1/2

Construa Mealy e Moor

- Cada seqüência de símbolos "a" é mapeada em um único símbolo "0";
- Cada seqüência de símbolos "b" é mapeada em um único símbolo "1";
- Cada seqüência de símbolos "c" é mapeada em um único símbolo "2".

Exemplos de entradas e correspondentes saídas:

abbccc	produz	012
abcaaabbcccc	produz	012012
abc	produz	012
abcabcabc	produz	012012012

Construa, para cada situação, uma Máquina de Mealy e uma Máquina de Moore que realizem a transdução proposta.

114. Construa um transdutor finito que:
   a) Aceite a linguagem $dd^*(,dd^* \mid \varepsilon)$ sobre o alfabeto $\{d,\text{``},\text{''}\}$, representando números em ponto flutuante com ou sem parte decimal;
   b) Converta as sentenças dessa linguagem para o formato:

$$(d \mid dd \mid ddd)(.ddd)^*(,d \mid ,dd \mid ,ddd \mid \varepsilon)$$

ou seja, agrupe os algarismos d de três em três, a partir da vírgula e em direção à esquerda, separando cada grupo de três com o símbolo ".". Além disso, no máximo os três primeiros algarismos depois da vírgula, se tantos houver, devem ser reproduzidos na saída. São exemplos de entradas e respectivas saídas:

$$dddd \rightarrow d.ddd$$
$$ddddd,dddd \rightarrow dd.ddd,ddd$$
$$ddddddd,d \rightarrow d.ddd.ddd,d$$
$$d,ddddddd \rightarrow d,ddd$$

Sugestão: use um autômato não-determinístico.

115. Construa um transdutor finito que converta números em hexadecimal no formato $xx^* - H$, sobre $\{0,1,2,3,4,5,6,7,8,9,a,b,c,d,e,f,-,H\}$, para números correspondentes em binário, no formato $yy^* - B$, sobre $\{0,1,-,B\}$. Zeros à esquerda no número hexadecimal de entrada devem ser desconsiderados. Cada dígito hexadecimal válido deve ser convertido em quatro dígitos binários. São exemplo de transdução:

$$3a1 - H \Rightarrow 001110100001 - B$$
$$0 - H \Rightarrow 0000 - B$$
$$000 - H \Rightarrow 0000 - B$$
$$0079f0 - H \Rightarrow 0111100111110000 - B$$
$$12 - H \Rightarrow 00010010 - B$$

Mealy e Moore
continua lá encima

116. Considere a linguagem $(x \mid X \mid \sqcup)^+$**.**, definida sobre o alfabeto $\{x,X,\sqcup,\textbf{.}\}$. Alguns exemplos de sentenças pertencentes a essa linguagem são:
   - x**.**
   - $\sqcup$**.**
   - $X\sqcup\sqcup\sqcup$**.**
   - $\sqcup\sqcup x\sqcup\sqcup$**.**
   - $\sqcup\sqcup xXX\sqcup XxxXx$**.**
   - $xXxx\sqcup xx\sqcup\sqcup x\sqcup xXX\sqcup\sqcup\sqcup$**.**

Essa linguagem representa frases em que as palavras são elementos de $(x \mid X)^+$, letras minúsculas e maiúsculas são representadas, respectivamente, pelos símbolos "x" e "X", cada espaço em branco, usado para separar as palavras, é representado pelo símbolo "$\sqcup$", e o ponto, ao final da frase, é representado pelo símbolo "**.**". Pede-se para construir uma seqüência de transdutores (Mealy ou Moore) que, além de aceitarem essa linguagem de entrada, incorporem, cada qual, uma das transduções abaixo especificadas, de forma cumulativa:
   a) Remoção do excesso de brancos entre palavras consecutivas, preservando um único espaço entre elas (exemplo: $x\sqcup\sqcup\sqcup xxx$**.** $\Rightarrow x\sqcup xxx$**.**);
   b) Remoção de todos os brancos no início da frase, antes da primeira palavra (exemplo: $\sqcup\sqcup\sqcup xxx$**.** $\Rightarrow xxx$**.**);
   c) Remoção de todos os brancos no final da frase, depois da última palavra e antes do ponto final (exemplo: $x\sqcup xxxx\sqcup\sqcup$**.** $\Rightarrow x\sqcup xxx$**.**);
   d) Substituição da primeira letra da primeira palavra da frase por uma letra maiúscula, convertendo todas as demais para minúscula (exemplo: $xXX\sqcup XxXx$**.** $\Rightarrow Xxx\sqcup xxxx$**.**);
   e) Reconhecimento e transdução de uma seqüência de frases, ou seja, da linguagem $((x \mid X \mid \sqcup)^+\textbf{.})^+$, garantindo a existência de exatamente um espaço em branco entre duas frases consecutivas, logo depois do ponto ao final da primeira e imediatamente antes da primeira palavra da frase seguinte (exemplo: $Xx\sqcup xx$**.**$Xxx\sqcup x$**.** $\Rightarrow Xx\sqcup xx$**.**$\sqcup Xxx\sqcup x$**.**);
   f) Remoção de frases vazias, ou seja, constituídas apenas por um espaço em branco e um ponto final (exemplo: X**.**$\sqcup$**.**$\sqcup Xx\sqcup x$**.** $\Rightarrow X$**.**$\sqcup Xx\sqcup x$**.**).

117. Considere a linguagem $xx^*(-xx^*)^*$, sobre o alfabeto $\{x,-\}$. Construa um transdutor finito que aceite essa linguagem e gere, na saída, cadeias correspondentes sobre o alfabeto $\{x,y,\#\}$ que reproduzam a

2/2

CONSTRUA AS MÁQUINAS DE Mealy e Moore

cadeia de entrada, mas com as seguintes modificações:

a) As subcadeias de entrada xx^* que contiverem três ou menos símbolos x devem ser reproduzidas de forma idêntica na saída (com três, dois ou um símbolo x respectivamente);

b) As subcadeias de entrada xx^* que contiverem quatro ou mais símbolos x devem ser reproduzidas na saída como $xxxy$;

c) O símbolo "$-$" da entrada deve ser substituído pelo símbolo "#" na saída.

Exemplos de entradas e correspondentes saídas:

i) $x-x$ gera $x\#x$;

ii) $xxx-xxxx$ gera $xxx\#xxxy$;

iii) $xxxxxx-xxx-xx$ gera $xxxy\#xxx\#xx$;

iv) $x-xx-xxx-xxxx-xxxxx$ gera $x\#xx\#xxx\#xxxy\#xxxy$.

118. Considere a linguagem $\oplus^*(\approx\oplus^*)^*$, definida sobre o alfabeto $\{\oplus,\approx\}$. Construa um transdutor finito (Mealy ou Moore) que, além de aceitar essa linguagem de entrada, gere uma linguagem de saída sobre o mesmo alfabeto da entrada, com as seguintes características:

a) Cada seqüência da forma $\oplus^i$, com $i>5$, deve gerar na saída a seqüência $\oplus^5$;

b) Cada seqüência da forma $\oplus^i$, com $i<3$, deve gerar na saída a seqüência $\oplus^3$;

c) Cada seqüência da forma $\oplus^i$, com $3\leqslant i\leqslant 5$, deve gerar na saída a seqüência $\oplus^i$ correspondente;

d) O símbolo "$\approx$" deve ser repetido na saída.

Exemplos de entradas e correspondentes saídas:

i) ε gera $\oplus\oplus\oplus$;

ii) $\oplus$ gera $\oplus\oplus\oplus$;

iii) $\approx$ gera $\oplus\oplus\oplus\approx\oplus\oplus\oplus$;

iv) $\approx\oplus\approx$ gera $\oplus\oplus\oplus\approx\oplus\oplus\oplus\approx\oplus\oplus\oplus$;

v) $\oplus\oplus\approx\oplus\oplus\oplus\oplus$ gera $\oplus\oplus\oplus\approx\oplus\oplus\oplus\oplus$;

vi) $\oplus\oplus\oplus\approx\oplus\oplus\oplus\oplus\oplus\oplus\approx$ gera $\oplus\oplus\oplus\approx\oplus\oplus\oplus\oplus\oplus\approx\oplus\oplus\oplus$.

119. Construa um transdutor finito (Mealy ou Moore) para a linguagem de entrada $(a\mid b\mid c\mid d)^*$, gerando a linguagem de saída $L\subseteq 1^*$, de tal forma que a quantidade de símbolos "1" na cadeia de saída indique a quantidade de subcadeias da forma bcd^* presentes na cadeia de entrada.

Exemplos de entradas e correspondentes saídas:

i) $abcdacbcdddbcacbcd$ gera 1111;

ii) $bcdabcddcaa$ gera 11;

iii) $aaaacdb$ gera ε.

120. Demonstrar que a linguagem de saída de um transdutor finito é sempre regular.

121. Demonstrar a equipotência entre as Máquinas de Mealy e as Máquinas de Moore.

122. Considere as linguagens de entrada L_E e de saída L_S definidas a seguir.

- $L_E=\{a,b,c\}^*$;
- $L_S\subseteq\{a,b,c,3,4,5\}^*$.

Obtenha um transdutor finito que efetue o mapeamento de $w\in L_E$ para $w'\in L_S$, de tal forma que w' seja uma representação compacta da cadeia w, conforme o seguinte critério: toda subcadeia presente na cadeia de entrada w que contenha três, quatro ou cinco símbolos repetidos em seqüência deverá ser substituída, na cadeia de saída w', pela subcadeia correspondente formada pelo símbolo que se repete e o número 3, 4 ou 5. São exemplos de transdução: $\varepsilon\to\varepsilon$, $a\to a$, $cccc\to c4$, $abca\to abca$, $ccccccccb\to c5c3b$ e $aaaabcaaabb\to a4bca3bb$.

"Pumping lemma" para as linguagens regulares

123. Sobre o "Pumping Lemma" para as linguagens regulares:

a) O que é?

b) Qual o seu enunciado?

c) Quais são as suas hipóteses?

d) No que se baseia a sua demonstração?

e) Qual é a sua principal aplicação?

f) Em quais passos pode ser decomposta essa sua principal aplicação?

124. Considere a linguagem L definida pela gramática: $(\{S,B,C,a,b,c\},\{a,b,c\},P,S)$, com: $P=\{S\to aS\mid aB, B\to bB\mid C, C\to cC\mid c\}$

a) Construa um autômato finito sem transições em vazio, sem não-determinismos e sem estados inúteis ou inacessíveis que aceite L.

b) Usando o autômato construído acima, selecione uma sentença de comprimento adequado e mostre que o "Pumping Lemma" das linguagens regulares é verificado para tal sentença.

125. Considere uma linguagem L qualquer. Com base no "Pumping Lemma", que estratégias você usaria para:

a) Tentar provar que L é regular?

b) Tentar provar que L não é regular?

126. Prove que as seguintes linguagens não são regulares. Nas expressões abaixo, $|w|_\sigma$ representa o número de ocorrências do símbolo σ na cadeia w.

a) $\{a^ib^{2i}\mid i\geqslant 1\}$;