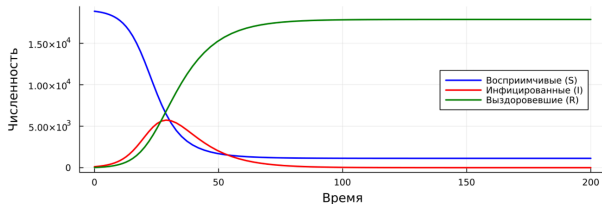
Начальные условия:  $S_0 = 18862.0$ ,  $I_0 = 119.0$ ,  $R_0 = 19.0$

Базовое репродуктивное число:  $R_0 = 3.0$

Критическое число инфицированных:  $I^* = 6333.33$

Случай 1:  $I_0 \leq I^*$  – эпидемия будет слабой

Динамика эпидемии (SIR модель)



Относительные доли населения

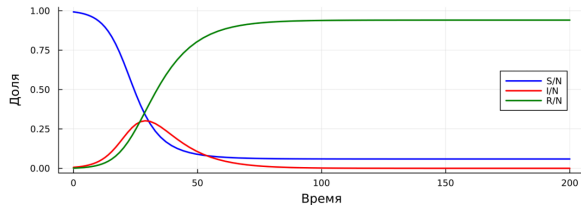


График сохранен в файл: epidemic\_dynamics.png



## 2.4 Прописываем код в среде OMEdit.

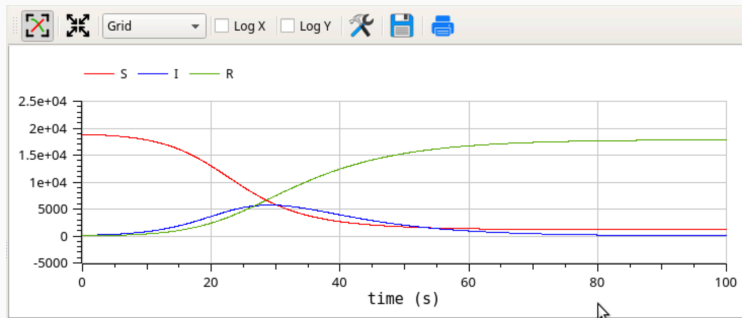
```
model EpidemicModel "Модель эпидемии SIR для варианта 13"  
  parameter Real N = 19000 "Общая численность населения";  
  parameter Real beta = 0.3 "Коэффициент заражения";  
  parameter Real gamma = 0.1 "Коэффициент выздоровления";  
  
  parameter Real S0 = N - 119 - 19 "Начальное число восприимчивых";  
  parameter Real I0 = 119 "Начальное число инфицированных";  
  parameter Real R0 = 19 "Начальное число выздоровевших";  
  
  Real S(start = S0, fixed = true) "Восприимчивые";  
  Real I(start = I0, fixed = true) "Инфицированные";  
  Real R(start = R0, fixed = true) "Выздоровевшие";  
  Real t "Время";
```

I

## 2.5 Продолжение кода.

```
15 equation
16 // Система уравнений SIR модели
17 der(S) = -beta * S * I / N;
18 der(I) = beta * S * I / N - gamma * I;
19 der(R) = gamma * I;
20
21 der(t) = 1;
22
23 annotation(experiment(StartTime=0, StopTime=200, Tolerance=1e-6, Interval=0.1));
24 end EpidemicModel;
```

## 2.6 Получаем такой график.



### 3. Результаты

---

### 3. Результаты

Мы реализовали модель в OpenModelica и на языке Julia. На выходе получили картинку: epidemic\_dynamics.png. Итоговый файл lab6.ibybn с кодом на языке Julia в JupiterNotebook. А также файл для симуляции в OpenModelica: EpidemicModel.mo