

Disponível em <https://github.com/reicavera/LFA>

Escolha os seus 3 exercícios que só você irá fazer para apresentar para o professor. Sua solução deve estar impecável.

1. Seja a GR $G = (\{P, A, B\}, \{a, b\}, R, P)$, onde R consta das regras:

$$P \rightarrow aP \mid bP \mid aA$$

$$A \rightarrow a \mid bB$$

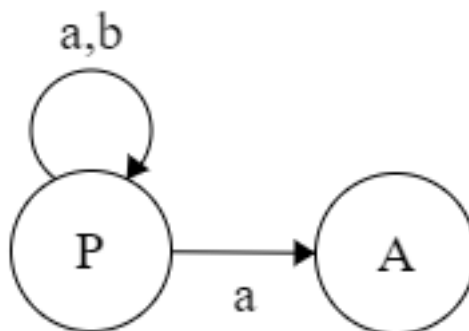
$$B \rightarrow bA$$

Construa, a partir de G , um AFN que aceite $L(G)$.

Partindo da variável P , temos que

$$P \rightarrow aP \mid bP \mid aA$$

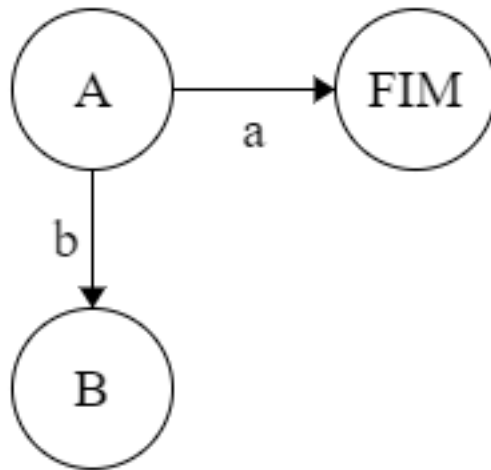
,que pode ser construído como:



Na variável A temos:

$$A \rightarrow a \mid bB$$

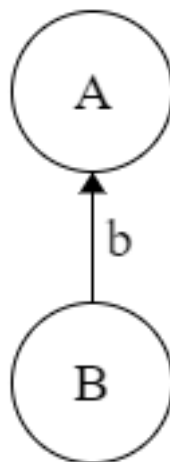
,que pode ser construído como:



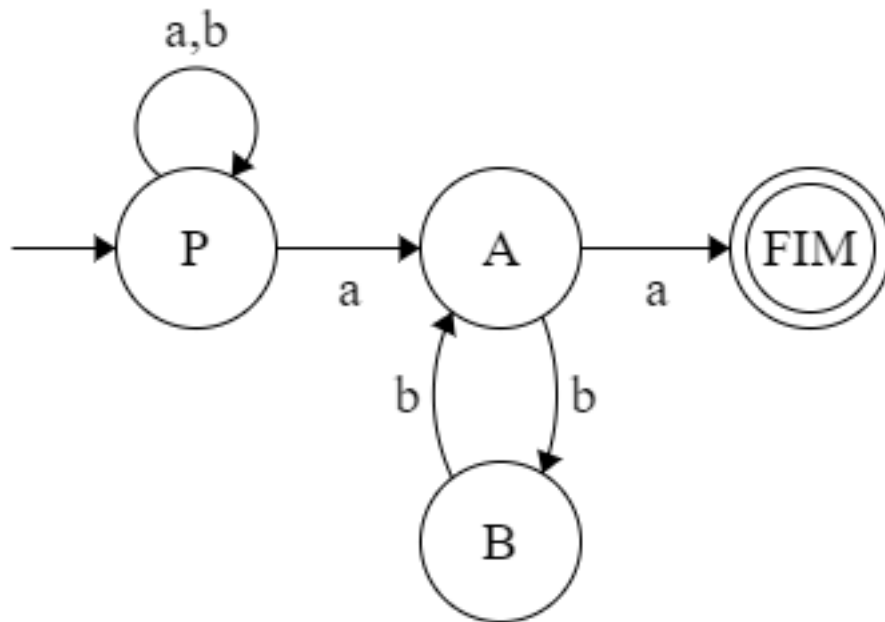
E por fim em B temos:

$$B \rightarrow bA$$

,que pode ser construido como:



Logo,Uma AFN que eceite $L(G)$ é:



2. Construa uma máquina de Mealy que determine o quociente da divisão por 3 de um número na representação binária.

Temos que o valor do n -ésimo algarismo menos significativo q_n do quociente é dado pela divisão do resto r_{n+1} obtido pelo resto da divisão do valor de q_{n+1} vezes a representação usada (nesse caso a binária) mais o n -ésimo algarismo menos significativo do dividendo d_n pelo divisor (no caso 3), ou seja, para divisão de 3 na representação binária temos que $q_n = (r_{n+1} * 2 + d_n) \div 3$. Dessa forma podemos criar uma máquina de Mealy através dos diferentes restos 00, 01 e 10 em representação binária:

estado 00 lendo 0: $000 \div 11 \rightarrow$ resto 00 e quociente 0

estado 00 lendo 1: $001 \div 11 \rightarrow$ resto 01 e quociente 0

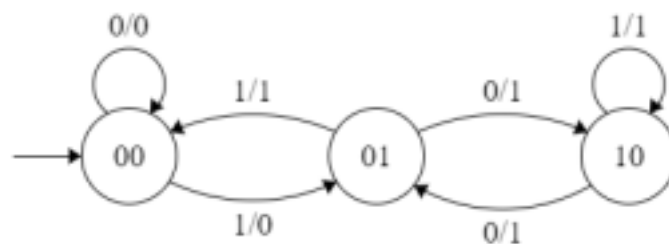
estado 01 lendo 0: $010 \div 11 \rightarrow$ resto 10 e quociente 0

estado 01 lendo 1: $011 \div 11 \rightarrow$ resto 00 e quociente 1

estado 10 lendo 0: $100 \div 11 \rightarrow$ resto 01 e quociente 1

estado 10 lendo 1: $101 \div 11 \rightarrow$ resto 10 e quociente 1

Assim, temos a seguinte máquina de Mealy:



3. Construa uma máquina de Mealy que some dois números na base binária. Os números devem ser supridos por meio do alfabeto $\{[0, 0], [0, 1], [1, 0], [1, 1]\}$, dígitos menos significativos em primeiro lugar. Por exemplo, para somar os números 13 e 20, pode-se suprir a palavra $[0,1][1,0][1,1][0,0][1,0]$, onde os primeiros dígitos, 01101, codificam o número 13, e os segundos, 10100, codificam o número 20; neste caso, a saída deve ser 100001, que codifica o número 33.

A soma em binário possui os seguintes resultados: $0+0=0$ $0+1=1$ $1+0=1$ $1+1=10$ Note que a soma $1+1$ cria um dígito a mais que pode influenciar na soma do dígito imediatamente mais significativo. Logo podemos criar uma máquina de Mealy que tenha estados 0 e 1 para indicar se há ou não interferência desse dígito a mais: estado 0 e $[0,0]$: $0+0=0 \rightarrow$ estado 0 e saída 0

estado 0 e $[1,0]$: $1+0=0 \rightarrow$ estado 0 e saída 1

estado 0 e $[0,1]$: $0+1=0 \rightarrow$ estado 0 e saída 1

estado 0 e $[1,1]$: $1+1=10 \rightarrow$ estado 1 e saída 0

estado 1 e $[0,0]$: $0+0+1=10 \rightarrow$ estado 0 e saída 1

estado 1 e $[0,1]$: $0+1+1=10 \rightarrow$ estado 1 e saída 0

estado 1 e $[1,0]$: $1+0+1=10 \rightarrow$ estado 1 e saída 0

estado 1 e $[1,1]$: $1+1+1=11 \rightarrow$ estado 1 e saída 1

Note ainda que é necessário criar um estado especial que será usado apenas quando toda a palavra já foi lida e estamos no estado 1:

estado 1 e λ : estado λ e saída 1

assim, temos a seguinte máquina de Mealy:

