

Universidade de São Paulo
Instituto de Ciências Matemáticas e de Computação

SSC0524 - Introdução à Teoria da Computação
Trabalho em Grupo 2
Simulador Universal de Máquina de Turing

INTEGRANTES:

PAULO ROBERTO DOMINGUES DOS SANTOS (11838721)
THIAGO RIBEIRO GOVEIA (10835942)

1. Introdução

Neste trabalho desenvolvemos um simulador universal de Máquinas de Turing, dadas as instruções do arquivo de entrada disponibilizado pelo professor. O simulador é capaz de simular máquinas de Turing determinísticas. Os estados possíveis do autômato são expressos na forma de um grafo onde cada nó representa um estado e possui transições que levam a outros estados. Para uma dada cadeia, o algoritmo percorre o grafo e verifica se o último estado encontrado (quando possível) é um estado de aceitação. Tem como limite de memória o tamanho da memória RAM do computador utilizado para executar o código. O código foi escrito na linguagem JavaScript, executado no ambiente Node.js.

2. Algoritmo

O programa lê o arquivo de entrada e extrai os números de estados, o conjunto de símbolos terminais, o alfabeto estendido da fita (com os símbolos), estados de aceitação, número de transições, as transições, calcula o índice do número de entradas e extrai as entradas.

Para criar o grafo de estados, utilizamos uma função que recebe como parâmetros o número de estados, os estados de aceitação e as transições, em seguida, adicionamos à cada nó do grafo um estado e suas transições e se esse estado é de aceitação.

Para setar as transições, utilizamos uma função que cria transições, onde cada transição é uma quintupla (estado de origem, símbolo lido, estado de destino, símbolo escrito e direção do movimento do cabeçote), criamos um array contendo cada transição à ela, cada array é identificada por uma chave, que é representada pelo mesmo símbolo terminal que aciona a transição. Cada transição guarda uma referência para o estado de destino.

Com a estrutura pronta, resta a lógica de aceitação e rejeição do processamento de cadeias. Para isso, percorremos o grafo verificando a aceitação utilizando uma função recursiva que recebe como entrada o estado e os símbolos da cadeia.

O procedimento é:

1. Leia o símbolo da fita na posição atual do cabeçote.
2. Se não houver transição e não estiver em estado de aceitação, rejeite.
3. Se houver transição e estiver em estado de aceitação, aceite.
4. Escrever símbolo de marcação na posição atual.
5. Mover cabeçote segundo a instrução de direção de movimento da transição.
6. Caso o retorno da função seja de aceitação, escreva "aceita" em uma linha do terminal, caso contrário, escreva "rejeita" e passe para a próxima linha.

3. Complexidade de espaço

A complexidade em espaço do processamento tem comportamento semelhante ao processamento de uma cadeia por uma Máquina de Turing pois a fita tem tamanho teórico

infinito, ou seja, no nosso algoritmo este é limitado à memória, dependendo do comportamento descrito pelas transições.

4. Complexidade de tempo

As complexidades de tempo do algoritmo para a construção da máquina de Turing é **$O(n+m)$** , sendo **n** o número de estados e **m** o número de transições.

O processamento de uma cadeia pela máquina terá complexidade em tempo dependente da linguagem processada pela máquina, pois o processamento pode se estender ao infinito dependendo das transições inseridas.