

ICMC-USP
Trabalho Individual 2
SCC-5832

1º. Semestre de 2012
Professor: João Luís G. Rosa - e-mail: joaoluis@icmc.usp.br
versão 1 - 08/05/2012

1 Objetivo

Desenvolver o entendimento de Linguagens Formais e seu potencial de representação através da implementação de simuladores de autômatos de pilha.

2 Descrição

Cada aluno deve projetar e desenvolver a aplicação abaixo, empregando a linguagem de programação C ou C++.

- *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 **24 e 25 de maio de 2012**, ao corretor automático Boca (<http://caravelas.icmc.usp.br/boca>). 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:

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 05/06/2012.

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:

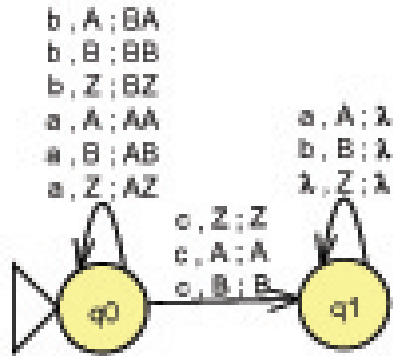
- 1^a. Linha: número de estados: para o conjunto de estados Q , assume-se os nomes dos estados de q_0 a q_{n-1} , onde n é o número de estados (Obs.: q_0 é o estado inicial). Portanto, basta entrar com o número de estados. Assuma $1 \leq n \leq 10$;
- 2^a. Linha: o conjunto de símbolos terminais (Σ): 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 conjunto de símbolos de pilha (Γ): 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: $q x Z q' \sigma$, 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).
- 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 Exemplo

- 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. abbcbbba

20. aabbcbbaa
21. bbabbacbbabbb
22. bbbbbcbbbbbb
23. -
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