



# Desenvolvimento de um software científico para a modelagem e simulação computacional

Dissertação de Mestrado

Mestrado em Ciência da Computação

**Brenno Lemos Melquíades dos Santos**    **Orientador: Prof. Alexandre B. Pigozzo**

19 de março de 2024





# Sumário

## 1 Introdução

- ▶ **Introdução**
- ▶ Referencial teórico
- ▶ Software para modelagem e simulação
  - Interface Gráfica
  - Representação Intermediária
- ▶ Aplicações do software
- ▶ Conclusões e trabalhos futuros



# Introdução

## 1 Introdução



# Sumário

## 2 Referencial teórico

- ▶ Introdução
- ▶ **Referencial teórico**
- ▶ Software para modelagem e simulação
  - Interface Gráfica
  - Representação Intermediária
- ▶ Aplicações do software
- ▶ Conclusões e trabalhos futuros

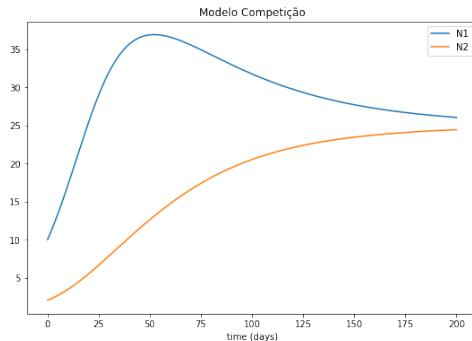
# Equações Diferenciais Ordinárias (EDOs)

## 2 Referencial teórico

- Usadas para estudar o comportamento populacional ao longo do tempo;
- Diversas aplicações em várias áreas do conhecimento;
- Cada equação descreve a concentração de uma população diferente;

$$\frac{dN_1}{dt} = r_1 \cdot N_1 (1 - W_{11} \cdot N_1 - W_{21} \cdot N_2)$$

$$\frac{dN_2}{dt} = r_2 \cdot N_2 (1 - W_{22} \cdot N_2 - W_{12} \cdot N_1) \quad (1)$$



Um modelo clássico da literatura é o modelo Predador-Presa. Este modelo descreve o comportamento de duas populações,  $H$  e  $P$ , que possuem uma relação de predação entre si.

Na equação, temos que

$\frac{dH}{dt} = r.H - a.H.P$	$H$	Presa
	$P$	Predador
(2)	$r$	Taxa de reprodução da presa
$\frac{dP}{dt} = b.H.P - m.P$	$m$	Taxa de mortalidade dos predadores
	$a$	Taxa de predação
	$b$	Taxa de reprodução dos predadores



# Sumário

## 3 Software para modelagem e simulação

- ▶ Introdução
- ▶ Referencial teórico
- ▶ **Software para modelagem e simulação**
  - Interface Gráfica
  - Representação Intermediária
- ▶ Aplicações do software
- ▶ Conclusões e trabalhos futuros

# Fluxograma de uso simplificado

3 Software para modelagem e simulação

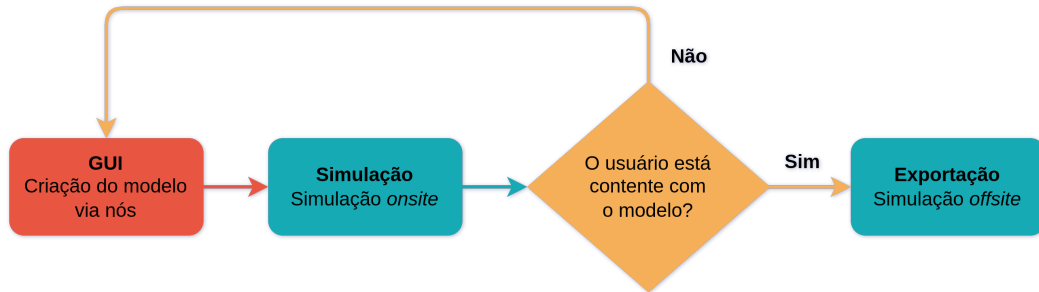


Figura 1: Fluxograma da experiência do usuário.



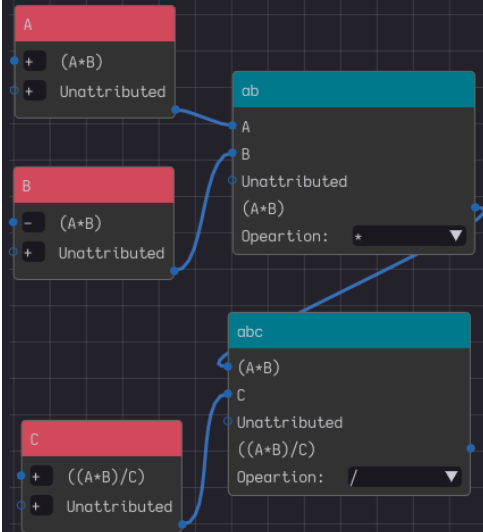
O software deve entregar as seguintes funcionalidades:

- Criação de modelos pela interface gráfica;
- Simulação do modelo e exibição dos resultados na interface;
- Exportação de PDF/Imagens com os resultados das simulações;
- Exportação de código equivalente ao modelo implementado;

# Representação do Modelo

## 3 Software para modelagem e simulação

- O software necessita de uma interface simples de ser usada;
  - Mas também deve naturalmente relembrar uma EDO;
- Após diversas iterações, chegamos numa interface baseada em programação visual;
  - Estas interfaces existem desde 1963, mas tiveram uma renascença com o avanço dos computadores e a necessidade de softwares de edição de imagem/áudio/vídeo;



# Fluxo de passagem de informações na interface

## 3 Software para modelagem e simulação

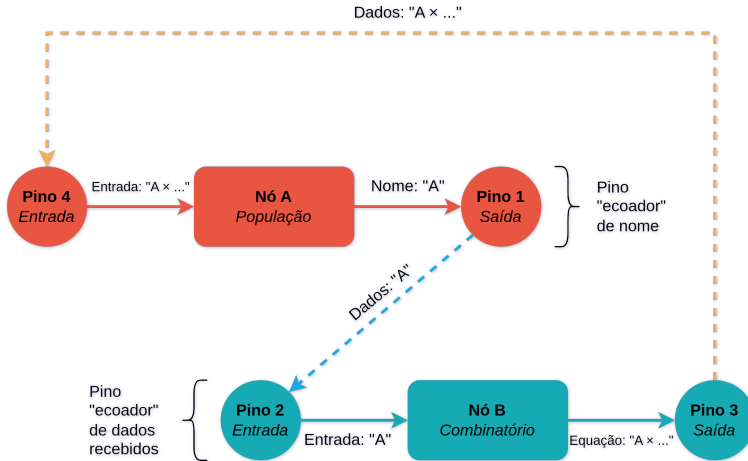


Figura 2: Relações entre nós e pinos.

## GRail — 1968

3 Software para modelagem e simulação





# Representação Intermediária

3 Software para modelagem e simulação

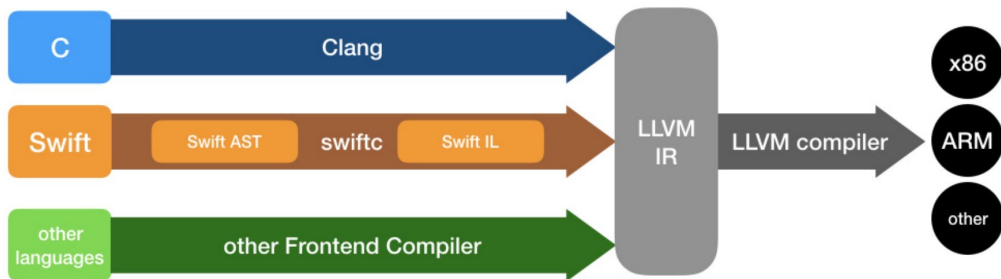


Universidade Federal  
de São João del-Rei

## Representação Intermediária

### 3 Software para modelagem e simulação

- Com o objetivo de realizar tantas transformações, torna-se necessário a utilização de uma Representação Intermediária (RI);
- Inspirados nas arquiteturas de compiladores modernos (GCC, baseados em LLVM), separamos a estrutura em back-end e front-end;
- Essa abordagem garante o desacoplamento entre estrutura e produtos finais;





## RI — Serde: Conversões automatizadas para JSON

3 Software para modelagem e simulação

### Rust

```
#[derive(Serialize, Deserialize)]
struct Person {
    name: String,
    age: u8,
    phones: Vec<String>,
    address: Address,
}
#[derive(Serialize, Deserialize)]
struct Address {
    street: String,
    city: String,
}
```

15/21

### JSON

```
{
  "name": "John Doe",
  "age": 43,
  "address": {
    "street": "1st St.",
    "city": "London"
  },
  "phones": [
    "+44 1234567",
    "+44 2345678"
  ]
}
```

- Structs *serializáveis* podem ser usadas diretamente em *templates*;
- Suponha a variável `people`: `Vec<Person>`:

#### minijinja

```
{% for person in people %}
    {{ person.name }}, {{ person.age }} anos.
    Contato: {% for phone_nb in person.phones -%}
        {{ phone_nb }}
        {%- if not loop.last %}, {% endif -%}
    {% endfor %}
{% endfor %}
```





# Sumário

## 4 Aplicações do software

- ▶ Introdução
- ▶ Referencial teórico
- ▶ Software para modelagem e simulação
  - Interface Gráfica
  - Representação Intermediária
- ▶ **Aplicações do software**
- ▶ Conclusões e trabalhos futuros



# Aplicação do software

4 Aplicações do software



# Sumário

## 5 Conclusões e trabalhos futuros

- ▶ Introdução
- ▶ Referencial teórico
- ▶ Software para modelagem e simulação
  - Interface Gráfica
  - Representação Intermediária
- ▶ Aplicações do software
- ▶ **Conclusões e trabalhos futuros**



# Conclusões e trabalhos futuros

## 5 Conclusões e trabalhos futuros

# Desenvolvimento de um software científico para a modelagem e simulação computacional