



UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO
DEPARTAMENTO DE INFORMÁTICA E ESTATÍSTICA
PROGRAMA DE PÓS GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

Aluno: André Luiz Lehmann

Disciplina: Teoria da Computação

Trabalho 1

Assim como a descrição formal de um AFND, cada autômato deste trabalho também é formado pela quádrupla:

- Lista de todos os estados;
- Lista dos estados finais;
- Estado inicial;
- Lista contendo todo o alfabeto aceito pelo autômato;
- Função de transição.

Abaixo segue um exemplo de AFND descrita em formato JSON que o presente simulador interpreta:

```
// -> q0 ---&a---> q1 ---b---> q2
{
  'estados' : ['q0', 'q1', 'q2'],
  'estadosFinais' : ['q2'],
  'estadoInicial' : 'q0',
  'alfabeto' : ['a', 'b'],
  'transicao' : [{
    'estado' : 'q0',
    'simbolo' : "",
    'proximosEstados': ['q1']
  }, {
    'estado' : 'q0',
    'simbolo' : 'a',
    'proximosEstados': ['q1']
  }, {
    'estado' : 'q1',
    'simbolo' : 'b',
    'proximosEstados': ['q2']
  }]
}
```

Aqui temos a descrição de um autômato não-determinístico que aceite tanto a String 'a' quanto 'ab'. Isto é possível pela utilização de uma regra de transição que aceite a String vazia "" vindo de q0 e indo para q1. Podemos ver também que este simulador aceita uma definição parcial da função de transição, pois nem todos os estados possuem transições para todos os símbolos do alfabeto.

A computação deste autômato por este simulador é realizada através da classe SimuladorNaoDeterministico, que recebe um objeto da classe AutomatoND representando o autômato e um objeto da classe String representando a String de entrada, e tem como saída um booleano indicando a aceitação ou não deste String pelo autômato.

Para que seja possível representar um autômato através de um objeto JSON, como o exemplo dado acima, foi criada a classe auxiliar AutomatoLoad, que realiza a leitura de um arquivo contendo o JSON e retorna um objeto do tipo AutomatoND.

A computação deste autômato por este simulador é realizada através da utilização de uma fila de processamento, onde cada elemento da fila contém o estado atual de um ramo de computação e a String a ser processada a partir daquele estado. A primeira iteração desta computação é realizada com o estado inicial definido no autômato e toda a String de entrada. A partir desta computação é feita uma verificação se a entrada já foi toda consumida e caso verdadeiro o simulador retorna que a String de entrada é aceita pelo autômato se o estado atual é um dos estados finais descritos pelo autômato.

A cada iteração, a função de transição é consultada (caso o autômato ainda não tenha aceito a String) e a lista de estados a serem computados é adicionada à fila de processamento, onde cada próximo estado é enfileirado juntamente da String de entrada conforme o conteúdo da mesma após a interpretação de cada entrada, onde quando a função de transição possui um símbolo associado ao estado atual a entrada é consumido e quando a função de transição possui uma transição pela String vazia, a entrada não mantida intacta.

Para facilitar a validação deste simulador, foi criada uma classe para a realização de testes unitários de AFND, a classe SimuladorNaoDeterministicoTest. Este classe realiza a leitura de vários autômatos definidos em arquivos JSON e confere a aceitação ou não de várias Strings para cada autômato.