

Atividade N1 – 2º Bimestre – Ano 2017

Linguagens Formais e Autômatos – Prof. Viotti

Construa um projeto em qualquer linguagem (C#, Java, PHP, etc..) que permita a aceitação ou não de palavras geradas através de uma linguagem regular que possa ser verificada por um autômato finito determinístico e não determinístico.

O projeto deve permitir a entrada de qualquer AFD ou AFN, juntamente com suas palavras a serem verificadas. Como resultado o projeto deve dar a resposta ACEITO ou NÃO ACEITO para cada palavra.

Não existe a necessidade da criação do grafo para representação dos autômatos.

A entrada da tabela de função de transição dos autômatos deve ser através de um arquivo texto. Esse arquivo texto deve seguir a seguinte estrutura:

1ª Linha: definição do tipo do autômato: AFD ou AFN

2ª Linha: Estados do autômato separados por vírgula: q1,q2,q3,qn

3ª Linha: Alfabeto aceito pelo autômato, também separados por virgula: 0,1,2

4ª Linha: Indica o estado inicial. Este estado deve fazer parte dos estados do autômato: q1

5ª Linha: Conjunto de estados finais. Pode haver mais de um estado final: q2,q3

6ª Linha até encontrar #####, compõe a função de transição do autômato, que deve obedecer o formato (estado,símbolo para consumo, estado destino): (q1,0,q2)

Exemplo de um autômato:

```
AFD
q0, q1, q2
0, 1
q0
q2
(q0, 0, q1)
(q1, 0, q2)
(q1, 1, q1)
#####
```

Este autômato exemplo aceita a ER (01*0).

Deve existir uma validação do arquivo de entrada do autômato para que as transições possuam estados válidos, o estado inicial e os estados finais façam parte dos estados do autômato, o arquivo está bem formado, entre outras.

O nome e localização do arquivo de entrada deverá ser informado pelo usuário logo no início da aplicação.



Após a informação do arquivo de entrada do autômato, deverá ser solicitado também o nome e localização do arquivo que contém as palavras a serem verificadas pelo autômato. O nome desse arquivo deverá possuir em sua extensão “.IN”.

Sua aplicação deverá gerar um arquivo com o mesmo nome do arquivo de palavras, porém com a extensão “.OUT”, no mesmo caminho do arquivo “.IN”. Dentro do arquivo “.OUT” deverá existir todas as palavras processadas pelo autômato acrescida do resultado “ACEITO” ou “REJEITADO”

Exemplo do arquivo de entrada de palavras: TESTE.IN

```
00
0111110
0001111000
100001
000
```

Exemplo do arquivo gerado pelo autômato: TESTE.OUT

```
00 ACEITO
0111110 ACEITO
0001111000 REJEITADO
100001 REJEITADO
000 REJEITADO
```

Regras e apresentação do trabalho

- O trabalho poderá ser feito sozinho ou em dupla;
- Data da entrega: 19/05/2017;
- A entrega será feita no laboratório, juntamente com a apresentação/execução do aplicativo;
- Serão fornecidos 3 autômatos (AFN, AFD ou ambos) para cada uma das duplas logo no início da aula. Será dado um tempo de 30 minutos para que vocês possam verificar se o projeto da dupla funciona com os autômatos fornecidos, e com isso vocês terão esse tempo para uma possível correção. (Não será permitido correções após esse tempo);
- Após os 30 minutos de verificação, todos os trabalhos deverão ser entregues via Moodle. Todo o código fonte;
- O professor irá verificar junto a cada dupla o funcionamento do mesmo;
- O arquivo com as palavras a serem testadas será fornecido somente neste momento;
- O valor do trabalho é 10,0 (dez) e será dividido da seguinte forma:
 - Validação das informações do arquivo de definição do autômato: 1,5 pontos;
 - Carga do arquivo de definição do autômato: 0,5 ponto;
 - Execução do autômato AFD: 4,0 pontos;
 - Execução do autômato AFN: 4,0 pontos;



- Para cada palavra aceita por engano ou rejeitada por engano será **descontado**: 0,2 pontos;
- A nota será fornecida logo após a apresentação da dupla.