



CENTRO FEDERAL DE EDUCAÇÃO TECNOLÓGICA DE MINAS GERAIS  
Unidade Leopoldina  
Engenharia da Computação

## Documentação Trabalho Teoria da Computação

TEORIA DA COMPUTAÇÃO

BRUNO LA GATTA OLIVEIRA  
JUAN HELPES ALBANO  
VICTOR DE SOUZA VILELA DA SILVA

Leopoldina, MG, Brasil  
Outubro de 2023

## Introdução

Uma máquina de Turing determinística (MTD) é um modelo de computador que segue uma sequência de passos predefinida para resolver um problema. Uma máquina de Turing não determinística (MTND) é um modelo de computador que pode seguir várias sequências de passos para resolver um problema. As MTDs são mais simples que as MTNDs, mas podem resolver apenas alguns problemas. As MTNDs são mais complexas, mas podem resolver mais problemas.

Aqui estão alguns exemplos de problemas que podem ser resolvidos por máquinas de Turing:

- Determinação da aceitação de uma cadeia de caracteres.
- Computação de funções recursivas.
- Resolução de problemas de busca.

As máquinas de Turing são um modelo fundamental na teoria da computação. Elas são usadas para definir conceitos como computability e algoritmos. O trabalho prático consistiu na implementação dessas máquinas e essa implementação foi feita em Python. O trabalho foi realizado em grupo, com três integrantes.

## Funcionalidades

A máquina de Turing implementada possui as seguintes funcionalidades:

- Leitura correta dos dois arquivos disponibilizados: "MT-padrão" e "MT-não-determinística", que representam, respectivamente, a descrição formal de uma Máquina de Turing padrão e uma Máquina de Turing não-determinística, e criação da máquina corretamente.
- Possibilita que o usuário submeta strings (ie., palavras) para que as mesmas sejam verificadas na Máquina de Turing. A submissão de strings deverá ser realizada através de um arquivo de teste que contenha diversas strings a serem testadas. Cada linha do arquivo será

composta por uma palavra e o nome do arquivo contendo as palavras a serem testadas deve ser passado como parâmetro pela linha de comando.

- Exibir a sequência de configurações que leva a aceitação na Máquina de Turing não-determinística. Em outras palavras, a computação passo a passo da Máquina de Turing não-determinística.

## Implementação

Um script em Python foi implementado para simular o funcionamento de uma Máquina de Turing (MT). O script permite a simulação de máquinas determinísticas e não determinísticas. Abaixo está uma descrição mais detalhada da implementação do código:

### 1. Importações e Classes:

- O código inicia com duas importações de bibliotecas. O módulo ``re`` é usado para trabalhar com expressões regulares, e ``copy`` é importado para criar cópias de objetos.
- Duas classes são definidas:
  - ``Grafo``: Esta classe representa um grafo que será utilizado para representar as transições da Máquina de Turing.
  - ``Instancia_MT``: Representa uma instância da Máquina de Turing e armazena informações importantes, como o estado atual, o caminho percorrido, a palavra de entrada e a posição do cabeçote.

### 2. Função “`configura_maquina``”:

- Essa função é o ponto de entrada do código e é responsável por configurar e simular a máquina de Turing com base no tipo especificado (determinística ou não determinística) e no arquivo de entrada, contendo as palavras a serem testadas.
- Ela lê as configurações da máquina de um arquivo de entrada, processa essas configurações e, em seguida, chama a função apropriada para a simulação.

### 3. Leitura de Configurações:

- O código solicita ao usuário que especifique o nome do arquivo de configuração da Máquina de Turing e o tipo de máquina desejado (determinística ou não determinística) por meio de interação com a entrada padrão.

#### **4. Função “simula\_MT\_deterministica”:**

- Essa função simula a execução de uma máquina de Turing determinística.
- Para cada palavra de entrada, a simulação é iniciada a partir do estado inicial. A máquina progride, aplicando as transições apropriadas, até que a palavra seja aceita ou rejeitada.

#### **5. Função “simula\_MT\_ao\_deterministica”:**

- Esta função simula a execução de uma máquina de Turing não determinística.
- Diferentemente da versão determinística, várias instâncias da máquina são criadas a partir do estado inicial. Cada instância é acompanhada em uma fila de execução.
- A simulação continua até que uma das instâncias seja aceita ou todas sejam rejeitadas. O caminho percorrido, caso a palavra seja aceita, é mostrado na saída do programa.

#### **6. Processamento do Arquivo de Configuração:**

- O código lê as configurações da máquina de Turing a partir do arquivo especificado. Isso inclui definições de estados, transições e a palavra de entrada.
- As configurações são processadas e organizadas em um dicionário (uma estrutura que contém em python).

#### **7. Execução da Máquina de Turing:**

- A máquina de Turing é simulada com base nas configurações lidas do arquivo. A simulação é realizada para cada palavra de entrada, e o resultado (aceitação ou rejeição) é exibido na saída padrão.

#### **8. Relatórios de Resultados:**

- Após a simulação de cada palavra, o código gera mensagens indicando se a palavra foi aceita ou rejeitada pela máquina de Turing.
- Em resumo, o código fornece uma implementação funcional de uma Máquina de Turing e oferece uma estrutura que permite simular palavras de entrada em máquinas de Turing, tanto determinísticas quanto não determinísticas. Os resultados da simulação são apresentados na saída padrão, o que ajuda a entender o comportamento da máquina de Turing em relação às palavras fornecidas.

## Testes

No trabalho prático de Teoria da Computação, foram realizados vários testes para verificar o devido funcionamento das Máquinas de Turing. Ao final, as máquinas tiveram o funcionamento esperado para cada string simples que foram projetadas para resolver. Os resultados dos testes mostraram que as máquinas funcionam conforme o esperado, aceitando todas as strings simples que foram projetadas para resolver.

## Conclusão

O trabalho prático foi concluído com sucesso. A máquina de Turing implementada atende aos requisitos especificados.

## Observações

- A máquina de Turing implementada é apenas um exemplo básico. Ela pode ser estendida para implementar máquinas de Turing mais complexas.
- O código-fonte da máquina de Turing implementada pode ser melhorado para torná-lo mais eficiente e legível.