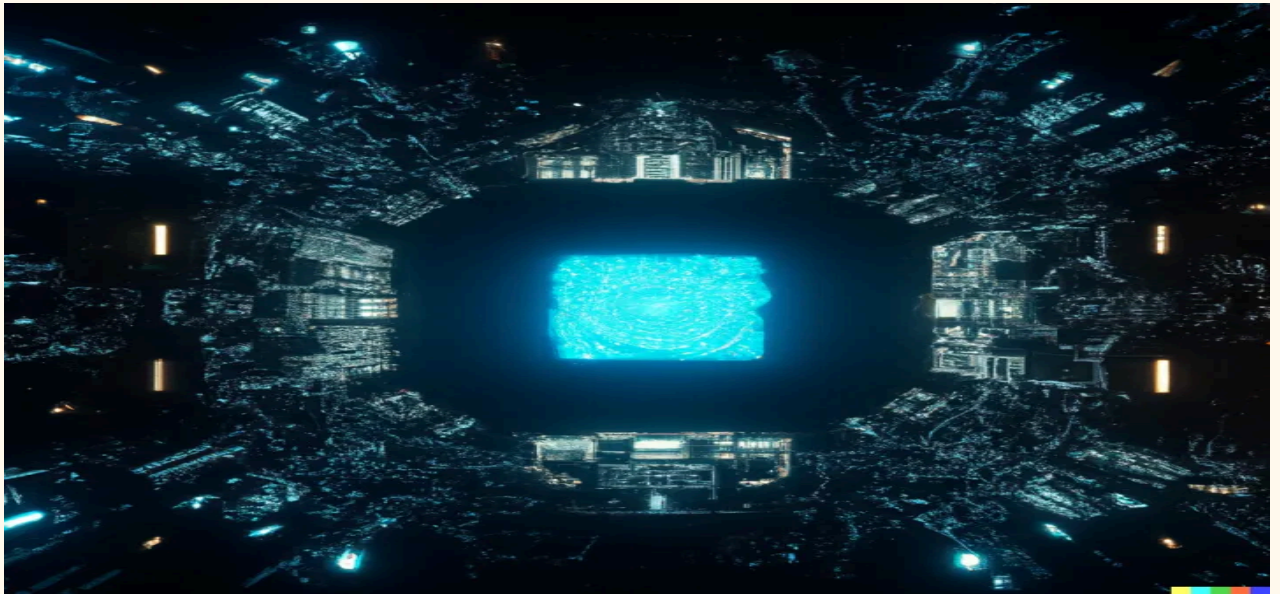


## TP1

# Autômatos Finitos (Não)Determinísticos

---



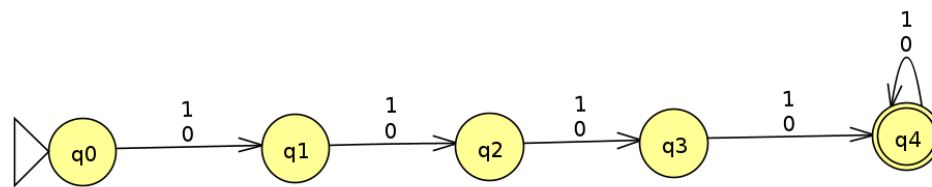
## INTRODUÇÃO

O trabalho prático trata-se de construir autômatos finitos determinísticos e não determinísticos que reconheçam algumas linguagens regulares, a fim de testar o conhecimento adquirido em aulas .

<b>INTRODUÇÃO</b>	<b>1</b>
Questão 1:	2
Questão 2:	3
Questão 3:	5
Questão 4:	6
Questão 5:	7
Questão 6:	8
Questão 7:	9
Questão 8:	10
Questão 9:	11
Questão 10:	12
Questão 11:	13

**Questão 1:**

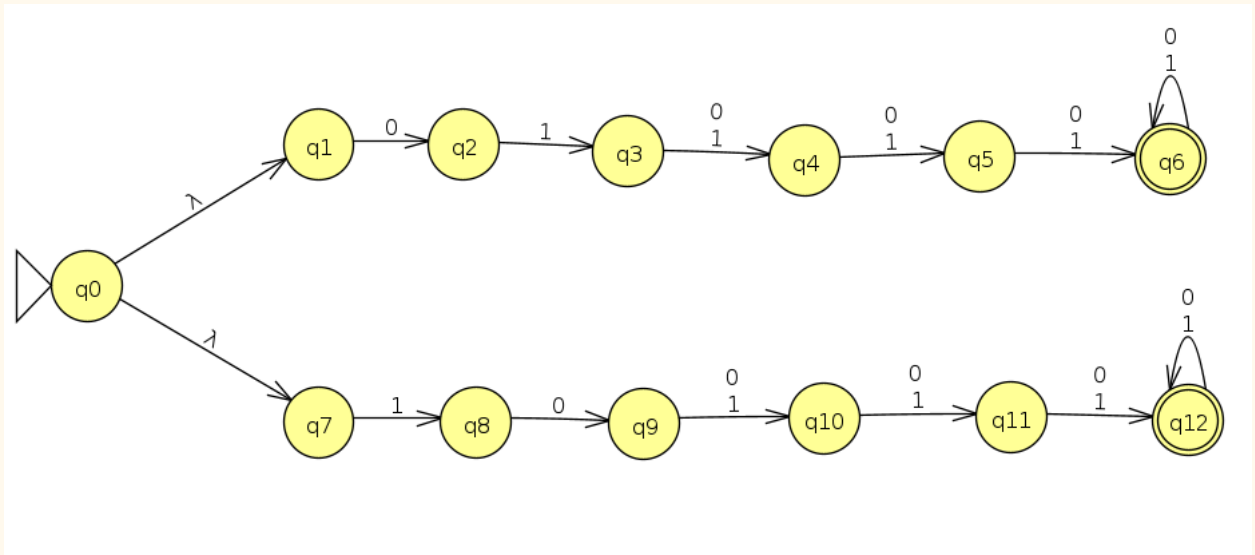
$\{w \in (0,1)^* \mid w \text{ tenha um tamanho maior do que } 3\}$



Input	Result
000	Reject
001	Reject
111	Reject
0000	Accept
1111	Accept
0101	Accept

### Questão 2:

$\{w \in (0,1)^* \mid |w| \geq 5, \text{ e cujos dois primeiros símbolos sejam sempre diferentes um do outro}\}$



Input	Result
000	Reject
110	Reject
0110	Reject
10101	Accept
01011	Accept
010111111	Accept

### Questão 3:

$\{w \in (0,1)^* \mid \text{cada } 0 \text{ de } w \text{ é imediatamente seguido de, no mínimo, dois } 1\text{'s}\}$

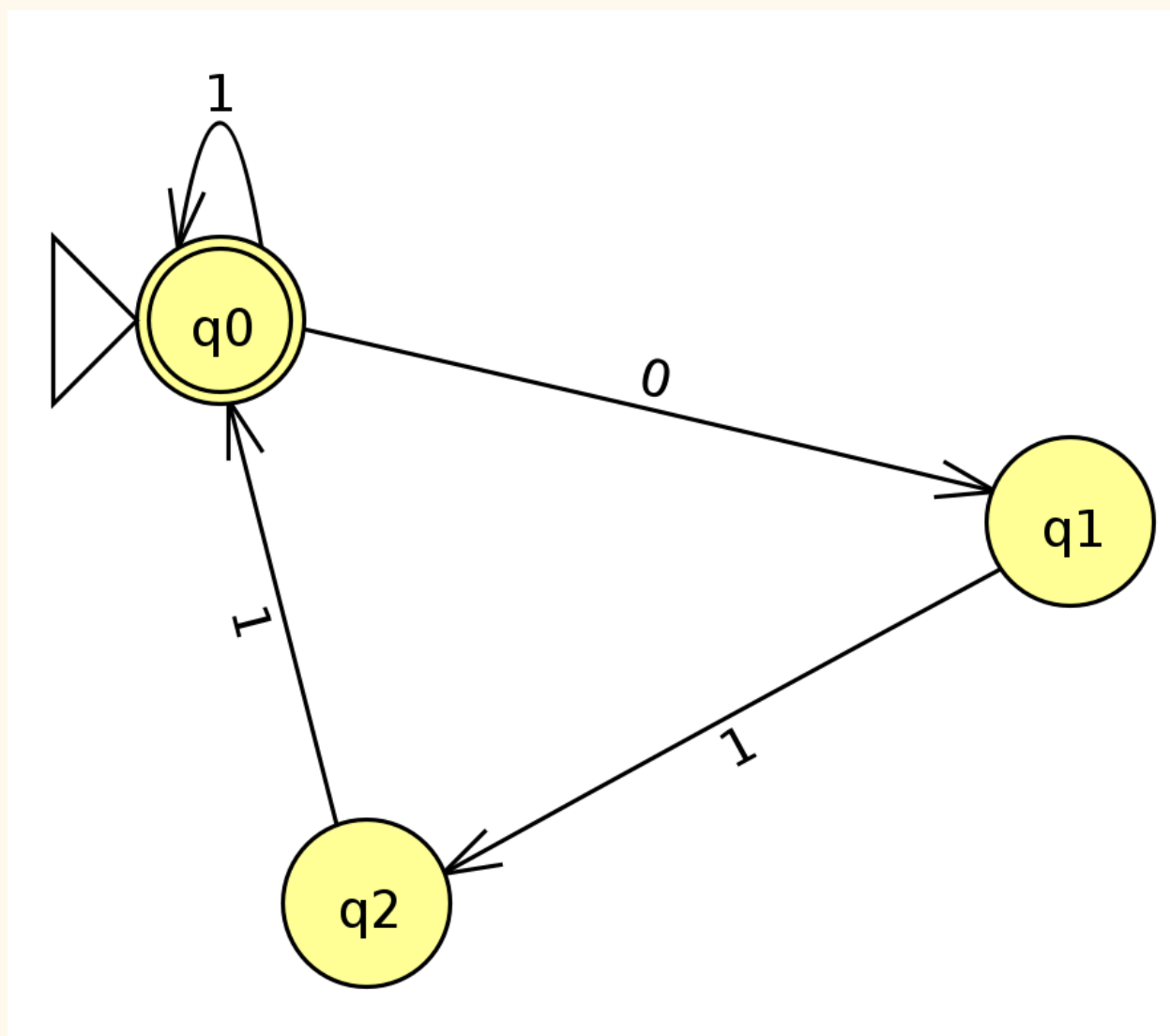


Table Text Size	
Input	Result
01	Reject
10	Reject
110	Reject
1	Accept
011	Accept
111011	Accept

### Questão 4:

$\{w \in (0,1)^* \mid |w| > 3 \text{ e os primeiros 3 símbolos de } w \text{ contêm, no mínimo, dois 1's}\}$

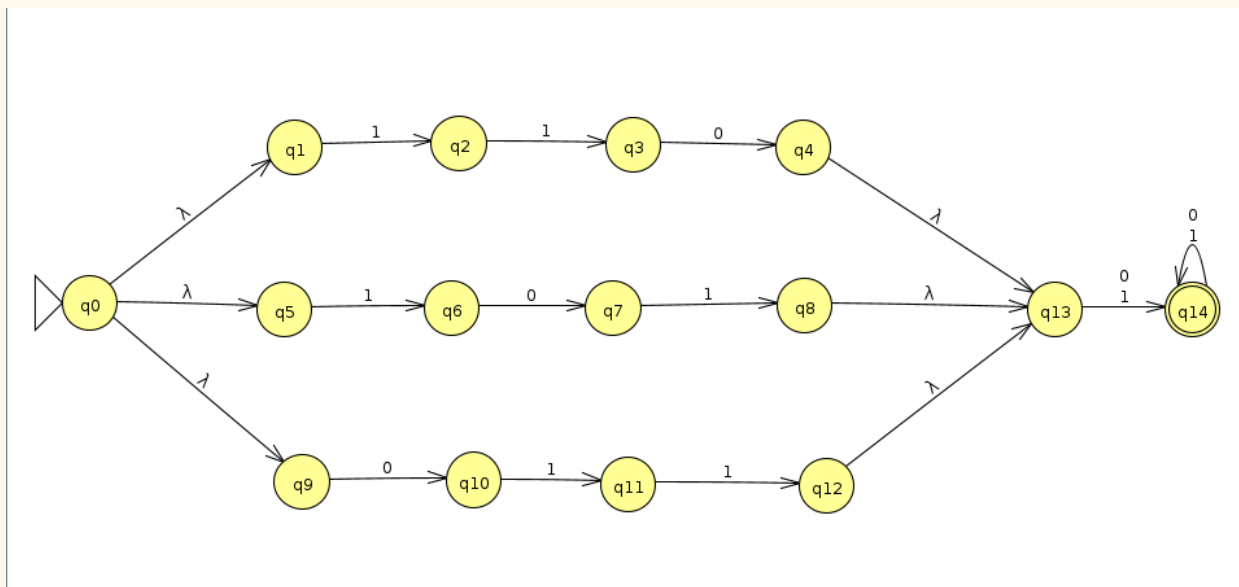


Table Text Size	
Input	Result
001	Reject
111	Reject
011	Reject
0111	Accept
1101	Accept
10100	Accept

### Questão 5:

$\{w \in (0,1)^* \mid w \text{ não possui a subpalavra } 110 \text{ e nem a subpalavra } 001\}$

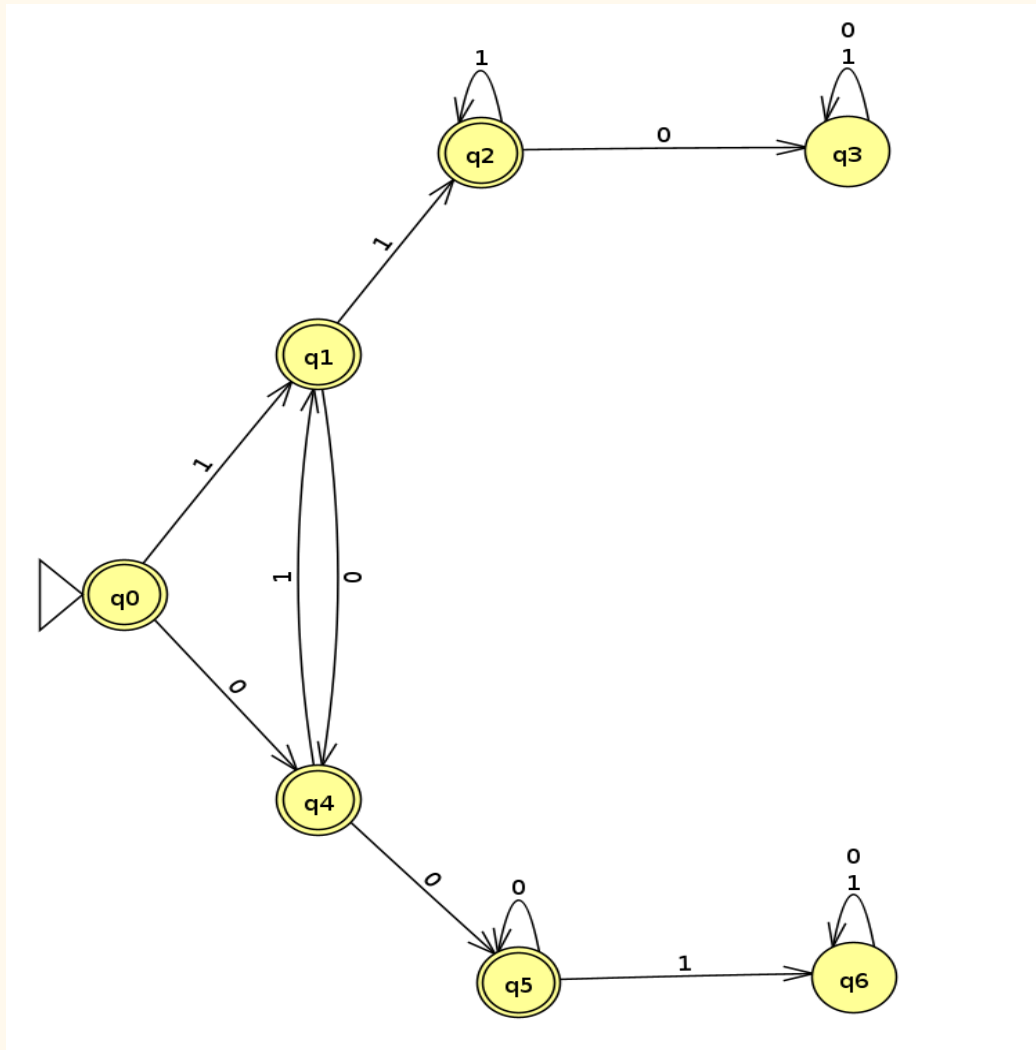
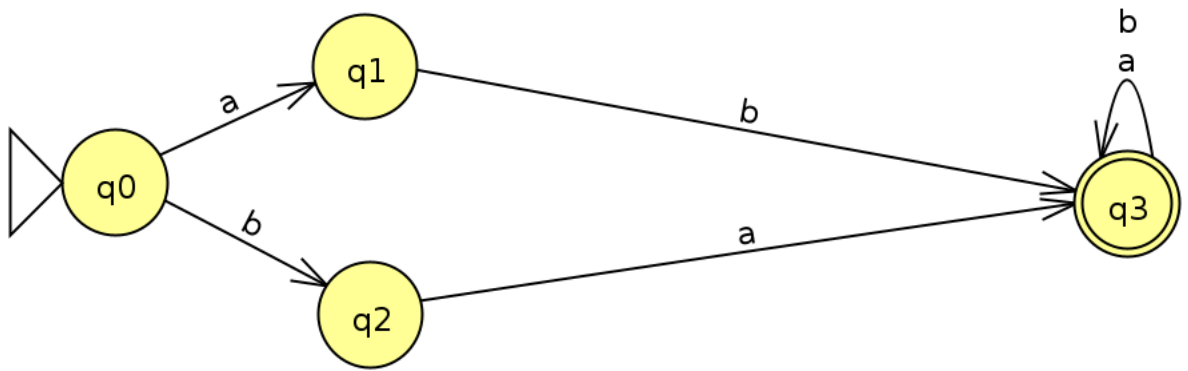


Table Text Size		
Input	Result	
111011	Reject	
11000	Reject	
001	Reject	
	Accept	
10101010	Accept	
00000000	Accept	
11111111	Accept	

Obs: inclui palavra vazia nos possíveis reconhecimentos, já que questão não especifica mínimo como as outras.

**Questão 6:**

$\{w \in (a,b)^* \mid w \text{ sempre começa com dois símbolos diferentes e } |w| > 1\}$

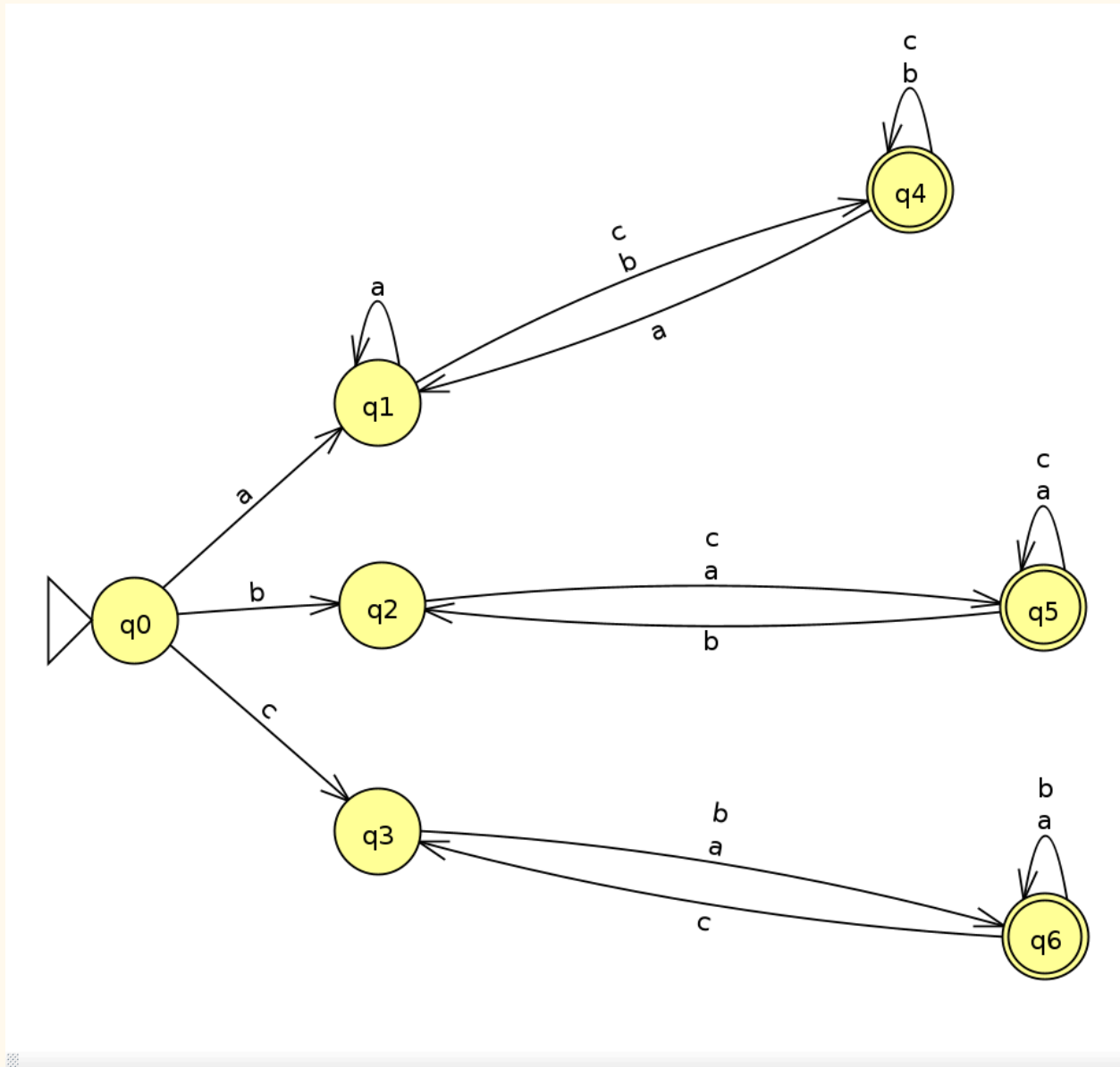


Input	Result
aa	Reject
bb	Reject
aba	Accept
bab	Accept
abaaaaa	Accept



### Questão 7:

$\{w \in (a,b,c)^* \mid \text{o primeiro símbolo de } w \text{ é sempre diferente do último}\}$



Input	Result
aa	Reject
aaaca	Reject
bcabcbab	Reject
ac	Accept
abc	Accept
acbcbcabcbab	Accept

### Questão 8:

$\{w \in (0,1)^* \mid w \text{ tem como prefixo } 00 \text{ ou } 111\}$

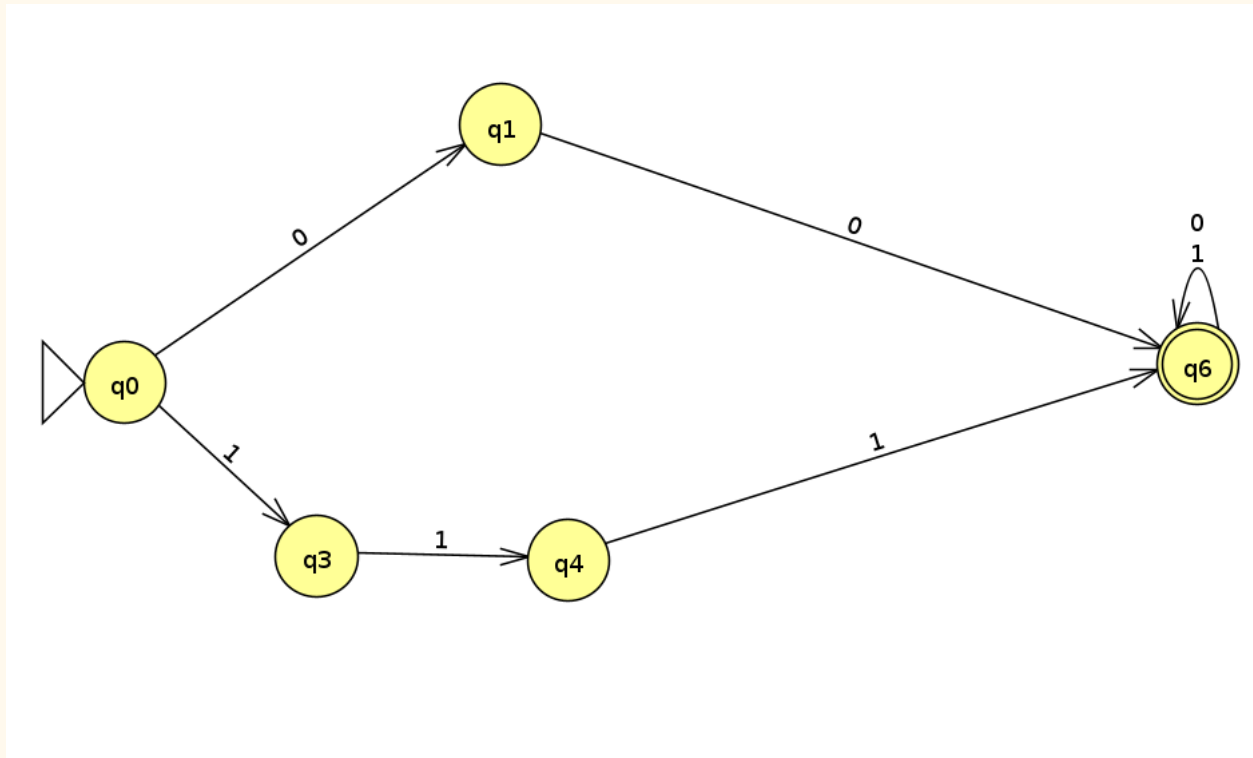


Table Text Size	
Input	Result
11	Reject
110101011111	Reject
01010101110101	Reject
00111110	Accept
11111111	Accept
00	Accept

### Questão 9:

$\{w \in (0,1)^* \mid w \text{ tem no mínimo dois 0's consecutivos após cada 1}\}$

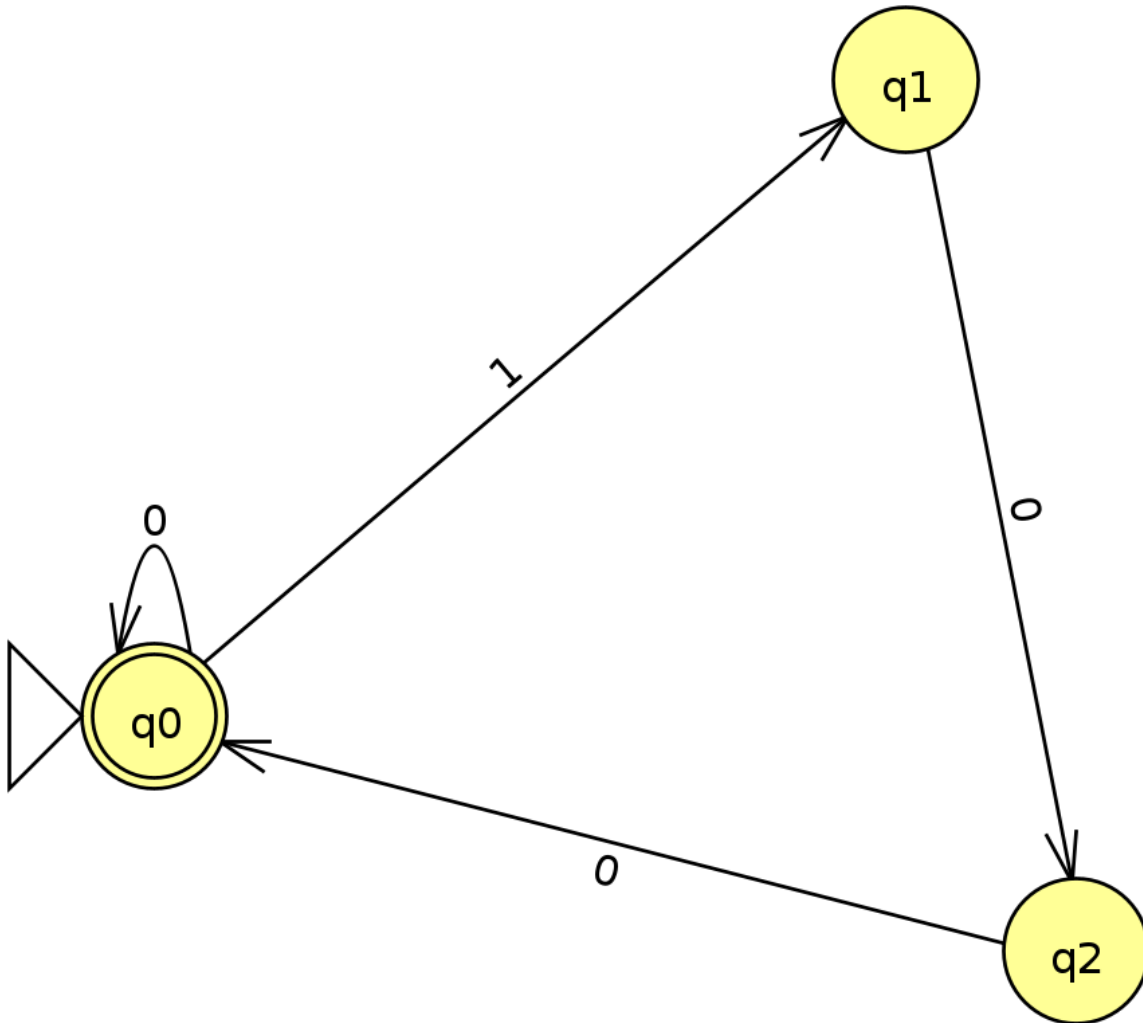


Table Text Size	
Input	Result
01	Reject
0010	Reject
01001100	Reject
000000000100	Accept
100100100100000000	Accept

### Questão 10:

$\{w \in \{0,1\}^* \mid w \text{ possui um único } 1 \text{ e um número ímpar de } 0\text{'s antes deste } 1 \text{ e um número par de } 0\text{'s após este } 1\}$

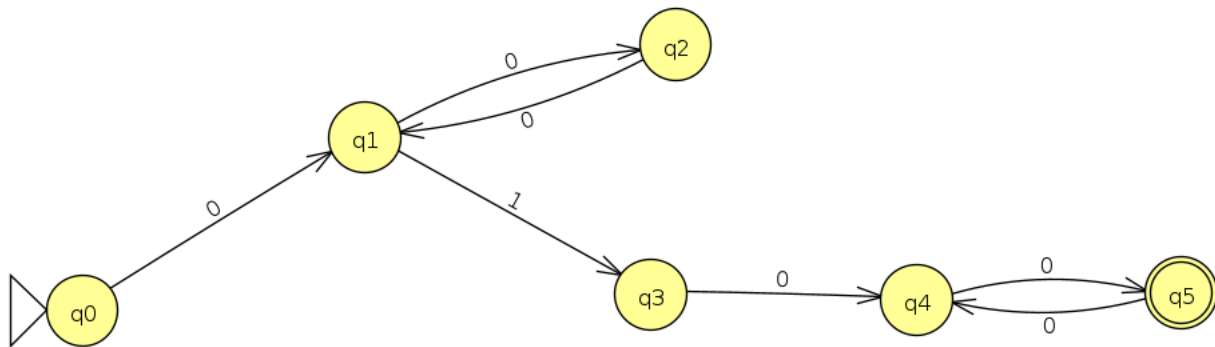


Table Text Size	
Input	Result
1	Reject
001000	Reject
0001001	Reject
0100	Accept
0000010000	Accept
0100000000	Accept

### Questão 11:

Três linguagens regulares quaisquer, definidas por você. Você deverá escrever também em português, a definição da linguagem, escrever da forma como foi feito nos dez primeiros itens aqui, fazer a máquina e fazer os testes de aceitação.

Definição : define-se linguagens regulares aquelas que podem ser lidas e reconhecidas por AFNs e AFDs.

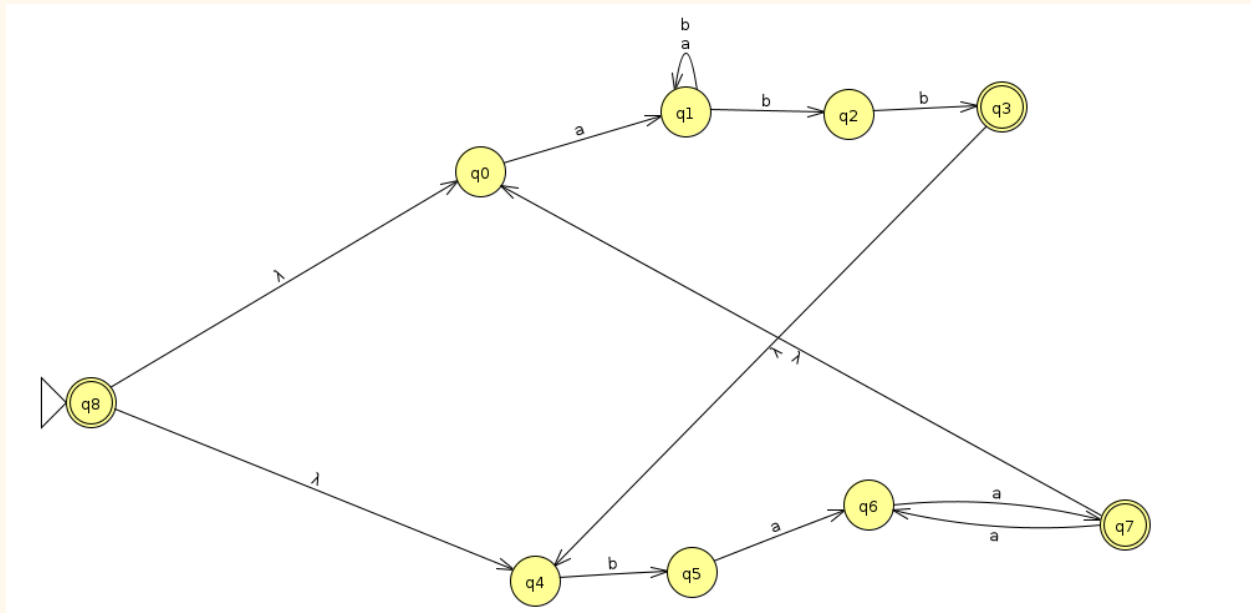
**11.1)** Sejam 3 máquinas M1, M2 e M3, que leem respectivamente as linguagens L1, L2, e L3, e estas linguagens definidas por:

L1:  $\{w \in \{a,b\}^* \mid \text{deve iniciar com o símbolo a e deve ter obrigatoriamente o sufixo bb}\}$

L2:  $\{w \in \{a,b\}^* \mid \text{palavra deve iniciar com o símbolo b e ser precedida de pares símbolos a's}\}$

L3:  $\{w \in \{a,b\}^* \mid \text{deve ser capaz de ler entradas vazias, tanto como reconhecer ou L1 e L2, como também ser capaz de reconhecer L1 seguida de L2 ou vice-versa}\}$

Autômato de L3:



Teste de L1:

Input	Result
a	Reject
aaab	Reject
abababab	Reject
aabb	Accept
aaaaaaaaabb	Accept
abababababb	Accept

Teste de L2:

Input	Result
baaa	Reject
ba	Reject
b	Reject
baa	Accept
baaaaaaaaaa	Accept
baaaa	Accept

Teste de L3:  $L3 = (L1 \cup L2) \cup (L1L2)$ :

Input	Result
abbaa	Reject
baabb	Reject
abababbaaaa	Reject
abbbbaa	Accept
baaabb	Accept
ababababaabbbaa	Accept

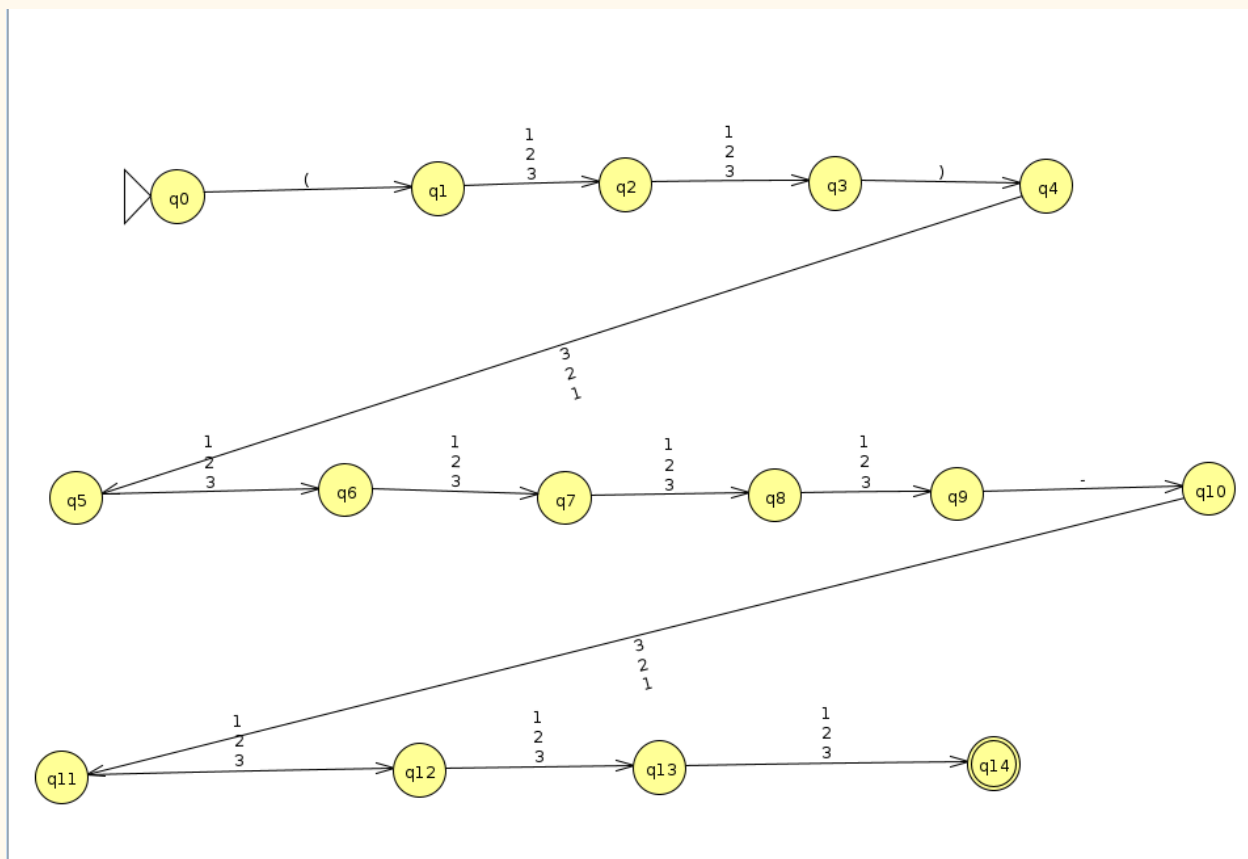
Explicação: esse autômato (máquina M3) é uma máquina capaz de ler entradas vazias, tanto como ler entrada de outras duas máquinas (M1 e M2); Por mais que isso não apresenta usos óbvios, sua utilidade pode abranger várias áreas, como por exemplo passar dados de uma máquina a outra, sendo essas máquinas interligadas por internet (IOT)

EX: relógio que lê batimentos cardíacos possui sua linguagem em seu sistema, assim como óculos por iot, que seriam capazes de serem lidos por um celular (máquina que reconheceria ambas as linguagens).

**11.2)** Existe uma máquina capaz de ler números telefônicos em formatação padrão sendo (xx)xxxxx-xxxx onde sua linguagem é definida por L:

L:  $\{w \in (1,2,3,(,),-)^* \mid w \text{ inicia com } (, \text{ após dois números recebe mais um símbolo } ), \text{ seguidos de 5 símbolos numéricos, seguido por } -, \text{ e por fim mais 4 símbolos numéricos}\}$

$$L: ((1,2,3)^{+2})(1,2,3)^5-(1,2,3)^4$$



Input	Result
(31)1231-2233	Reject
(11)8002-4922 é o funk do yudi que vai dar PS2	Reject
(21)999-209-503	Reject
(21)12332-3212	Reject
(11)23312-3311	Accept
(31)12312-2233	Accept



Explicação: Tratasse de uma abstração de reconhecimento de números telefônicos, obviamente com escalas reduzidas já que em vez de reconhecer os 9 algarismos possíveis reconhece apenas 3;

Também é possível notar que não reconhece a entrada vazia/ Palavra vazia, já que isso em sistema de reconhecimento de números telefônicos, como uma agenda, não faria sentido.

**11.3)** Suponhamos uma máquina que reconhecesse discos lidos, sabendo assim se um game é de playstation ,xbox ou nintendo , máquina essa chamada MB (maquina brainiac), então é proposto uma máquina que reconheça discos de todas as plataformas e também discos que não sejam apenas de jogos , essa máquina se chamaria MUC (máquina universal code)

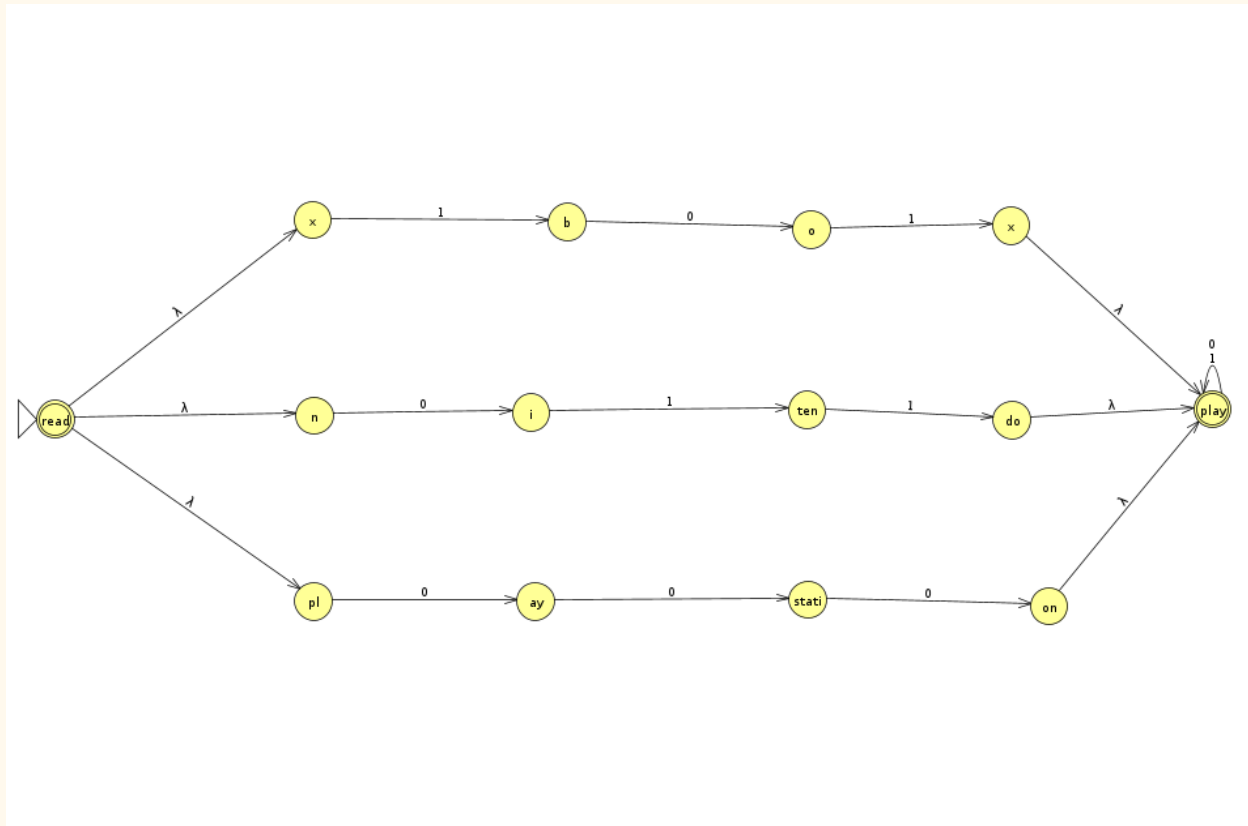
proposta: MB possui L1 como linguagem e MUC possui L2 , sendo L2 uma máquina que possui um fecho de kleene.

L1:  $\{w \in (0,1)^* \mid w \text{ tem como prefixo } 101, 011 \text{ ou } 000, \text{ mas também reconhece entrada de } \lambda\}$

L2:  $L2^*$

L2:  $\{w \in (0,1)^*\}$

MB:

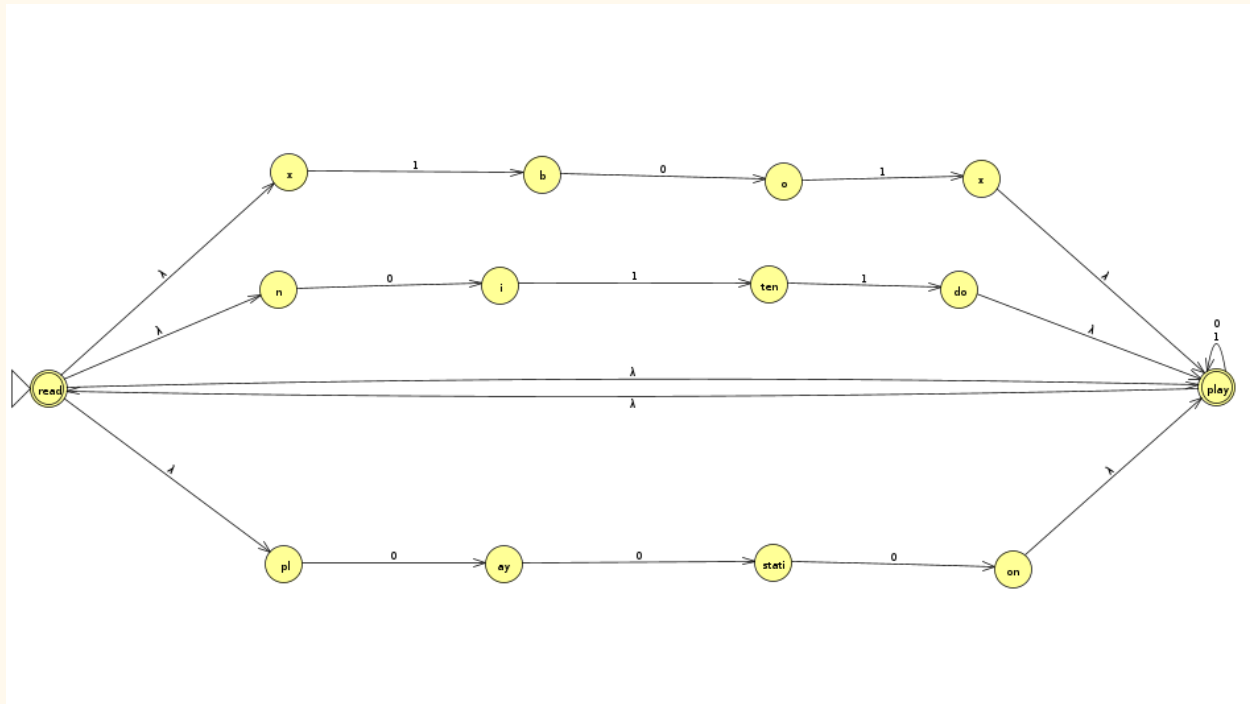


Input	Result
11101010101	Reject
001111101010	Reject
01011010101	Reject
	Accept
1010101010101110101111	Accept
0110101010101110001111	Accept
0001111111111111	Accept

Reconhece todos os prefixos que determina a plataforma, o restante da entrada seria uma abstração do binário do game.

A entrada vazia sendo reconhecida seria a abstração de que o leitor pode não ter um disco pra ler.

MUC:



Input	Result
11101010101	Accept
001111101010	Accept
01011010101	Accept
	Accept
1010101010101110101111	Accept
0110101010101110001111	Accept
000111111111111111	Accept
	Accept
10	Accept
1	Accept
011	Accept
000	Accept
1111111111111111	Accept
0000000000000000	Accept
01010101010101010	Accept
1110001110011100	Accept

Leria até mesmo sem precisar de um sufixo para definir a plataforma, mas ainda sim conseguiria ler a linguagem L1, já que esta encontra-se abrangida dentro de L2.