CS3062 Theory of Computing
Assignment 2

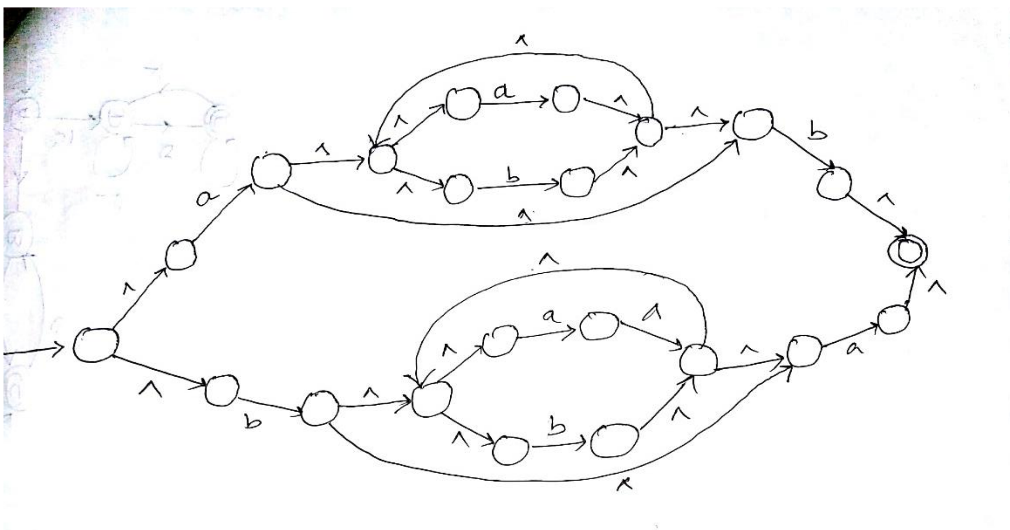
P.Thineshan
160633A

1. L is the language over {a, b} such that, for every string in L, if it starts with a then it ends with b and if it starts with b then it ends with a.

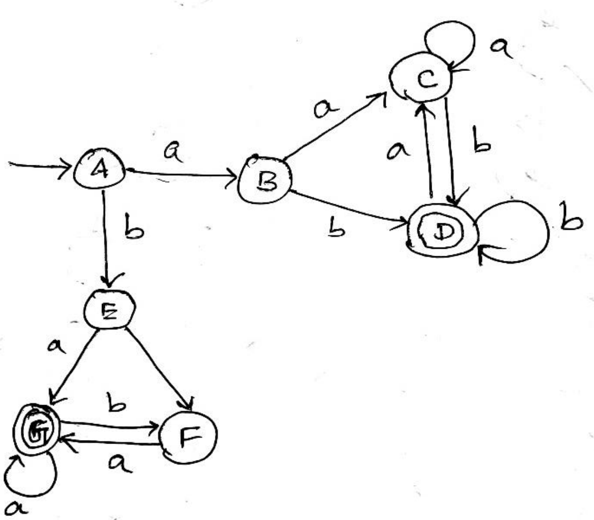
(a) Give a regular expression that represents the language L.

a(a+b)*b + b(a+b)*a

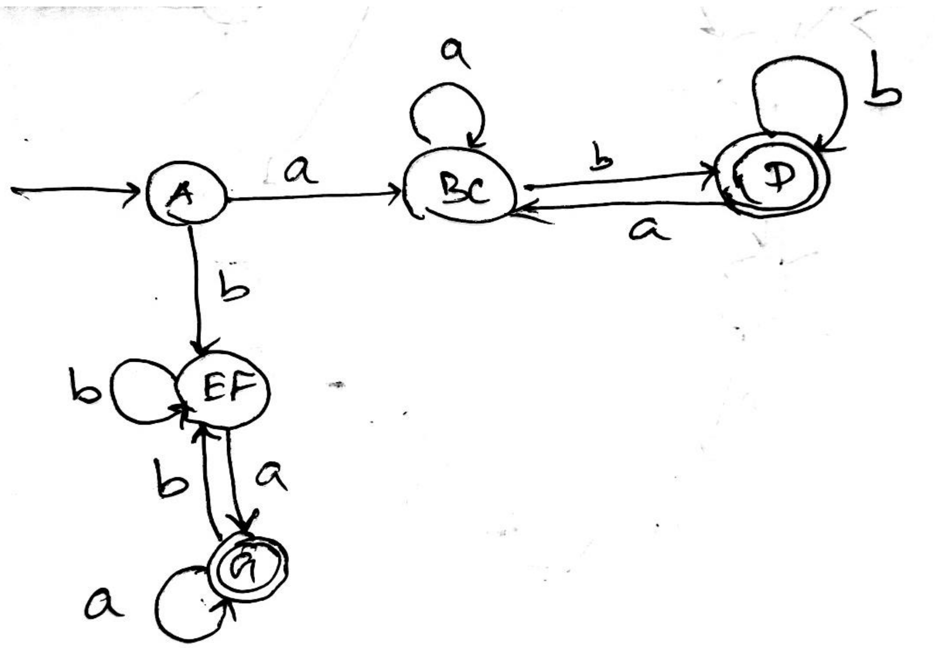
(b) Construct NFA-Λ for the above expression using thompson's construction.



(c) Provide DFA corresponding to this language.



(d) Minimize the states of above DFA.



2. S → Ab | aaB, A → a | Aa, B → b

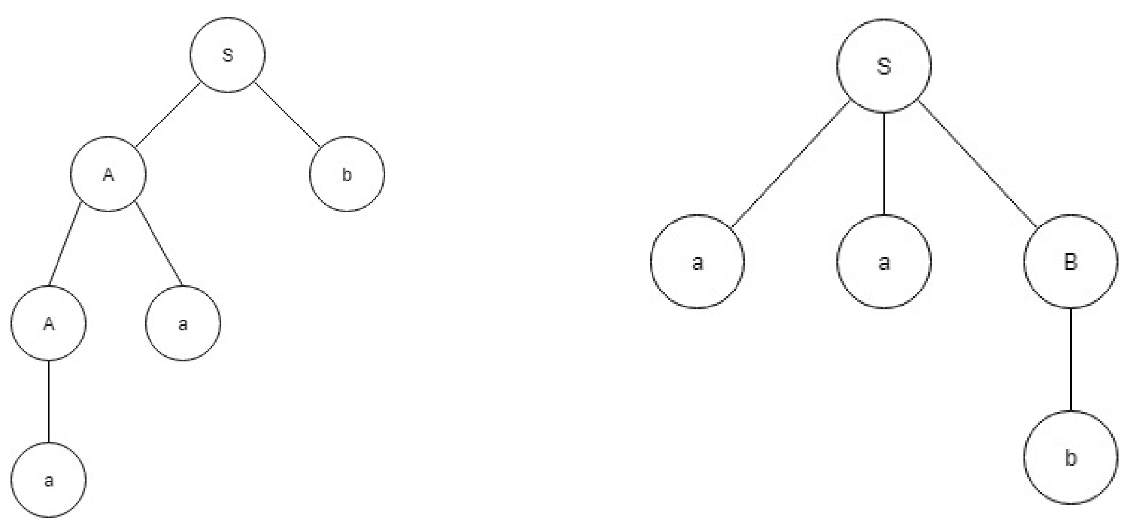
(a) S → Ab S → aaB

S → Aab S → aab

S → aab

string 's' that has at least two leftmost derivations = aab

(b) Two derivation trees for the string 's'

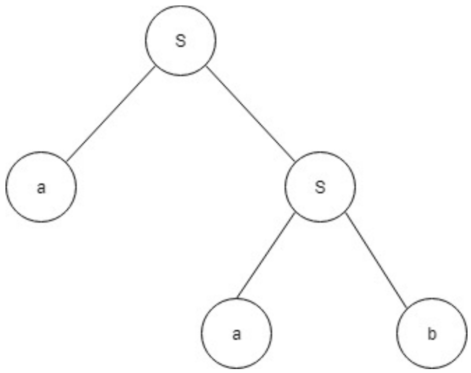


(c) Equivalent unambiguous context free grammar

$$S \rightarrow aS \mid ab$$

(d) $S \rightarrow aS$

$S \rightarrow aab$



3. Construct a PDA (non-determinism is allowed) for each of the following languages.

(a)

$L = \{ a^i b^j c^k \mid i, j, k \geq 0 \text{ and } i+j = k \}$

$\delta(q_0, a, z_0) = (q_0, az_0)$

$\delta(q_0, a, a) = (q_0, aa)$

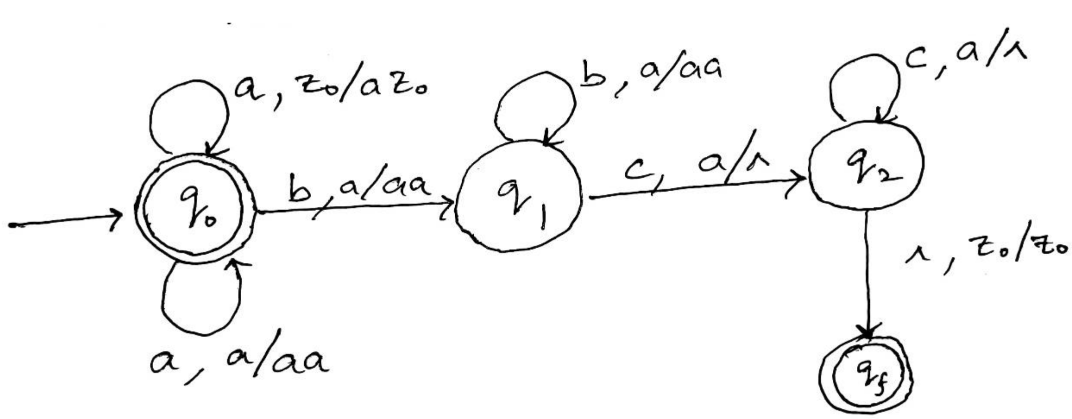
$\delta(q_0, b, a) = (q_1, aa)$

$\delta(q_1, b, a) = (q_1, aa)$

$\delta(q_1, c, a) = (q_2, \wedge)$

$\delta(q_2, c, a) = (q_2, \wedge)$

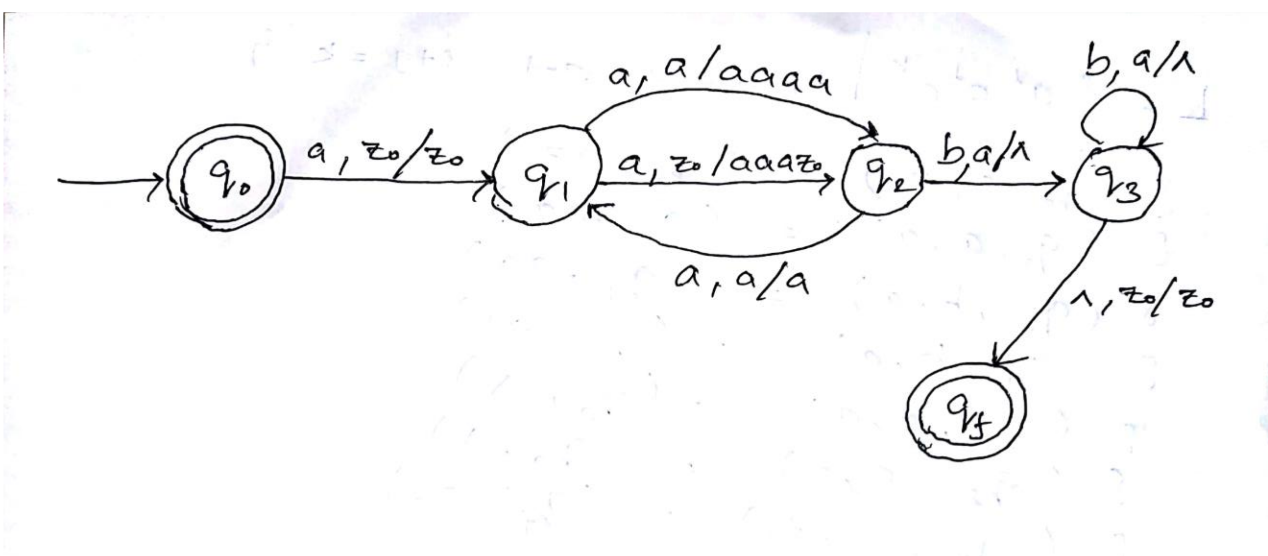
$\delta(q_2, \wedge, z_0) = (q_f, z_0)$



(b)

$$L = \{a^{2n} b^{3n} \mid n \geq 0\}$$

$$\delta(q_0, a, z_0) = (q_1, z_0)$$
$$\delta(q_1, a, z_0) = (q_2, aaaz_0)$$
$$\delta(q_2, a, a) = (q_1, a)$$
$$\delta(q_1, a, a) = (q_2, aaaa)$$
$$\delta(q_2, b, a) = (q_3, \Lambda)$$
$$\delta(q_3, b, a) = (q_3, \Lambda)$$
$$\delta(q_3, \Lambda, z_0) = (q_f, z_0)$$

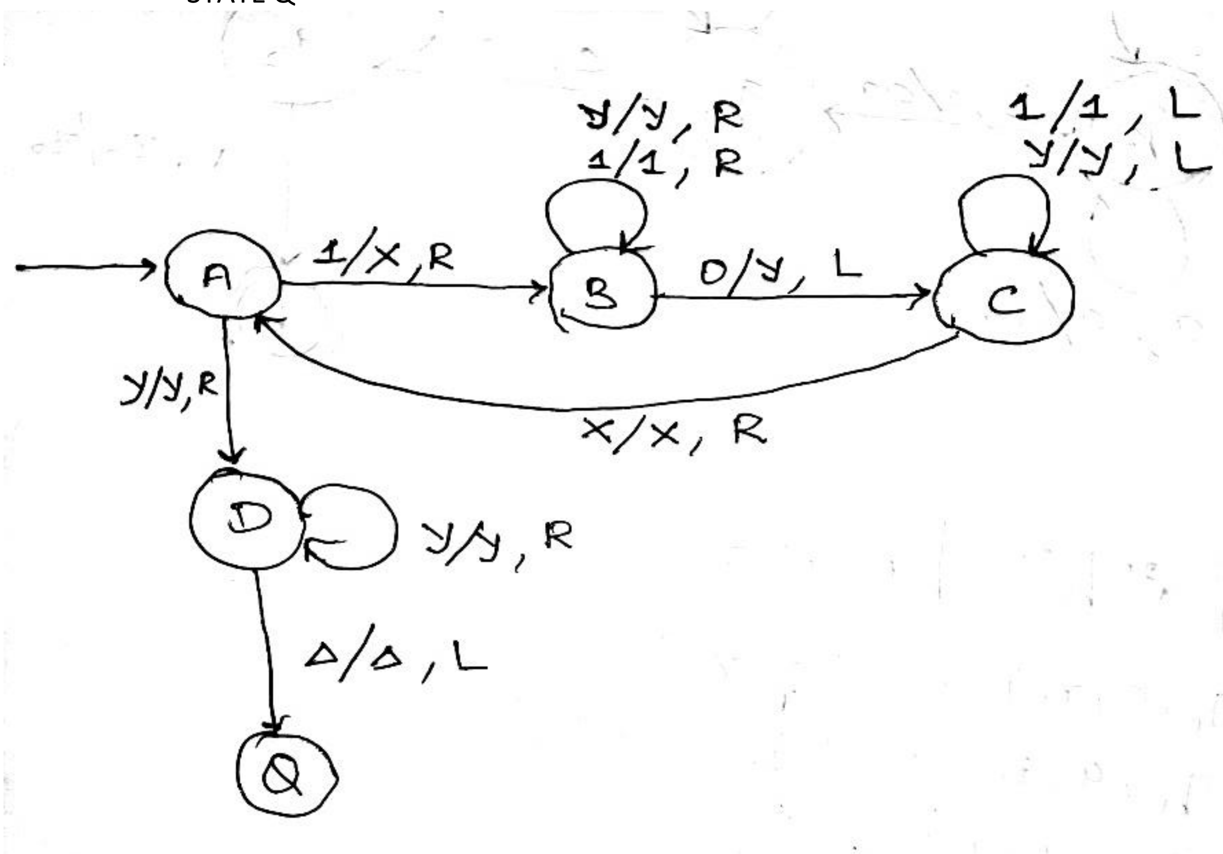


4. Design a Turing Machine that decides the language L := $\{1^n 0^n \mid n >= 1\}$.

Assumption: We will replace 1 by X and 0 by Y
Approach used:
First replace a 1 from front by X, then keep moving right till you find a 0 and replace this 0 by Y and move left. Now keep moving left till you find a X. When you find it, move a right, then follow the same procedure as above.

A condition comes when you find a X immediately followed by a Y. At this point we keep moving right and keep on checking that all 0's have been converted to Y. If not then string is not accepted. If we reach Δ then string is accepted.

- **Step-1:**
  Replace 1 by X and move right, Go to state B.
- **Step-2:**
  Replace 1 by 1 and move right, remain on same state
  Replace Y by Y and move right, remain on same state
  Replace 0 by Y and move right, go to state C.
- **Step-3:**
  Replace 1 by 1 and move left, remain on same state
  Replace Y by Y and move left, remain on same state
  Replace X by X and move right, go to state A.
- **Step-4:**
  If symbol is Y replace it by Y and move right and Go to state D
  Else go to step 1
- **Step-5:**
  Replace Y by Y and move right, remain on same state
  If symbol is Δ replace it by Δ and move left, STRING IS ACCEPTED, GO TO FINAL STATE Q



5.

For each part identify whether the given language L is context-free or non-context-free language. Prove your answer.

a. L = $\{a^n b^j \mid n \le j^2$ and n, j ∈ Z}

b. L = $\{a^n \mid n$ is prime}

c. L = $\{a^n b^j c^k \mid k = j*n$ and n, j, k ∈ Z}

(a) L = $\{a^n b^j \mid n \le j^2$ and n, j ∈ Z}

This is not context-free

Assume for contradiction that L is a context-free language. We apply the pumping lemma. Let m be the parameter of the pumping lemma. We choose to pump the string $a^{m^2}b^m \epsilon L$. We have that $a^{m^2}b^m$ = uvxyz, with |vxy|<= m and |vy|>=1.

We examine all the possible cases for the position of string vxy. First we note that the string v cannot span simultaneously both a^{m^2} and b^m , since if we pump up v (repeat v), the resulting string is not in the language (a's are mixed with b's). Therefore, it must be that v is either within a^{m^2} or within b^m . The same holds for y. Below are the rest of the cases. Notice that in all cases we obtain a contradiction, and therefore the language **L is not context-free**.

(i) v is within a^{m^2} and y is within b^m . We have that v = a^k and y = b^l , with 1 <=k + l <= m (since |vxy| <= m and |vy|>=1). Consider the case where l >=1. From the pumping lemma we have that $uv^0xy^0z \epsilon L$. Therefore, $a^{m^2-k}b^{m-l} \epsilon L$, and thus, it must be that m^2-k <= $(m-l)^2$. However, this is impossible since:

$(m-l)^2$ <= $(m-l)^2$ (since l>=1)

= m^2 +2m + 1

< m^2-k (since k <= m)

Consider now the case where l = 0. It must be that k >= 1 (since k + l >= 1). From the pumping lemma we have that $uv^2xy^2z \epsilon$ L. Therefore, $a^{m^2+k}b^m \epsilon$ L, which is impossible since m^2+k>m^2

(ii) v and y are within a^{m^2}. If we pump up v and y(repeat them) we obtain a string of the form , $a^{m^2+k}b^m$, with k>=1, which obviously is not in the language

(iii) v and y are within b^m . If we pump down v and y (remove them) we obtain a string of the form $a^{m^2}b^{m-k}$, with k >= 1, which obviously is not in the language.

(b) $L = \{a^n \mid n \text{ is prime}\}$

This is not context-free

Assume L is a context-free language. We apply the pumping lemma. Let m be the parameter of the pumping lemma. Let p be a prime such that p >= m.

We choose to pump the string $a^p \epsilon L$. Since a^p = uvxyz, we have that $v = a^k$ and $y = a^l$, with k+l >= 1 (since |vy|>=1). From the pumping lemma we have that $uv^{1+p}xy^{1+p}z \epsilon$ L, and therefore $a^{p+kp+lp} \epsilon$ L. Subsequently, $a^{p(1+k+l)} \epsilon$ L., which is impossible since p(1+k+l) is not a prime. Thus, we have a contradiction and the language **L is not context-free**.

(c) $L = \{a^n b^j c^k \mid k = j*n \text{ and } n, j, k \in Z\}$

This is not context-free

Assume L is a context-free language. We apply the pumping lemma. Let m be the parameter of the pumping lemma. Wechoose to pump the string $a^m b^m c^{m^2} \epsilon$ L. We have that $a^m b^m c^{m^2}$ = uvxyz, with |vxy|<= m and |vy|>=1.

We examine all the possible cases for the position of string vxy. First we note that the string v cannot span simultaneously both a^m and b^m, since if we pump up v (repeat v), the resulting string is not in the language (a's are mixed with b's). Similarly, v cannot span both b^m and c^{m^2}. Therefore, it must be that v is either within a^m or b^m or c^{m^2}. The same holds for y. Below are the rest of the cases. Notice that in all cases we obtain a contradiction, and therefore the **language L is not context-free**.

(i) v is within b^m and y is withinc^{m^2}. We have that $v = b^k$ and $y = c^l$, with 1 <=k + l <= m (since |vxy| <= m and |vy|>=1).
Consider the case where k >=1. It must be that l<m (since k+l<=m). From the pumping lemma we have that $uv^0xy^0z \epsilon L$. Therefore, $a^m b^{m-k} c^{m^2-l} \epsilon L$, and thus, it must be that m.(m-k) = m^2 - l . However, this is impossible since:

m.(m-k) = m^2 -mk

<= m^2 -m (since k>=1)

< m^2-l (since l < m)

Consider now the case where k = 0. It must be that l >= 1 (since k + l >= 1). From the pumping lemma we have that $uv^0xy^0z \epsilon$ L. Therefore, $a^m b^m c^{m^2-l} \epsilon$ L, which is impossible since m.m not equal m^2 -l

(ii) v and y are within b^m .
If we pump down v and y (remove them) we obtain a string of the form $a^m b^{m-k} c^{m^2}$, with k >= 1, which obviously is not in the language.

(iii) v and y are within c^{m^2}. If we pump up v and y(repeat them) we obtain a string of the form , $a^m b^{m-k} c^{m^2-k}$, with k>=1, which obviously is not in the language

(iv) v and y are somewhere within $a^m b^m$
Similar to cases (ii) and (iii)