

# Computação Escalável

## Trabalho 1: Processamento de Dados de um Simulador de Rodovias

### Arquivos e Pastas

```
COMPUTACAO_ESCALAVEL_A1
|
├── etl
|   |   car_classes.cpp           → Classes auxiliares do ETL
|   |   dashboard_class.cpp       → Classe Dashboard
|   |   dashboard_with_threads.cpp → Dashboard com Threads
|   |   etl.cpp                   → ETL
|   |   external_service.cpp      → Serviço Externo
|
├── report
|   |   report.pdf                → Relatório do trabalho
|   |   report.tex                → Arquivo fonte do relatório
|   |   ***(arquivos auxiliares do LaTeX)
|
├── mock.py                       → Simulador de Rodovias
|   |   ansi.py                   → Classe de cores
|   |   mock.py                   → Scrip do Simulador
|   |   parameters.py             → Parâmetros do simulador
|
├── dashboard_test.cpp             → Teste do Dashboard
|
├── world_creator.py               → Criação do mundo
```

### Execução

A sequência de execução do trabalho é a seguinte: criação do mundo, execução do simulador e execução do ETL. Há uma forma simplificada de fazer tudo isso. Basta executar o script `run.py`, localizado no diretório raiz do projeto:

```
python run.py
```

Contudo, caso essa forma não funcione, é possível executar cada parte do trabalho separadamente.

Para executar o simulador, basta estar no diretório raiz do projeto e executar o script de criação do mundo:

```
python world_creator.py
```

Após a criação do mundo, basta executar o simulador:

```
python mock/mock.py
```

Por fim, a fim de executar o etl, é preciso estar dentro do diretório etl

```
cd etl  
g++ etl.cpp -o etl.exe  
./etl.exe
```

## Avaliação

- Modelagem geral do sistema.
- Identificação e solução dos problemas de concorrência.
- Relatório contendo a modelagem, os problemas/soluções e demais decisões de projeto.
- Eficiência do ETL considerando as características de paralelismo e concorrência.
- Solução para o acesso ao serviço externo legado.
- Solução para a diferença de prioridade entre análises.
- Projeto e implementação dos mocks.
- Organização do projeto e qualidade do código.
- Manual de instruções.

## Relatório

- ☒ Modelagem do Trabalho
- ☒ Mock
- ☒ ETL
- ☒ Dashboard
- ☒ Relatório
- ☒ Problemas e Soluções
- ☒ Conclusão

## Integrantes

- Breno Marques Azevedo
- Bruno Pereira Fornaro
- Luis Fernando Laguardia
- Vanessa Berwanger Wille

- Vinicius Hedler