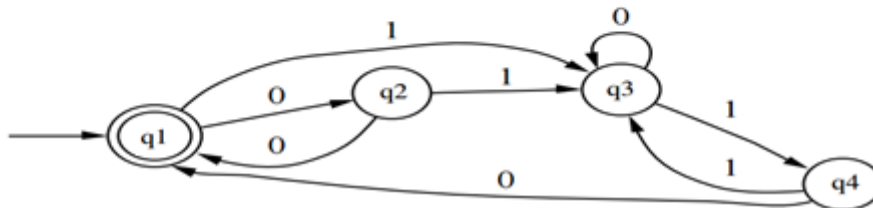
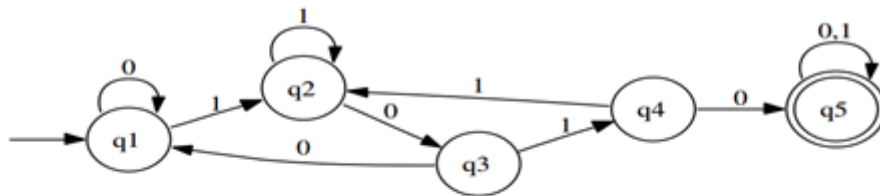


- 1- Mostre que as seguintes linguagens são lineares à direita (i.e. regulares)
 - a) O conjunto de todas as palavras sobre o alfabeto inglês que começam e terminam com a mesma letra
 - b) $\{w \in \{a,b,c\}^* \mid |w|_a \text{ é par ou } |w|_b \text{ é ímpar}\}$
 - c) $\{w \in \{a,b,c\}^* \mid |w|_a \text{ é par e } |w|_b \text{ é ímpar}\}$
 - d) $\{w \in \{a,b,c\}^* \mid |w|_a \equiv 3 \pmod{4}\}$

- 2- Descreva usando expressões regulares as linguagens reconhecidas pelos seguintes autômatos



- 3- Escreva um autômato finito para simular o comportamento de uma máquina que vende café expresso a 25 centavos e capuchino a 45 centavos. Assuma que a máquina aceita moedas de 5, 10, 25 e 50 centavos e também de um real; porém não devolve troco. O tipo de café é escolhido após ter inserido dinheiro suficiente.

- 4- Escreva um autômato finito para reconhecer sequências de parênteses balanceados com profundidade máxima de aninhamento igual a três. Por exemplo, as sequências $()()$, $((()))$ ou $((()))()$ pertencem à linguagem do autômato porém as sequências $((()))$ ou $((()((()))))$ não.

- 5- Construa um autômato finito para a seguinte expressão regular $((a^* | bd)^* a(b | \epsilon) c)^* d$
 - a) Usando o método de Thompson
 - b) Um autômato **não determinístico sem ϵ -transições**
 - c) Um autômato **determinístico**

- 6- Escreva uma expressão regular e um autômato finito para gerar uma atribuição múltipla estilo C++. A expressão da parte direita pode ser um identificador ou um número real. A parte esquerda pode estar composta por vários identificadores separados pelos operadores =, +=, -=, *=, /= e %=.

Exemplo: quantidade = Soma += Ultimo -= Anterior *= - 18.45