

Implementação de autômato de pilha

Execução

O projeto conta com exemplos de autômatos para teste, siga o exemplo abaixo para executar o algoritmo:

```
$ python3 StackAuto.py automatos/ex5.1.txt aaabb
```

Onde

- "[StackAuto.py](#)" é o algoritmo de simulação do autômato de pilha;
- "ex5.1.txt" é o formato de trabalho do exemplo localizado no diretório automatos/
- "aaabb" é a palavra de entrada.

Funcionamento do código

A função principal (main()) do algoritmo possui:

- Um vetor de estados (classe Estado), onde fica armazenado o conjunto de estados do autômato assim que eles são criados;
- Um vetor de autômatos (classe Automato), onde ficam armazenados todos os autômatos criados no decorrer da execução (solução para o não-determinismo).

Cada autômato possui os atributos:

- Entrada (classe Pilha), onde será recebida a palavra de entrada;
- Pilha (classe Pilha), que será utilizada como fita de trabalho;
- Estado (classe Estado), representando o estado atual do autômato, começa apontando para o estado inicial e alterna entre os demais estados de acordo com a validação das transições de cada estado.

Cada estado possui seu conjunto de transições (classe Transicao), que são geradas no método Estado.mount() de acordo com as transições dadas no formato de trabalho a partir da linha 7.

O primeiro laço de repetição trata de executar todos os autômatos, o mesmo opera enquanto estiver autômatos no vetor e enquanto nenhum resultado foi alcançado.

Conforme o número de transições válidas no estado atual da máquina, o autômato em questão pode precisar ser duplicado para que todas as transições válidas sejam executadas.

Se o autômato parou em um estado onde não existem transições válidas, o mesmo é removido do vetor de autômatos ou o algoritmo retorna (de acordo com a quantidade de máquinas que ainda têm para executar).

O algoritmo retorna:

- 0 se entrar em um estado de aceitação ou se a pilha estiver vazia;
- 1 se a palavra não for aceita.

Desenvolvedores

Everton Junior de Abreu

Lucas Daniel Deolindo dos Santos

Lucas Souza Santos

Licença

MIT