

UNIVERSIDADE FEDERAL DA FRONTEIRA SUL  
CIÊNCIA DA COMPUTAÇÃO – Campus Chapecó  
CCR: Linguagens Formais e Autômatos

Descrição do PROJETO:

Construção de aplicação para gerar Autômato Finito Determinístico.

Braulio Mello

Última atualização: 23/10/25

Data limite de apresentação regular: até a penúltima semana letiva de aula.

Na última semana é permitida a apresentação, com correções, para recuperação de rendimento.

Objetivo:

Construção de uma aplicação capaz de gerar um AFD livre de estados inalcançáveis e mortos.

Entrada: arquivo com a relação de tokens e/ou GRs de uma linguagem.

Saída: Autômato Finito Determinístico (AFD) e mínimo sem a aplicação de classes de equivalência entre estados.

Descrição:

A aplicação realiza a carga de tokens (palavras reservadas, operadores, símbolos especiais, ...) e Gramáticas Regulares (GR) a partir de um arquivo fonte (texto), gerando o AFND.

Por exemplo, se o arquivo de entrada tem as seguintes informações:

-----  
se  
entao  
senao  
  
<S> ::= a<A> | e<A> | i<A> | o<A> | u<A>  
<A> ::= a<A> | e<A> | i<A> | o<A> | u<A> | ε  
-----

Então, a ação de carga destas informações gera o AFND ilustrado na tabela a seguir. Observe que, exceto o estado inicial, não há 'reuso' dos estados seguintes para incluir as transições no AF. Ou seja, para cada nova transição cria-se um novo estado destino.

δ	s	e	n	t	a	o	i	u
S	A, H	C, M			M	M	M	M
A		B						
*B								
C			D					
D				E				
E					F			
F						G		
*G								
H		I						
I			J					
J					K			
K						L		
*L								
*M		M			M	M	M	M

Neste AF exemplo, os estados finais que reconhecem os respectivos tokens são:

B: se

G: entao

L: senao

M: variavel

No arquivo de carga, usar notação BNF para as GRs.

Para cada token e gramática, a aplicação gera o conjunto de transições rotuladas em um único AF durante o procedimento de carga. No AF, apenas o estado inicial é compartilhado entre diferentes tokens/gramáticas. Os demais estados são exclusivos para as transições dos demais símbolos dos tokens e/ou estados das GRs. Ou seja, após o estado final, na carga, todas as demais transições têm um novo estado como destino.

O AF será indeterminístico quando ocorrer uma ou mais situações em que dois tokens ou sentenças definidas por GR iniciam pelo mesmo símbolo.

Determinização:

Aplicar o teorema de determinização para obter o AFD.

Minimização:

O AFD resultante deve ser submetido ao processo de minimização, contudo, sem aplicar Classe de Equivalência. Não há restrição quanto a estrutura de dados para representar o AF (matriz, dicionário...)

Estado de erro:

Ao final da minimização, acrescentar um último estado final. Este será o estado de erro. Todas as células do AFD não mapeadas devem ser preenchidas com estado de erro. Todas as transições a partir do estado de erro permanecem no estado de erro.

Entrega (até penúltima semana letiva de aula):

- Código fonte da aplicação
- Relatório, em formato de artigo, contendo: identificação autores, resumo, introdução, referencial teórico básico (conceitos essenciais para compreensão do trabalho e trabalhos correlatos), especificação e implementação da solução para gerar AFDs, conclusão e referencial bibliográfico.
- upload no moodle em arquivo único antes da apresentação
- a penúltima semana letiva de aula é a data limite. O trabalho pode ser apresentado assim que estiver pronto no decorrer do semestre.

Apresentação e avaliação:

- Trabalho individual ou em duplas
- Implementação operacional dá direito à apresentação
- A avaliação e atribuição da nota são aplicadas perante apresentação (demonstração da aplicação e arguição). Apenas a entrega da implementação não reverte em nota, permanecendo 'zero'.
- Qualidade da solução, requisitos contemplados, domínio do processo de especificação e implantação da aplicação (perante arguição), teor/clareza/conteúdo do artigo são os principais referenciais para composição da nota.