CS5338

– Formal Languages Spring 2019 –

Assignment 2

Due: February 22, 2019

Submitted By

Ahmed Raza,

a\_r904,

A04775175

**1. Give a regular expression for each of the following languages. (20)**

a. The set of binary strings not containing consecutive 1’s.

(0+10)\*(E+1)

b. The set of binary strings containing exactly one instance of 11 somewhere inside.

(0+10)(0+10)\*110(0+01)\*

c. The set of binary strings with at most one pair of consecutive 1’s — i.e, if 11 is present, it can occur exactly once.

(0+10)(0+10)\*+ (0+10)(0+10)\*1100\*+(0+10)(0+10)\*1100\*0(10\*0)\*10\*

d. The set of binary strings containing at least two zeros somewhere.

(1+0)\*01\*0(1+0)\*

e. The set of binary strings containing at least two consecutive zeros and ends with 1.

(((0+11\*0)(11\*0)\*00+(0+11\*0)(11\*0)\*01(11)\*(0+10))00\*1+((0+11\*0)(11\*0)\*01(11)\*1+((0+11\*0)(11\*0)\*00+(0+11\*0)(11\*0)\*01(11)\*(0+10))1)(1+01)\*000\*1)(1+00\*1)\*

**2. Write a regular expression that accepts only valid email addresses. (20)**

^([a-zA-Z0-9\_\-\.]+)@([a-zA-Z0-9\_\-\.]+)\.([a-zA-Z]{2,5})$

**3. Let R, S and T be any three regular expressions. Prove True or False for the following. (20)**

**a. (e + R)\*S=R\*S**

LHS => (e + R)\*S

Since e+X=X (law of identity in algebraic laws of regular expressions)

So, (e+R)=R

=> (R)\*S

=>R\*S= RHS, proved

TRUE

**b. (R + S)\*S\*=(R\*S)\***

LHS=(R+S)\*S\*

Assuming that string is not empty the above expression can end with R as well as S,

Possible string can be RRR.

RHS=(R\*S)\*

Assuming that string is not empty the above expression can end with S,

Possible string can be RRRS.

FALSE

**c. S (RS+R)\*=(SR+R)\*R**

LHS= S (RS+R)\*

Assuming that string is not empty the above expression will always start with S,

Possible string can be SRSR.

RHS= (SR+R)\*R

Assuming that string is not empty the above expression can Start with R or S,

Possible string can be RRR.

FALSE

**d. (R + S)\*=R\*+S\***

FALSE

reason: RS ∈ L((R + S)\*) – L(R\* + S\*)

**4. Use Pumping Lemma to prove these languages are not regular. (20)**

**a. {0n 1 0n | n ³ 1}**

L={0n 1 0n | n ³ 1}

Proof:

If L is a regular language, then £ a constant N such that γ string w ε L, such that |w|>= N. £ a way to break w into 3 parts, w=xyz.

Such that,

1. |y|>0
2. |xy|<=N
3. For all K>=0

All strings of the form x\*y^k\*z ε L.

Let, w=0^p10^p

Clearly, |w|>=p & w ε L

By ii. X & y are composed of only zeros

By i. y=0^k, for some i>0

By iii. We can take K = 0 and resulting string will be in L

Since x\*y^0\*z should be in L

x\*y^0\*z= xz = 0^(p-k)10^p, which is clearly not in L, and this contradiction with pumping lemma proves that

L is not a regular language.

**b. {0n 1m | n £ m}**

L={0n 1m | n £ m}

Proof: If L is a regular language, then £ a constant N <= t such that γ String w ε L, |w|>=N

£ a way to break w into 3 parts w=xyz

Such that,

1. |y|>0
2. |xy|<=N
3. For all K>=0

All strings of the form x\*y^k\*z ε L.

Let w= 0^n1^m, where (n+m)>=N and m>=n

Let m=n+1 so, number of 1’s (m) >= number of 0’s (n)

Now, to choose a ‘y’ to generate a new string for all k>=0, we can’t have 0’s in ‘y’ as it will generate new strings out of pattern, and total number of 0’s will become greater then number of 1’s.

We cant have 1’s in ‘y’ as, when k=0 we get a new string in which number of 1’s wo;; be less than number of 0’s, which isn’t acceptable force this language.

Hence, L is not a regular language.

**c. Set of string of 0s and 1s whose length is a perfect square.**

Proof: If L is a regular language, then £ a constant N <= t such that γ String w ε L, |w|>=N

£ a way to break w into 3 parts w=xyz

Such that,

1. |y|>0
2. |xy|<=N
3. For all K>=0

All strings of the form x\*y^k\*z ε L.

Let w=0^N^2, here N^2 number of 0’s (or perfect square)

So, N^2>N & w ε L

‘w’ need not to contain any 1’s to prove that L is non-regular.

If k=2, then y is non-empty, which means it contains atleast one 0. And atmost N by II.

Here, xyz contains perfect square number of 0’s.

Now, making ‘y’ twice.

1<=|y|<=N => 2<=|y^2|<=2N

Now length of new string is ‘l’ letters longer, 1<=l<=N

But no longer enough to get the next perfect square after N^2 which is (N+1)^2

Since, (N+1)^2=N^2+2N+1

And after adding the at-most N, we are still short with (2N+1) in above expression.

Hence, L is proved to be not a regular language

**d. Set of string of 0s and 1s where some strings are repeated.**

Since, L is a language of set of strings of 0’s and 1’s in the form of ww. The language is clearly infinite

While selecting a string |w|>=m, holding 3 properties

String = 0^n10^m1 has length>m.

So over here |xy|<=m, |y|>0

* Xy=0^n
* Y=0^k
* W=o^(m-n)10^m1

And the string is supposed to be in L.

But it is not in the form of ww,

So, L is proved to be not a regular language

**5. Provide the proof for these theorems. (20)**

**a. If L and M are regular languages, then so is L È M.**

Proof. Let L = L(E) and M = L(F). Then L(E + F) = L È M by definition.

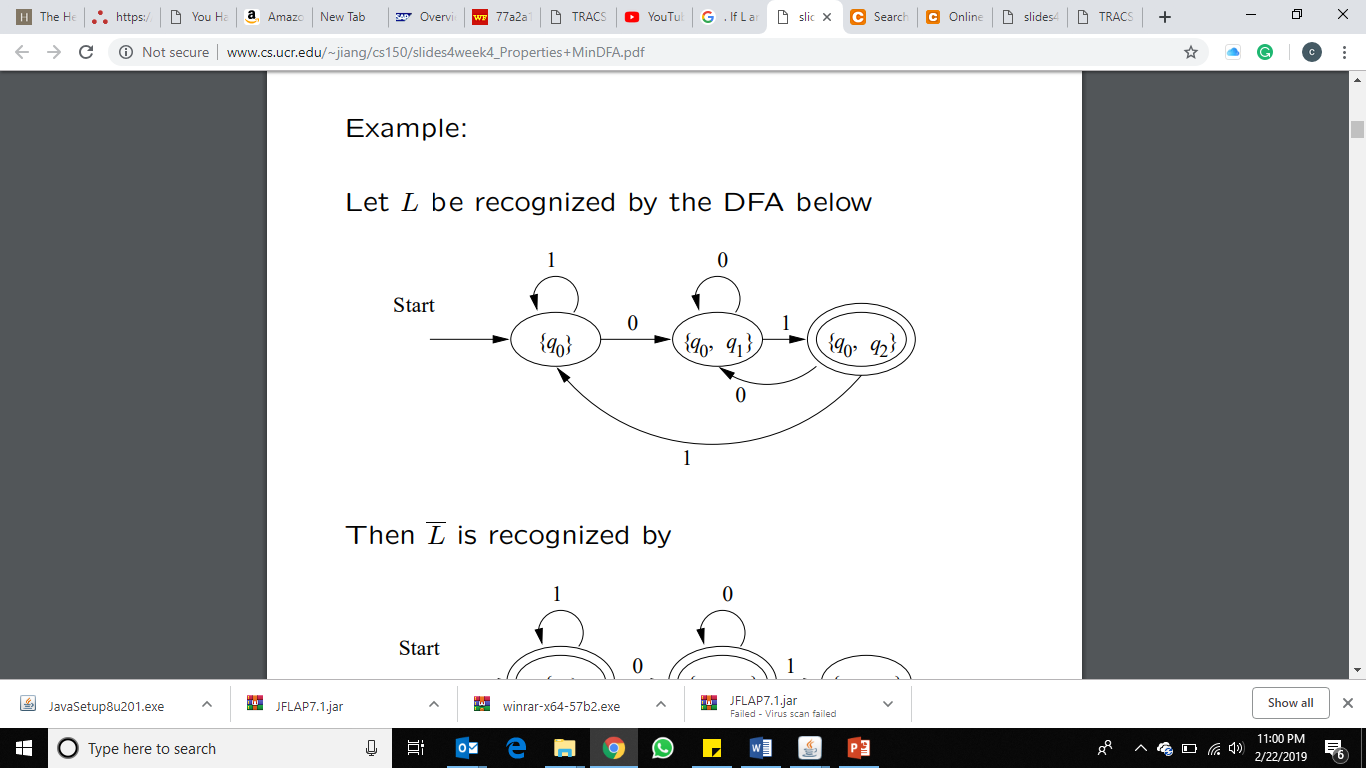
Theorem: If L is a regular language over Σ, then so is L = Σ∗ \ L.

Proof: Let L be recognized by a DFA A = (Q, Σ, δ, q0, F).

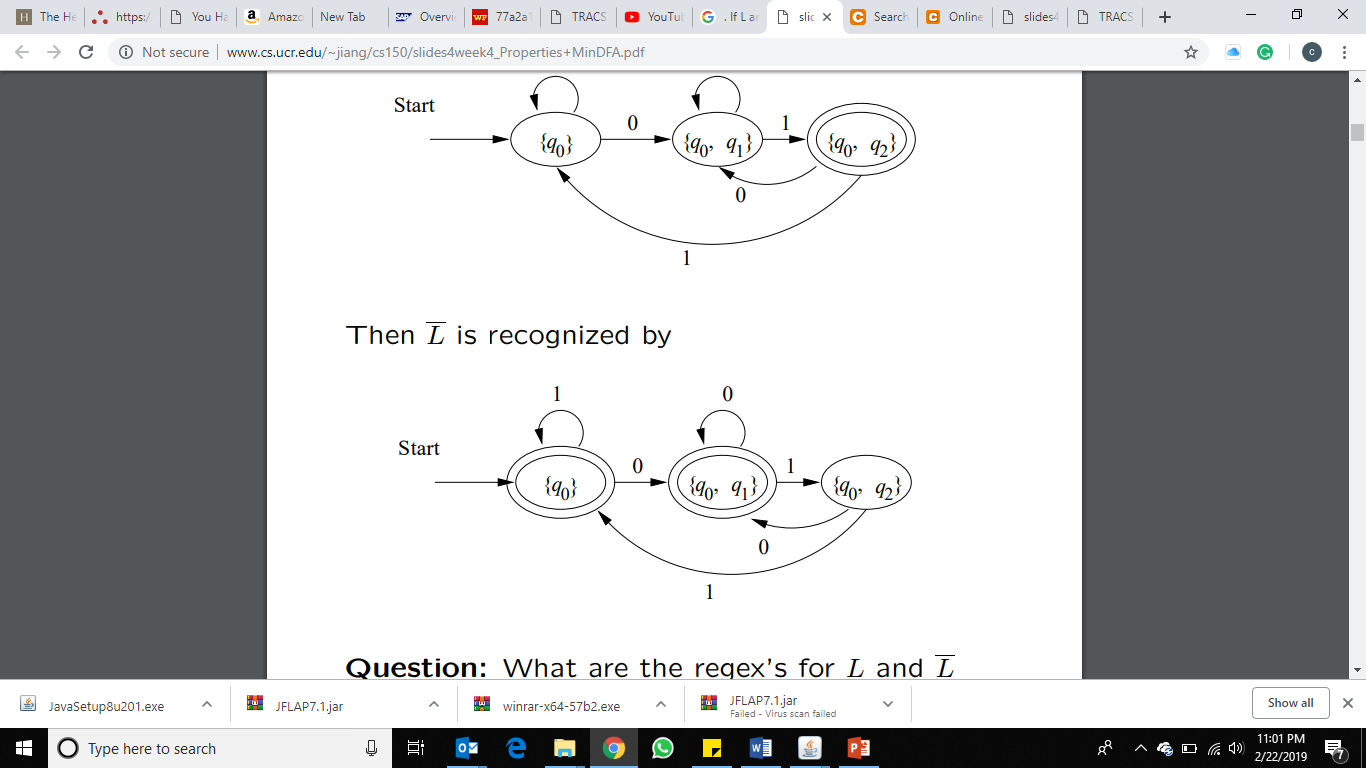
Let B = (Q, Σ, δ, q0, Q \ F).

Now L(B) = L

Example: Let L be recognized by the DFA below



Then L is recognized by



Hence, L È M is also regular.

**b. If L and M are regular languages, then so is L Ç M.**

Proof: By DeMorgan’s law L ∩ M = L ∪ M.

We already that regular languages are closed under complement and union.

Let L be the language of

AL = (QL, Σ, δL, qL, FL)

and M be the language of

AM = (QM, Σ, δM, qM, FM)

We assume that both automata are deterministic.

Formally

AL∩M = (QL ×QM, Σ, δL∩M, (qL, qM), FL ×FM),

where

δL∩M((p, q), a) = (δL(p, a), δM(q, a))

Hence, L Ç M is also regular.