ICMC-USP

Trabalho em Grupo 1

SCC-0205

2º. Semestre de 2009 - Turma C
Professor: João Luís G. Rosa - e-mail: joaoluis@icmc.usp.br
versão 1 - 25/08/2009

1 Objetivo

Desenvolver o entendimento de Linguagens Formais e seu potencial de representação através da implementação de simuladores de autômatos finitos e de ferramentas para o estudo de gramáticas e expressões regulares.

2 Descrição

O trabalho deve ser preferencialmente realizado em grupos de três. Cada grupo deve selecionar uma das aplicações sugeridas abaixo (opção 1 ou 2), projetá-la e desenvolvê-la, empregando uma das seguintes linguagens de programação: C, C++, Pascal ou Java.

- **Opção 1**: *Simulador Universal de Autômatos Finitos*: O programa deve aceitar a especificação de um AFD ou AFN e a partir daí para uma dada lista de cadeias, dizer quais as que pertencem (saída: **aceita**) e quais as que não pertencem (saída: **rejeita**) à linguagem reconhecida pelo autômato.
- **Opção 2**: *Simulador Universal de Autômatos de Pilha*: O programa deve aceitar a especificação de um APN e a partir daí para uma dada lista de cadeias, dizer quais as que pertencem (saída: **aceita**) e quais as que não pertencem (saída: **rejeita**) à linguagem reconhecida pelo autômato. Utilizar a **aceitação pela pilha vazia**.

3 Produto

O programa a ser implementado neste projeto deve seguir rigorosamente os formatos de entrada e saída (ver seção "Arquivos Texto de Entrada e de Saída" abaixo), uma vez que todos os projetos serão submetidos, no período de **28 de setembro a 02 de outubro de 2009**, ao corretor automático Boca (`http://frangolino.icmc.usp.br/boca`). O sistema permitirá que se escolha a opção (1 ou 2) na submissão. Recomenda-se que a primeira submissão ocorra antes do prazo final, para que sejam possíveis eventuais correções. **O prazo final é improrrogável**. Além do programa, um relatório com a descrição do trabalho deverá ser entregue (ver seção "Critérios" abaixo).

4 Critérios

Os critérios de correção dos trabalhos são:

 Continua na próxima página...

1. (80%) O programa funciona corretamente para todos os casos de teste;
2. (20%) **Documentação**: relatório simples que explica as técnicas utilizadas para implementar a máquina escolhida. Discutir a qualidade da solução implementada, a estruturação do código e a eficiência da solução em termos de espaço e tempo. A documentação deverá ser entregue na primeira aula após o final do prazo de submissão do trabalho, ou seja, no dia 06/10/2009.

Atenção: **O plágio (cópia) de programas não será tolerado. Quaisquer programas similares terão nota zero independente de qual for o original e qual for a cópia.**

5 Arquivos Texto de Entrada e de Saída

Arquivo Texto de Entrada:

- 1^a. Linha: número de estados: para o conjunto de estados Q, assume-se os nomes dos estados de q_0 a q_{n-1}, onde n é o número de estados (Obs.: q_0 é o estado inicial, quando houver um único estado inicial: para a opção 1, quando se tratar de AFD ou para a opção 2). Portanto, basta entrar com o número de estados. Assuma $1 \leq n \leq 10$;
- 2^a. Linha: o conjunto de símbolos terminais (Σ): entrar com a quantidade de símbolos terminais seguida dos elementos separados por espaço simples. Assume-se tamanho máximo igual a 10;
- 3^a. Linha: o número de estados iniciais para a opção 1 (se for AFD, é igual a 1: q_0; se for AFN, usa-se q_0, q_1, etc. para os estados iniciais); o conjunto de símbolos de pilha (Γ) para a opção 2 (APN): entrar com a quantidade de símbolos de pilha seguida dos elementos (de um caractere) separados por espaço simples. Assume-se que o símbolo inicial Z_0 é representado por Z. Assume-se tamanho máximo igual a 10;
- 4^a. Linha: o conjunto de estados de aceitação (F): entrar com a quantidade de estados de aceitação seguida dos elementos separados por espaços. Lembre-se de entrar apenas com os números de 0 a 9;
- 5^a. Linha: o número de transições (δ) da máquina (máximo de 50).
- a partir da 6^a Linha: as transições: entra-se com um δ em cada linha, com os elementos separados por espaço. Para o AFD/AFN (opção 1): $q\ x\ q'$, onde $q, q' \in Q$, $x \in \Sigma \cup \{\lambda\}$. Para o APN (opção 2): $q\ x\ Z\ q'\ \sigma$, onde $q, q' \in Q$, $x \in \Sigma \cup \{\lambda\}$, $Z \in \Gamma$ e $\sigma \in \Gamma^*$. Represente a cadeia vazia (λ) como "-".
- Linha depois das transições: entrar com o número de cadeias de entrada (máximo de 10).

 Continua na próxima página...

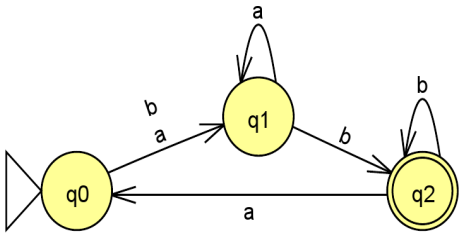
- Próximas Linhas: cadeias de entrada: entrar com uma em cada linha. Comprimento máximo de cada cadeia = 20 símbolos.

Arquivo Texto de Saída:

- a partir da 1^a. Linha: a informação sobre a aceitação ou não da respectiva cadeia de entrada, **na ordem** do arquivo de entrada. Se a cadeia de entrada pertencer à linguagem reconhecida pelo autômato, a cadeia de saída será "aceita". Caso a cadeia de entrada não pertença à linguagem reconhecida pelo autômato, a cadeia de saída será "rejeita".

6 Exemplos

- Exemplo 1 - Autômato finito determinístico (AFD) que processa a linguagem regular $(a+b)a^*bb^*(a(a+b)a^*bb^*)^*$.



Arquivo Texto de Entrada[1]:

1. 3
2. 2 a b
3. 1
4. 1 2
5. 6
6. 0 a 1
7. 0 b 1
8. 1 a 1
9. 1 b 2
10. 2 a 0
11. 2 b 2
12. 10

[1]Os números das linhas **não** devem aparecer no arquivo-texto. Estão colocados aqui apenas para facilitar o entendimento.

 Continua na próxima página...

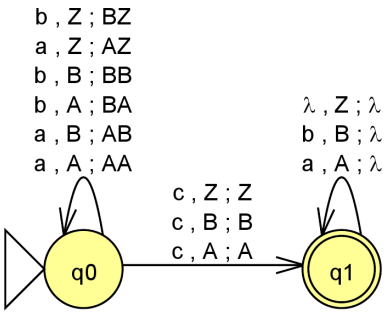
13. abbbba
14. aabbbb
15. bbabbabbabbb
16. bbbbbbbbbbb
17. -
18. abababababab
19. bbbbaabbbb
20. abba
21. a
22. aaa

Arquivo Texto de Saída:

1. rejeita
2. aceita
3. aceita
4. aceita
5. rejeita
6. rejeita
7. aceita
8. rejeita
9. rejeita
10. rejeita

 Continua na próxima página...

- Exemplo 2 - Autômato de pilha (APN) que processa a linguagem wcw^R, $w \in \{a, b\}$.



Arquivo Texto de Entrada:

1. 2
2. 2 a b
3. 3 Z A B
4. 1 1
5. 12
6. 0 a Z 0 AZ
7. 0 a A 0 AA
8. 0 a B 0 AB
9. 0 b Z 0 BZ
10. 0 b A 0 BA
11. 0 b B 0 BB
12. 0 c Z 1 Z
13. 0 c A 1 A
14. 0 c B 1 B
15. 1 a A 1 -
16. 1 b B 1 -
17. 1 - Z 1 -
18. 10
19. abbcbba
20. aabbcbbaa
21. bbabbacbbabbb
22. bbbbbcbbbbbb
23. -

 Continua na próxima página...

24. abababababab
25. bbbbacabbbb
26. abba
27. c
28. aaa

Arquivo Texto de Saída:

1. aceita
2. aceita
3. rejeita
4. rejeita
5. rejeita
6. rejeita
7. aceita
8. rejeita
9. aceita
10. rejeita