Model of computation assignment 2
Gong lehan 925775

1.

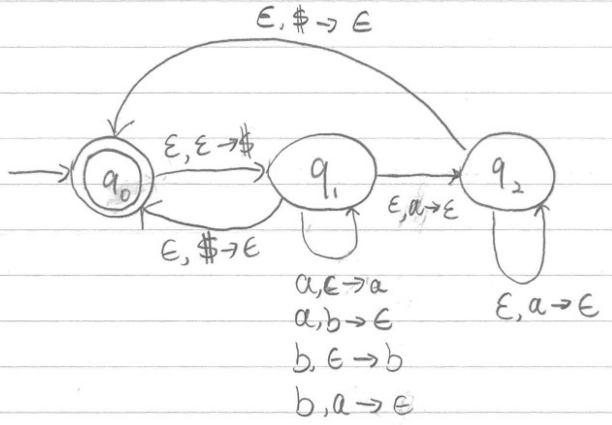
a. S → $a \mid b \mid aAa \mid bBb$

F → $a \mid b$
A → $FAF \mid a$
B → $FBF \mid b$

b. S → $aSa \mid Ab \mid bA$

A → $aA \mid a$

2 a).



Date:
Page:

2.b) part 1. prove ($L(G) \subseteq C$), C : $\{a,b\}^*$ a string that no.a $\geq$ no.b
(no negative no.a or no.b)

Induction hypothesis : $P(S) ::= (S \in C)$

base case 1 : ($S = \epsilon$), 0 no.a and 0 no.b
base case 2 : ($S = a$), 1 no.a and 0 no.b
$\therefore$ $P(\epsilon)$ and $p(a)$ are true.

Inductive case 1 : $f = (aSb)$, by induction hypothesis, $S \in C$
no.a in f = no.a in S + 1
no.b in f = no.b in S + 1
Since no.a in S $\geq$ no.b in S ($S \in C$)
no.a in S + 1 $\geq$ no.b in S + 1
$\therefore$ no.a in f $\geq$ no.b in f
$P(f)$ is true

Inductive case 2 : $f = (bSa)$, same prove as above, it only switch a and b position, so no.a and no.b is the same.
$\therefore$ $P(f)$ is true.

Inductive case 3 : $f = (SS)$, by induction hypothesis, $S \in C$
no.a in f = (no.a in S) $\times 2$
no.b in f = (no.b in S) $\times 2$
since no.a in S $\geq$ no.b in S ($S \in C$)
(no.a in S) $\times 2$ $\geq$ (no.b in S) $\times 2$
$\therefore$ no.a in f $\geq$ no.b in f
$P(f)$ is true.

Part 1 proved : $L(G)$ is in C ($L(G) \subseteq C$)

Date:
Page:

2. b) part 2. prove $C \subseteq L(G)$, C: a string that no.a $\geq$ no.b $\{a,b\}^*$

(no. a and no. b can't be negative)

Let n be the length of string in C

S(n) be the string set that with length n, and in C,

Induction hypothesis: all strings in S(n) also in L(G)

base step: when $n=0$, $S(n) = \{\epsilon\}$, (ϵ is in L(G))

when $n=1$, $S(n) = \{a\}$, (a is in L(G))

$\therefore$ $S(n) \subseteq L(G)$ while $n \leq 1$

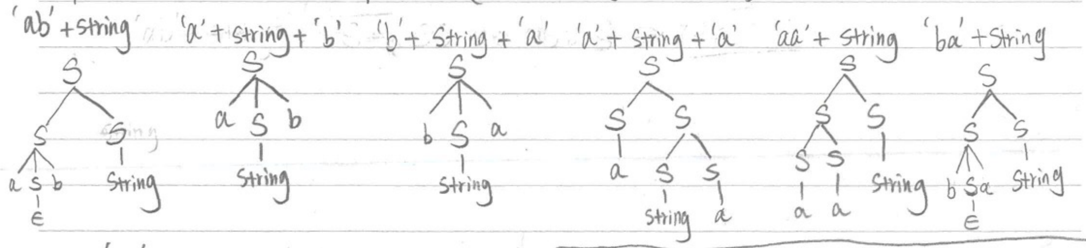
when $n > 1$:

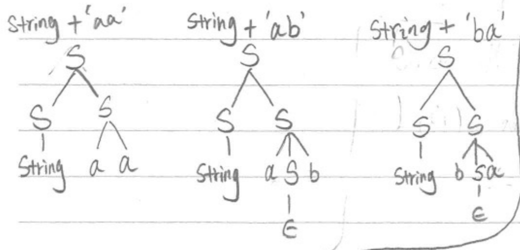
S(n) :{ for each string in S(n-2):

'ab' + string, 'ba' + string,
'a' + string + 'b',
'b' + string + 'a',
'a' + string + 'a',
'aa' + string,
string + 'aa',
string + 'ab', string + 'ba'

}

express each statement in the loop as L(G): (string is in S)





Since all 9 of the transforms of string is in the L(G), and base case, n=0, 1 also in the L(G).

$\therefore$ $C \subseteq L(G)$

Since, we prove $C \subseteq L(G)$ and $L(G) \subseteq C$.

now we know that the L(G) generates C.

Date: . .
Page:

Challenge 3.

a) Yes, regular language closed in concatenation,

therefore $R \circ R$ is regular language if R is,

and $R \circ R \circ R = (R \circ R) \circ R$ is regular language

Since $(R \circ R)$ is Regular language.

and R is also Regular language.

b) No, function triple (L) concateate the string in L two times.

if R is a regular language, triple (R)

Let R be $\{a^m b^l \mid m, l \geq 0\}$,

triple(R) become $\{a^m b^l a^m b^l a^m b^l \mid m, l \geq 0\}$

assume triple (R) is a regular language.

and p is the pumping length.

Consider $a^p b^l a^p b^l a^p b^l \in$ triple (R) with length greater than P.

By the pumping lemma, $a^p b^l a^p b^l a^p b^l = xyz$, with xy^iz in triple(R) for all $i \geq 0$, $y \neq \epsilon$, and $|xy| \leq P$.

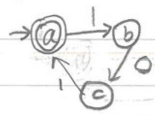
Since $|xy| \leq P$, y consists entirely of as.

But then $xyyz \notin$ triple(R), a contradiction.

$\therefore$ even R is a regular language, Triple (R) not necessarily regular.

Challenge 3.

c) Let $R = \{(101)^*\}$ which is a regular language with DFA



In snip(R), since R's length always $3K$, $K \in \mathbb{N}$, (possibly 0) and base case of R is a palindrome (101) with length 3.

therefore $\text{snip}(R) = \{w \mid w \in (101)^m (11 \cup 1001 \cup \epsilon)(101)^m, m \geq 0, m \in \mathbb{N}\}$

assume snip(R) is regular, and P is the pumping length

consider $(101)^P 1001 (101)^P \in \text{snip}(R)$ with length greater than P

By pumping lemma, $(101)^P 1001 (101)^P = xyz$, with xy^iz in snip(R) for all $i \geq 0$, $y \neq \epsilon$, and $|xy| \leq P$

Since $|xy| \leq P$, and $y \neq \epsilon$.

No matter how you pump, the '1001' part of the string with be moved and no longer in the middle of the string.

But, the langauage snip(R) only allow one '1001' in the string and only allow it in the middle.

$xyyz \notin \text{snip}(R)$, contradition.

$\therefore$ when R is regular, snip(R) not necessary regular.