CS-302

HW-3

Page-1

Gorkem YAR

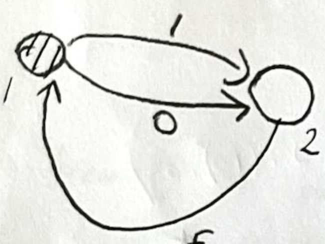
27970 ~~Gorkem~~

1)

$$E = \left(1 + (0+101)^*\right)^* = \left(1^* + (0+101)^{**}\right)^* = \left(1^* + (0+101)^*\right)^*$$

$$\Rightarrow \left(1 + (0+101)\right)^* = \left(\underbrace{(1+0)^*}_{\text{All possible strings}} + 101^*\right)^* \doteq \left((1+0)^*\right)^* = (1+0)^*$$

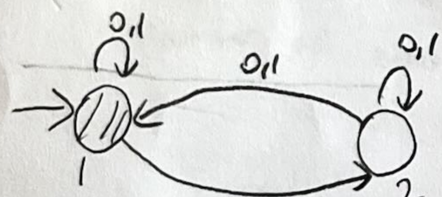
e-NFA



NFA Transition Table

Q	σ	Q'
1*	0	1,2
1*	1	1,2
2	0	1,2
2	1	1,2

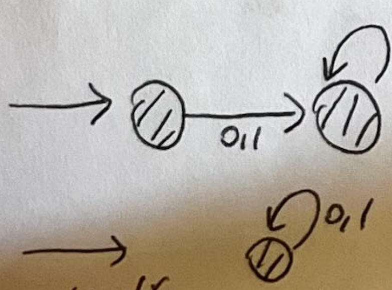
NFA



NFA To DFA Transition Table

Q	σ	Q'	
→ 1*	0	1,2*	A
1*	1	1,2*	A
1,2	0	1,2	
1,2	1	1,2	

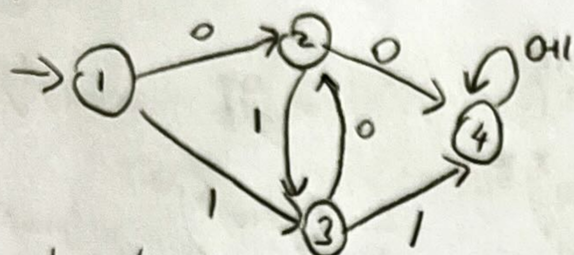
DFA



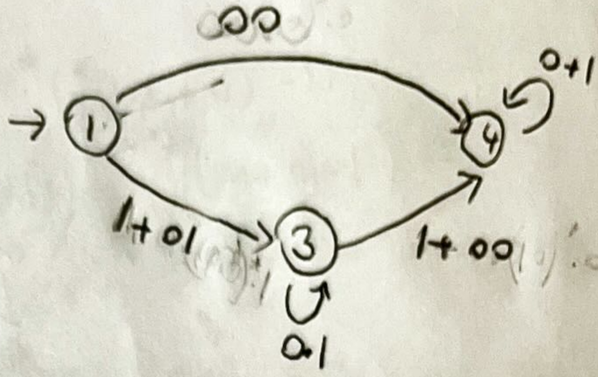
CS-302
HW-3
Page-2

Gorkem YAR
27970

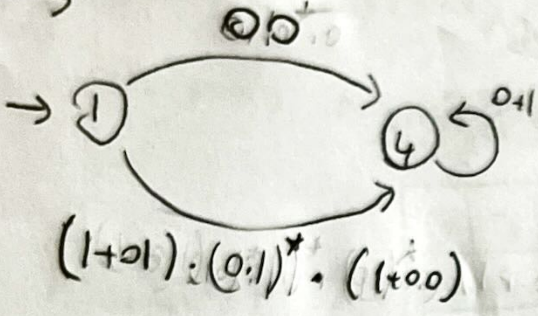
2)



eliminate 2



eliminate 3



$$RE = \left((1+01)\cdot(01)^*\cdot(1+00) + 00 \right)(0+1)^*$$

CS-302
HW-3
Page-3

Görkem YAR
27970

Question 3:

a) step 1: Let $N > 1$ be the no. of states of a DFA that accepts the language $\{ww^R \mid w \in (0+1)^*\}$

step 2: I choose $m = \underbrace{0^N \cdot 1}_{w} \cdot \underbrace{1 \cdot 0^N}_{w^R}$ $k > N$

step 3: $m = 0^N \cdot 1 \cdot 1 \cdot 0^N = x \cdot y \cdot z$, where $|x \cdot y| \leq N$ and $|y| > 0$

After setting $p = |x|$, $q = |y|$, $r = |z|$, $|xyz| = p+q+r = 2N+2$

$p+q \leq N$

$q > 0$

step 4: According to pumping Lemma $x \cdot y^j \cdot z \in L$ for any j.

if $j = 0$ than by pumping Lemma $xz \in L$ and it should obtain the $xz = w \cdot w^R$ form. But we removed some number of zeros from the first part. $xz = \underbrace{0^{N-q} \cdot 1 \cdot 1 \cdot 0^N}_{q>0}$

This is impossible for satisfying $xz = w \cdot w^R$ form

Hence $xz \notin L$, a contradiction

as a result L is not a regular language

CS-302

HW-3

Page 4

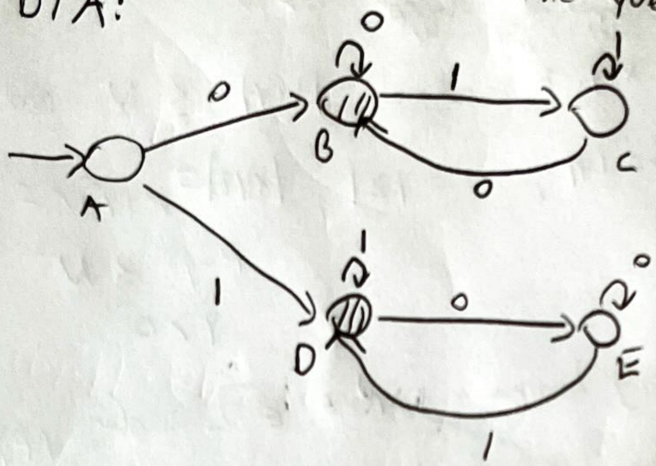
Gökhan YAR

27970 ~~Gökhan~~

Question 3-b)

This is a regular language. We cannot proof it by using Pumping Lemma. Since Pumping Lemma can only be used for proving not being a regular language. We can solve the question by 5 states.

DFA:



Final states = {B, D}

CS-302

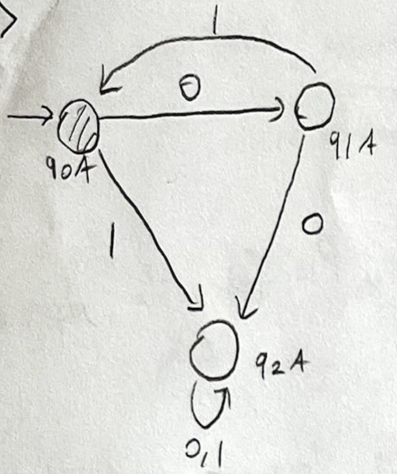
Görkem YAR
27970 ~~Görkem~~

HW-3
page-5

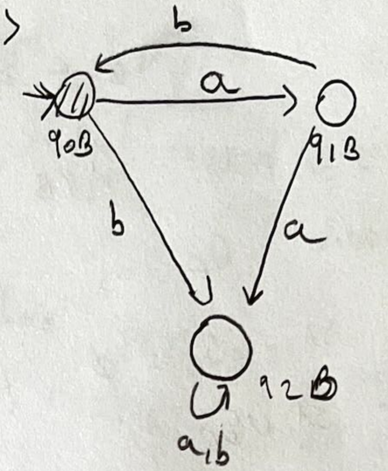
Question 4:

b) Let's start with DFA versions of $\underbrace{(01)^*}_{A}$ and $\underbrace{(ab)^*}_{B}$

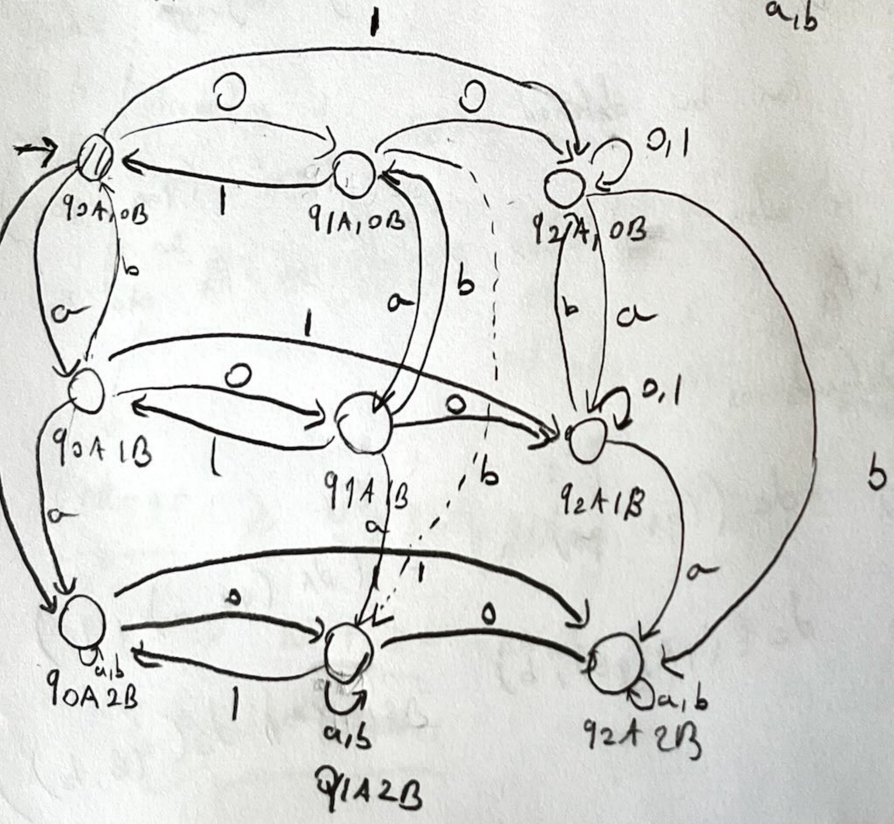
A>



B>



P>



Görkem YAR
27970 ~~Görkem~~

Question-4

a) $A = (Q_A, S_A, d_A, q_{0A}, F_A)$, $B = (Q_B, S_B, d_B, q_{0B}, F_B)$

$Q_A \cap Q_B = \phi$ $S_A \cap S_B = \phi$

$L_A \subseteq S_A^*$

$L_B \subseteq S_B^*$

The interleaving product $A \| B$ as DFA $P = (Q_P, S_P, d_P, q_{0P}, F_P)$

We can define Q_P as $Q_A \times Q_B$

Also, since $S_A \cap S_B = \phi$ the interleaving language should be $S_P = S_A \cup S_B$

The initial state can be defined as $q_{0P} = (q_{0A}, q_{0B})$

The final states is also a cross product of F_A and F_B

$F_C = F_A \times F_B$

d_C can be defined as:

$\forall a \in S_A$, $d_C((q_A, q_B), a) = (d_A(q_A, a), q_B)$

$\forall b \in S_B$, $d_C((q_A, q_B), b) = (q_A, d_B(q_B, b))$

Görkem YAR
27970

Question 6:

Exercise 4.1.2

b)

step 1: Let $N > 1$ be the no of states of a DFA that accepts the language $\{0^n \mid n \text{ is the perfect cube}\}$

step 2: I choose $w = 0^{N^3}$ that accepted by the language
$N^3 > N$ since $N > 1$

step 3: $w = 0^{N^3} = x.y.z$ where $|x.y| \leq N$ and $|y| > 0$

After setting $p = |x|$, $q = |y|$, $r = |z|$, $|xyz| = p+q+r = N^3$

$p+q \leq N$

$q > 0$

step 4: According to pumping Lemma $x.y^j z \in L$ for any j.

Let's take $j = 2$ than $x.y^2.z \in L$ should hold.

$|x.y^2.z| \geq (N+1)^3$ since it should be cube of a number and after N the smallest number is $N+1$

$\downarrow$

$\underbrace{p+2q+r}_{N^3+q} \geq N^3 + 3N^2 + 3N + 1$

$N^3 + q \geq N^3 + 3N^2 + 3N + 1$

$\underbrace{q \geq 3N^2 + 3N + 1}$

this cannot be true since $p+q \leq N \rightarrow q \leq N$

by contradiction L is not a regular language

Görkem YAR
27970 Görkem

Question-6

Exercise 4.1.2

c) Step 1: Let $N \geq 1$ be the no of states of a DFA that accepts the language $\{0^n \mid n \text{ is a power of } 2\}$

Step 2: I choose $w = 0^{2^N}$ that accepted by the language

$$2^N > N \quad \text{since } N \geq 1$$

Step 3:

$w = 2^N = x.y.z$, where $|x.y| \leq N$ and $|y| > 0$

After setting $p = |x|$, $q = |y|$, $r = |z|$

$$|xyz| = p+q+r = 2^N$$

$$p+q \leq N$$

$$q > 0$$

Step 4: According to Pumping Lemma $x.y^i.z \in L$ for any i

Let's take i as 2. Then $x.y^2.z \in L$ then $|x.y^2.z|$ should be a power of 2 since $|x.y^2.z| > |x.y.z| = 2^N$ it should satisfy the following equation:

$$|x.y^2.z| \geq 2^{N+1}$$

$$\Rightarrow p+2q+r \geq 2^N + 2^N$$

$$\Rightarrow 2^N + q \geq 2^N + 2^N$$

$$\Rightarrow \underbrace{q \geq 2^N}$$

but this cannot be true since $p+q \leq N \Rightarrow q \leq N$

so by contradiction L is not a regular language

CS-302
HW-3

Gökhan YAR
27970

Question 6

Exercise 4.1.2

h) step 1: Let $N \geq 1$ be the no of states of DFA that accepts the language $\{ w1^n$, where w is a strings of 0's and 1's of length $n. \}$

step 2: I choose $m = \underbrace{0^N}_{w} . 1^N$ that accepted by the language

step 3: $m = 0^N . 1^N = x.y.z$ where $|x.y| \leq N$ and $|y| > 0$

After setting $p = |x|$, $q = |y|$, $r = |z|$

$|x.y.z| = p+q+r = 2N$

$p+q \leq N$

$q > 0$

step 4: According to Pumping Lemma $x.y^j.z \in L$ for any j

Let's take j as 2 then $\underbrace{x.y^2}.z \in L$

since $x.y \leq N$ the looping part will be part of 0's

so $x.y^2.z = \underbrace{0^{N+q}}.1^N$

this cannot be in the form of $w.1^N$ since the length of $w > N$

Proof by contradiction

$x.y^2.z \notin L$ so L is not a regular language

Görkem YAR
27970 Görkem

Question-6

Exercise 4.3.3 : Since L is a ~~w~~regular language we can have a DFA for this language. Let's say L^c is the complement of this language. We can use reachibility algorithm in the L^c, if any initial state can reach a final state in L^c than $L \neq \Sigma^*$. The reachibility problem from particular set of states is $O(n^2)$. So we can solve this question using complement and reachibility algorithm in $O(n^2)$.

Exercise 4.3.4 : Take the product automaton of regular languages L_1 and L_2. Use reachibility algorithm. If any final state is reachable from an initial state then languages L_1 and L_2 both accept the same string. Let's say the number of states in L_1 is Q_1 and in L_2 is Q_2.

$m = Q_1 \cdot Q_2$

The reachibity algorithm from any set of states is $O(n^2)$.
So, in this case it would be $O(m^2) = O(Q_1^2 \cdot Q_2^2)$

CS-302
HW-3
Page-11

Görkem YAR
27093

Question-6

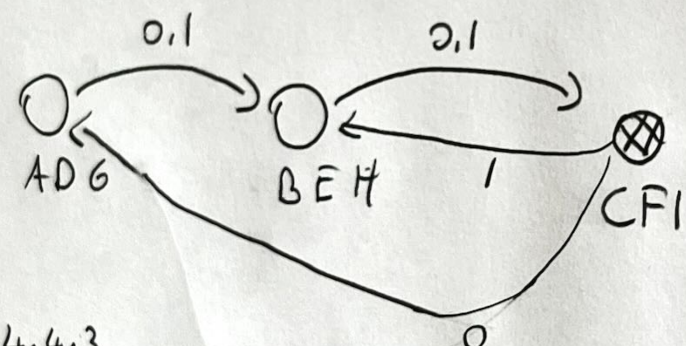
Problem 4.4.2

a)

	A	B	D	E	G	H	C	F	I
A		1		1		1	0	0	0
B			1		1		0	0	0
D				1		1	0	0	0
E					1		0	0	0
G						1	0	0	0
H							0	0	0
C									
F									
I									

→ The distinguishability table
The equivalent states
(A, D, G), (B, E H), (C, F, I)

b)



Problem 4.4.3

As it stated in the lecture Table filling algorithm can distinguish string into j-equivalent pairs in the (j-2)'th run.

firstly
(j=0 , eq pairs =2), (j=1 , eq-pairs = 3) ... ,
(j = n-2 , eq-pairs = n)

on the (n-2)'th run our algorithm can distinguish states in to n equivalent pairs.

Each run of the algorithm shows the length of the distinguishing string. Therefore, if p and q are distinguishable, a string with at most $\boxed{n-2}$ length should able to distinguish them.