01 – Construa uma Máquina de Turing que recebe a codificação binária x de um número natural n e computa a codificação binária do dobro de n. Mostre a execução de M para n = 21 (binary = 10101), n = 8 (binary = 1000), n = 1 (binary = 1) e n = 0 (binary = 0). Dica: para calcular o dobro de um número binário basta adicionar o dígito 0 no final do número.

I) Lógica utilizada para resolver o problema.

O Exercício solicita que seja impresso o dobro do número binário inserido (acrescentar 0 no final). A partir desse pressuposto, a Máquina de Turing deverá ler todos os valores da esquerda para a direita e acrescentar o dígito "0" na primeira célula com valor em branco (), exibindo um erro caso não exista nenhum valor entre o início e a célula em branco (caso o número não tenha sido informado).

II) Indicação do alfabeto utilizado para construir a Máquina.

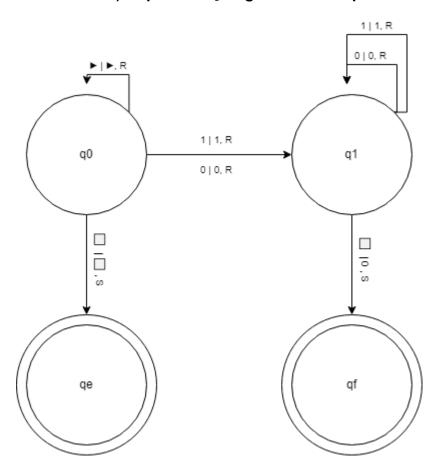
$$\Sigma = \{ \triangleright, 1, 0, \square \}$$

Q = \{q0, q1, qe, qf\}

III) Tabela de transição.

Tabela de Transições	
q0, ► = q0, ►, R	q1, ► =
q0, 1 = q1, 1, R	q1, 1 = q1, 1, R
q0, 0 = q1, 0, R	q1, 0 = q1, 0 ,R
q0, = qe, , S	q1, = qf, 0, S

IV) Representação gráfica da Máquina de Turing.



V) Execuções solicitadas.

n = 21 (binary = 10101)

•	1	0	1	0	1	
<u>q0</u> ▶	1	0	1	0	1	
•	<u>q01</u>	0	1	0	1	
•	1	<u>q10</u>	1	0	1	
•	1	0	<u>q11</u>	0	1	
•	1	0	1	<u>q10</u>	1	
•	1	0	1	0	q11	
•	1	0	1	0	1	<u>q1</u>
•	1	0	1	0	1	<u>qf0</u>
•	1	0	1	0	1	0

n = 8 (binary = 1000)

•	1	0	0	0	
<u>q0</u> ▶	1	0	0	0	
•	<u>q01</u>	0	0	0	
•	1	<u>q10</u>	0	0	
•	1	0	<u>q10</u>	0	
•	1	0	0	<u>q10</u>	
•	1	0	0	0	<u>q1</u>
•	1	0	0	0	<u>qf0</u>
•	1	0	0	0	0

n = 1 (binary = 1)

•	1	
<u>40</u> ₽	1	
•	<u>q01</u>	
•	1	<u>q1</u>
•	1	<u>qf0</u>
•	1	0

n = 0 (binary = 0)

•	0	
<u>40</u> ₽	0	
•	<u>q00</u>	
•	0	<u>q1</u>
•	0	<u>qf0</u>
•	0	0

02 – Construa uma Máquina de Turing que recebe uma cadeia binária x e verifique se x é um palíndromo (isto é, x se lido da esquerda para direita é igual a ele mesmo na ordem inversa). Demonstre a execução de M para x = 101; x = 1001 e x = 010010. Dica: percorra a cadeia com um estado de leitura e ao chegar sobre o espaço em branco, retorne com um estado de verificação. Reescreva os dígitos lidos e verificados com símbolos para controlar o que já foi lido e verificado.

I) Lógica utilizada para resolver o problema.

A atividade solicita a verificação de uma cadeia binária, informando se ela é um palíndromo. A partir disso, e do fato de não ser possível a conservação de múltiplas variáveis, a máquina de turing deverá ler a fita da esquerda para direita, guardando o primeiro dígito binário no estado e o substituindo pelo símbolo play ">", passar dígito por dígito mantendo o mesmo estado até encontrar uma célula em branco "", após isso, deverá retornar (direita para a esquerda) e ler o primeiro dígito binário, sobrescrevendo com uma casa em branco caso o valor seja o mesmo do valor salvo no estado e entrando em estado de erro caso o valor seja diferente. Caso o valor seja o mesmo salvo no estado, a máquina deverá retornar (ir para a esquerda) todas as casas diferente de > e ", usando de um estado de volta. Caso encontre o símbolo >, todo o processo será reiniciado, criando um loop que terminará sempre com o encontro de > e " nas casas centrais (êxito) ou com erro.

II) Indicação do alfabeto utilizado para construir a Máquina.

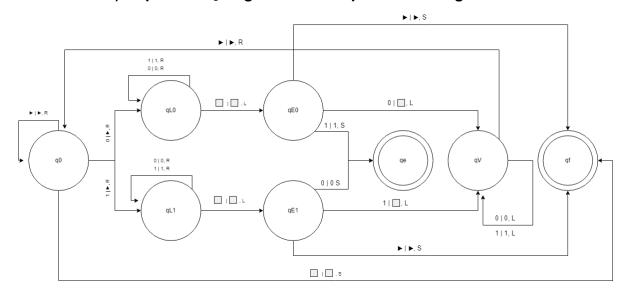
$$\Sigma = \{ \blacktriangleright, 1, 0, \bigcirc \}$$

Q = \{q0, qL0, qL1, qE0, qE1, qV, qe, qf\}

III) Tabela de transição.

Tabela de Transições						
q0, ► = q0, ►, R	qL0, ► =					
q0, 0 = qL0, ▶, R	qL0, 0 = qL0, 0, R					
q0, 1 = qL1, ▶, R	qL0, 1 = qL0, 1, R					
q0,	qL0,					
qL1, ► =	qE0, ▶ = qf, ▶, S					
qL1, 0 = qL1, 0, R	qE0, 0 = qV,, L					
qL1, 1 = qL1, 1, R	qE0, 1 = qe, 1, S					
qL1, = qE1, , L	qE0, =					
qE1, ► = qf, ►, s	qV, ► = q0, ►, R					
qE1, 0 = qe, 0, S	qV, 0 = qV, 0, L					
qE1, 1 = qV,, L	qV, 1 = qV, 1 L					
qE1, =	qV,					

IV) Representação gráfica da Máquina de Turing.



V) Execuções solicitadas.

x = 101

•	1	0	1	
<u>q0</u> ▶	1	0	1	
•	<u>q01</u>	0	1	
•	•	<u>qL10</u>	1	
•	٨	0	<u>qL11</u>	
•	•	0	1	qL1
•	٨	0	<u>qE11</u>	
•	•	<u>qV0</u>		
•	<u>qV</u> ▶	0		
•	٨	<u>q00</u>		
•	•	•	qL0	
•	•	qE0►		
•	٨	qf▶		
•	•	•		

•	1	0	0	1	
<u>q0</u> ▶	1	0	0	1	
•	<u>q01</u>	0	0	1	
•	٨	<u>qL10</u>	0	1	
•	•	0	<u>qL10</u>	1	
•	٨	0	0	<u>qL11</u>	
•	•	0	0	1	qL1
•	•	0	0	qE11	
•	•	0	<u>qV0</u>		
•	٨	<u>qV0</u>	0		
•	gV►	0	0		
•	•	<u>q00</u>	0		
•	•	•	<u>qL00</u>		
•	•	•	0	qL0	
•	•	•	<u>qE00</u>		
•	•	qV►			
•	•	•	<u>q0</u>		
•	•	•	qf		
•	•	•			

x = 010010

•	0	1	0	0	1	0	
<u>q0</u> ▶	0	1	0	0	1	0	
•	<u>q00</u>	1	0	0	1	0	
•	•	<u>qL01</u>	0	0	1	0	
•	•	1	<u>qL00</u>	0	1	0	
•	•	1	0	<u>qL00</u>	1	0	
•	•	1	0	0	<u>qL01</u>	0	
•	•	1	0	0	1	<u>qL00</u>	
•	•	1	0	0	1	0	qL0
•	•	1	0	0	1	<u>qE00</u>	
•	•	1	0	0	<u>qV1</u>		
•	٨	1	0	<u>qV0</u>	1		
•	٨	1	<u>qV0</u>	0	1		
•	٨	<u>qV1</u>	0	0	1		
•	qV►	1	0	0	1		
•	٨	<u>q01</u>	0	0	1		

•	•	•	<u>qL10</u>	0	1		
•	•	•	0	<u>qL10</u>	1		
•	•	•	0	0	<u>qL11</u>		
•	•	•	0	0	1	qL1	
•	•	•	0	0	<u>qE11</u>		
•	•	•	0	<u>qV0</u>			
•	•	•	<u>qV0</u>	0			
•	•	<u>qV</u> ▶	0	0			
•	•	•	<u>q00</u>	0			
•	•	•	•	<u>qL00</u>			
•	•	•	•	0	qL0		
•	•	•	•	qE00			
•	•	•	qV►				
•	•	•	•	q0			
•	•	•	•	qf			
•	•	•	•				

03 – Construa uma Máquina de Turing que recebe duas cadeias binárias x e y de igual número de bits, separadas por um símbolo \cup , e verifica se cada bit de x e y são iguais (operação XNOR) indicando V caso todos os bits forem iguais e F caso houverem bits diferentes. A máquina deve retornar um estado indicando V ou F. Mostre a execução de M para (x, y) = (0, 0), (x, y) = (101, 010) e (x, y) = (111, 111). Dica: percorra a primeira cadeia com um estado de leitura e a segunda cadeia com um estado de verificação. Reescreva os dígitos lidos e verificados com símbolos para controlar o que já foi lido e verificado.

I) Lógica utilizada para resolver o problema.

A atividade solicita que a máquina verifique se cada bit dos dois valores binários inseridos são iguais. A partir dessa solicitação, a máquina deverá iniciar a partir do play ">" e ler o primeiro valor informado da esquerda para a direita, salvando seu valor no estado, substituindo sua célula por ">" e seguindo para a direita, mantendo seu estado até encontrar a primeira célula contendo "\(\pi\)", após encontrar com essa casa, deverá substituir o estado de leitura do primeiro bit para o estado de escrita do bit lido, mantendo as células "\(\pi\)" com o mesmo símbolo e passando para a próxima célula até encontrar o primeiro bit do segundo valor, caso o valor não seja o mesmo do estado de escrita, deverá ocorrer o erro e a máquina será pausada. Caso encontre o mesmo valor do estado, a célula terá seu valor alterado para "\(\pi\)" e seu

estado será mudado para um estado de retorno, andando células para esquerda, os valores das células e o estado de retorno não serão alterados até encontrar a primeira célula contendo "▶", assim reiniciando o looping da máquina, que será repetido até encontrar um erro ou até que a célula "▶" encontre a célula "□", finalizando com êxito.

II) Indicação do alfabeto utilizado para construir a Máquina.

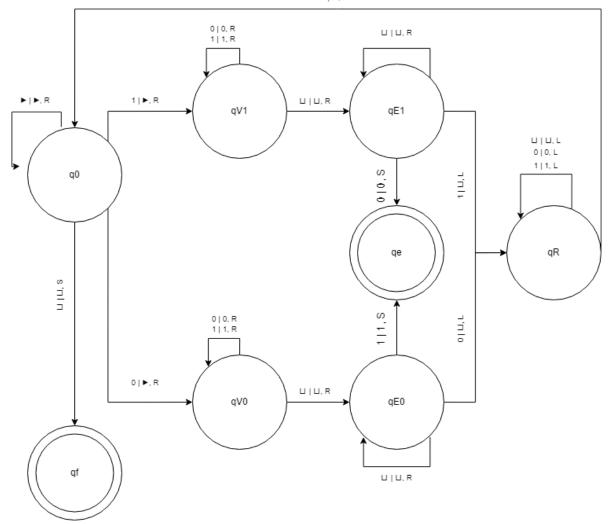
$$\Sigma = \{ \blacktriangleright, 1, 0, \sqcup, \blacksquare \}$$

Q = {q0, qV1, qE1, qR, qV0, qE0, qe, qf}

III) Tabela de transição.

Tabela de Transições					
q0, ► = q0, ►, R	qV1, ► =				
q0, 1 = qV1, ►, R	qV1, 1 = qV1, 1, R				
q0, 0 = qV0, ▶, R	qV1, 0 = qV1, 0, R				
q0, ⊔ = qf, ⊔ , S	qV1, ⊔ = qE1, ⊔, R				
q0, =	qV1, =				
qE1, ► =	qR, ► = q0, ►, R				
qE1, 1 = qR, ⊔, L	qR, 1 = qR, 1, L				
qE1, 0 = qe, 0, S	qR, 0 = qR, 0, L				
qE1, ⊔ = qE1, ⊔, R	qR, ⊔ = qR, ⊔, L				
qE1, =	qR, =				
qV0, ► =	qE0, ► =				
qV0, 1 = qV0, 1, R	qE0, 1 = qe, 1, S				
qV0, 0 = qV0, 0, R	qE0, 0 = qR, ⊔, L				
qV0, ⊔ = qE0, ⊔, R	qE0, ⊔ = qE0, ⊔, R				
qV1, =	qE0, =				

IV) Representação gráfica da Máquina de Turing.



V) Execuções solicitadas.

$$(x, y) = (0, 0)$$

•	0		0	
40 Þ	0	Ш	0	
•	q00	Ш	0	
•	•	qV0⊔	0	
•	•	Ш	qE00	
•	•	qR⊔	Ш	
•	qR▶	q0⊔	Ш	
•	•	qf⊔	Ц	
•	•	Ш	Ш	

(x, y) = (101, 010)

•	1	0	1	Ц	0	1	0	
q0 ▶	1	0	1	П	0	1	0	
•	q01	0	1	П	0	1	0	
•	•	qV10	1		0	1	0	
•	•	0	qV11		0	1	0	
•	•	0	1	qV1⊔	0	1	0	
•	•	0	1	Ш	qE10	1	0	
•	•	0	1		qe0	1	0	
•	•	0	1	Ш	0	1	0	

(x, y) = (111, 111)

•	1	1	1	Ш	1	1	1	
40 Þ	1	1	1		1	1	1	
•	q01	1	1		1	1	1	
•	•	qV11	1		1	1	1	
•	•	1	qV11	⊔	1	1	1	
•	•	1	1	qV1⊔	1	1	1	
•	•	1	1	Ш	qE11	1	1	
•	•	1	1	qR⊔		1	1	
•	•	1	qR1			1	1	
•	•	qR1	1			1	1	
•	qR►	1	1		Ц	1	1	
•	•	q01	1		П	1	1	
•	•	•	qV11			1	1	
•	>	•	1	qV1⊔	Ц	1	1	
•	•	•	1		qE1⊔	1	1	
•	•	•	1			qE11	1	
•	•	•	1		qR⊔		1	
•	•	•	1	qR⊔			1	
•	•	•	qR1	Ш			1	
•	•	qR▶	1				1	
•	•	•	q01	Ш	Ш	Ш	1	
•	•	•	•	qV1⊔			1	
•	•	•	•	Ш	qE1⊔	Ц	1	
•	•	•	•	П	Ц	qE1⊔	1	

•	•	•	>	Ш	Ш	Ш	qE11	
•	٨	•	•			qR⊔		
•	•	•	•		qR⊔	Ш	Ш	
•	•	•	•	qR⊔				
•	•	•	qR►			П	Ш	
•	•	•	•	q0⊔		Ш	Ш	
•	•	•	•	qf⊔		П	Ш	
•	•	•	•	Ш	Ш	П	П	