TUTORIAL – 1A

(FINITE AUTOMATA & REGULAR LANGUAGES)

Problem-1

Let A,B be languages over an alphabet Σ, and
C = A−B.

Which of the following statements must be true?
a) If A and B are regular, then C is regular.
b) If A and C are regular, then B is regular.
c) If B and C are regular, then A is regular.
d) If C is regular, then A and B are regular.

Problem-2

Consider the following language over the alphabet {a,b}:

L_1 = { x $\in$ {a,b}* | x starts with *ab* but does not end with *ab* }.

a) Write a regular expression for L_1.

b) Design a DFA for L_1.

Problem-3

The language, L_2 = { uvv^Rw | $u,v,w \in \{a,b\}+$ } is regular.

Here, v^R is the reverse of v.

Answer the following:

a) Design a regular expression whose language is L_2.

b) Convert the regular expression of Part (a) to an equivalent NFA.

c) Convert the NFA in Part (b) to an equivalent DFA.

Problem-4

Construct a regular expression over the alphabet {a,b,c} for:

L_3 = { x $\in$ {a,b,c}* | x has 4n+1 b's for some integer n >= 0 }.

Then, do the following:

a) Construct an NFA from it.

b) Then, build the equivalent DFA.

Problem-5

Design a DFA for the following language:

L_4 = { x ∈ {a, b}* | #a(x) ≡ 1 (mod 2) and #b(x) ≡ 2 (mod 3) }.

where, #a(x) and #b(x) denote the number of a's & b's in x.

What changes do you make in the automaton if the language is changed as follows:

L'_4 = { x ∈ {a, b}* | #a(x) ≡ **0** (mod 2) and #b(x) ≡ **0** (mod 3) }.

Problem-6

Consider the language,

L_5 = { x ∈ {a, b}* | x ends with 3 consecutive b's }.

Answer the following.

(a) Design a ε-NFA and then an equivant NFA for L_5.

(b) Using subset construction, construct an equivalent DFA for the NFA from Part-(a).

Problem-7

Two regular expressions over the same alphabet are called equivalent if they generate the same language.

Prove/Disprove the equivalence of the following pairs of regular expressions over the alphabet {a,b}.

i) (ab+a)*a and a(ba+a)*

ii) (ab*a+ba*b)* and (ab*a)*+(ba*b)*

THANK YOU !