

SIN 131 – Introdução a Teoria da Computação – Projeto – Parte 1

Desenvolvimento de um sistema para implementar os conceitos fundamentais da Teoria da Computação

Observação: Projeto deve ser implementado em linguagem *Python/Java/GoLang* ou similares, equipes de até dois alunos. Não será aceito o uso de bibliotecas já prontas!!

1. Objetivos

- O programa deve ser capaz de receber como entrada um Autômato Finito Determinístico (AFD) ou Autômato Finito Não-determinístico (AFN).
- Converter um AFN em AFD.
- Simular a aceitação de palavras no AFD e AFN.
- Demonstrar a equivalência entre AFN e o AFD.
- Minimizar AFDs.
- Construção de um front-end para receber as entradas do programa.

2. Aspectos Gerais

O projeto consiste em construir e avaliar uma aplicação

3. Especificação do Problema e Estrutura do Projeto

- **Entrada do AFN:** Seu programa deve ser capaz de receber como entrada um AFN. O AFN é definido por um conjunto de estados, um alfabeto, um conjunto de transições, um estado inicial e um conjunto de estados de aceitação.
- **Conversão de AFN para AFD:** Implemente o algoritmo para converter o AFN em um AFD equivalente.
- **Simulação de Aceitação de Palavras:** Após a conversão, seu programa deve ser capaz de receber uma palavra (uma sequência de símbolos do alfabeto) e simular a execução do AFD com essa palavra.
- **Demonstração de Equivalência:** Após a conversão do AFN para o AFD, seu programa deve demonstrar que o AFN original e o AFD convertido são equivalentes. Isso pode ser feito verificando se ambos aceitam a mesma linguagem.
- **Minimização de AFDs:** Após a conversão de um AFN para um AFD, implemente o algoritmo de minimização de estados para reduzir o número de estados do AFD resultante, se possível.

4. Como será avaliado?

A avaliação do projeto consiste em dois itens avaliativos entregues de forma escalonada em duas etapas.

Primeira: Uma submissão na tarefa correspondente do Moodle da *Etapa 1* (backend) - **20%:**

- Link para um Repositório Versionado (aberto) [Etapa 1]

Segundo: Uma submissão na tarefa correspondente do Moddle *Etapa 2* (front-end) – **10%**

- Link para um Repositório Versionado (aberto) [Etapa 2]
- Um PDF contendo a descrição da estrutura do projeto, contendo possíveis telas, diagramas *Unified Modeling Language (UML)*, tecnologias escolhidas para garantir os requisitos do projeto.

Apresentação: Máximo 15 minutos por grupo;

- Deverá apresentar o racional da implementação do projeto:
 - Potencialidades e Limitações.
 - Requisitos Funcionais específicos (se houver casos extras).
- Para validação da entrega como **correta** das demais etapas, é fundamental que seja feito um teste de funcionalidade ao vivo.

5. Critérios

- Corretude da implementação: O programa converte corretamente o AFN para o AFD? O programa simula corretamente a aceitação de palavras pelo AFD? O programa demonstra corretamente a equivalência entre o AFN e o AFD? O programa constrói corretamente um AFN a partir de uma expressão regular? O programa minimiza corretamente o AFD?
- Robustez: O programa lida adequadamente com entradas inválidas ou malformadas?
- Legibilidade do código: O código é bem estruturado, seguindo as boas práticas de programação? Os nomes das variáveis e funções são descritivos? Comentários são usados para explicar partes complexas do código?

6. Considerações Finais

Situações adversas ou dúvidas serão resolvidas ao longo da disciplina.