

11 September, 2019

Context-free - Language.

এই language কৈমনি describe অস্থায় কৈমনি context-free-grammar
আগবঢ়া.

On, n কৈমনি nonregular language. Cfr. almost
recursion এখন অস্থায়, n বিনি zero কৈমনি ইত্যাব পর । নিচের
Back অস্থায় আবশ্যিক কৈমনি recursion use অস্থায়
-এক্সেপ্রেছ কৈমনি context-free - Language use অস্থায়

example:

$A \rightarrow OA_1$

$A \rightarrow B$

$B \rightarrow \#$

—> প্রয়োজিত line G1 (grammar) Rule.

A can be $OA_1 \rightarrow$ এই line কৈমনি symbol A
string Cfrir. OA_1 .

So, প্রয়োজিত rule কৈমনি symbol —> অস্থায়

—> cfrir string আবশ্যিক. Separated by an arrow.

productions —> প্রয়োজিত line G1 (এক্সেপ্রেছ কৈমনি কৈমনি) G1 (এক্সেপ্রেছ কৈমনি)

$A \rightarrow OA1$

A can be B

$A \rightarrow B$

B can be $\#$

$B \rightarrow \#$

ගෙව ප්‍රතිචරණ මෙය එහි අනුමත පාඨමාලා වාස්තු නේ.

So, C යුතු ඇත් terminal. ($O, 1, \#$ terminal)

එක්ස්ප්‍රෝස් ඇත් variable. (ගෙව, A, B variable)

නැත්තු String නිෂ්පාදන rule හා ප්‍රාග්‍රැම් ප්‍රාග්‍රැම් Starting rule

ඟීඩ් ග්‍රැම් හි ගැනීම rule හා ප්‍රාග්‍රැම් ප්‍රාග්‍රැම් Starting rule

ඟීඩ් ග්‍රැම් හි $000\#111$ හි ප්‍රාග්‍රැම් ප්‍රාග්‍රැම් ප්‍රාග්‍රැම් ප්‍රාග්‍රැම් accept බජ්‍යා තිබූ වාස්තු

starting (ගෙව නොවු) start value variable, ප්‍රාග්‍රැම් ප්‍රාග්‍රැම්

ඟීඩ් ග්‍රැම් ඇත් variable හි A .

$A \Rightarrow O \underline{A} 1$

$\Rightarrow \underline{O} O \underline{A} 1 1$

$\Rightarrow \underline{O} O O \underline{A} 1 1 1$

$\Rightarrow \underline{O} O O B 1 1 1$

$\Rightarrow \underline{O} O O \# 1 1 1$

and (ගැනීම් නිෂ්පාදන) $\rightarrow \left\{ \begin{array}{l} O \rightarrow A \\ A \rightarrow A \\ B \rightarrow A \\ \# \rightarrow A \end{array} \right.$

So, ඇත් String නිෂ්පාදන rule ගැනීම describe නම්

අනුමත -last String (ගැනීම් variable නිෂ්පාදන මාධ්‍ය ප්‍රාග්‍රැම් ප්‍රාග්‍රැම්)

සහ ඇත් තුළු තුළු නිෂ්පාදන මාධ්‍ය ප්‍රාග්‍රැම්

සහ ඇත් තුළු තුළු නිෂ්පාදන මාධ්‍ය ප්‍රාග්‍රැම්

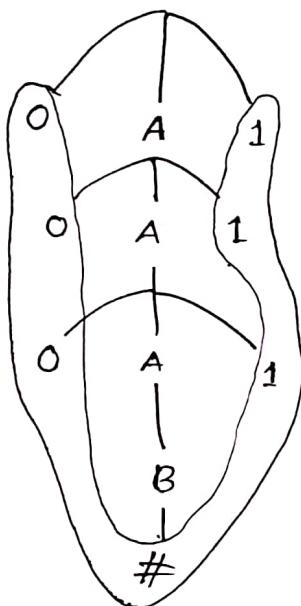
$0^n \# 1^n \rightarrow$ ගැනීම් තුළු finite automata

ගැනීම් තුළු පාඨමාලා වාස්තු. But context-free-language නිෂ්පාදන මාධ්‍ය මාධ්‍ය.

variable
ගැනීම්

ଲାଙ୍ଘେ ଗେନ ଫ୍ରେସ୍ ପରେ ଟ୍ରେ ଅଧ୍ୟାତ୍ମଣ

$A \rightarrow$ start variable.



000#111

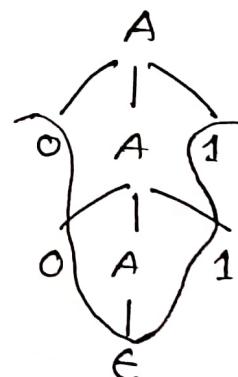
example 2:

$$1. A \rightarrow OA1$$

$$2. A \rightarrow \epsilon$$

0011

$$\begin{aligned} A &\Rightarrow OA1 \\ &\Rightarrow OOA11 \\ &= OOE\epsilon 11 \\ &= 0011 \end{aligned}$$

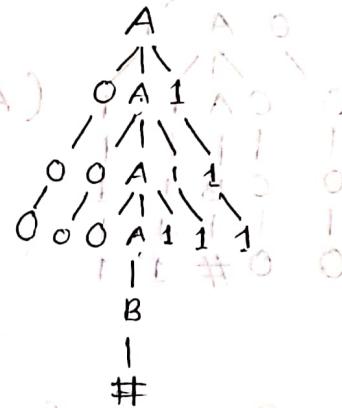


12 September, 2019

$$\begin{array}{l} A \rightarrow 0A1 \\ A \rightarrow B \\ B \rightarrow 0\# \end{array}$$

000#111

11 # 00



0 (1AO-A) 11A00 = 1AO - A
0 (1AO-A) 11A00 = 0 (B-A) 11B00 = 0 (B-A) 11#00 =

String 19220 is accepted by NFA because 19220

1. Substitution Rules / Production

→ . # → ∅

2. Variables

→ . # → ∅

3. terminals.

→ . # → ∅

4. Start variable.

→ . # → ∅

All these steps together make up a derivation.

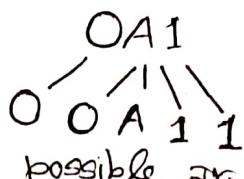
String generation (in a way): 1. derivation

2. parse tree.

→ pictorial view.

The sequence of substitutions to obtain a String:

0011 → 0₁₁ String
0 → 0₁₁ grammar step or possible an



00#11

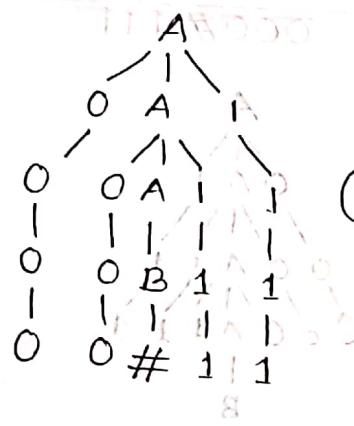


$$A = OA1$$

$$= 0OA11 \quad (A \rightarrow OA1)$$

$$= 0OB11 \quad (A \rightarrow B)$$

$$= 00\#11 \quad (B \rightarrow \#)$$



$$(A \rightarrow OA1)$$

so, কোটি grammar দ্বাৰা এই language কোটি accept কোৱে?

$$B \rightarrow \# \quad \therefore \#$$

$$A \rightarrow B \rightarrow \# \quad \therefore \#$$

$$A \rightarrow OA1 \rightarrow OB1 \rightarrow O\#1 \quad \therefore$$

Conseq. একটি Hash Grammar হলো যাতে প্রতিটি terminal একই অক্ষরের মুক্ত এবং প্রতিটি non-terminal একই অক্ষরের মুক্ত।

$$\therefore L = \{ \# , 0\#1, 00\#11, \dots \}$$

$$P = \{ 0^n\#1^n \mid n \geq 0 \}$$

G2: A → OA1

$A \rightarrow e$

start variable A.

$\vdash A \rightarrow \epsilon$.

$$A \rightarrow OA1 \rightarrow O\epsilon 1 \rightarrow O1$$

$$A \rightarrow 0A1 \rightarrow 00E11 \rightarrow 0011.$$

TAO → A

$$S \Delta \emptyset = A$$

$$2 \Delta C = A$$

Georgia. $\{G = \{e, \sigma, 0011, \dots\}\}$

$$\Rightarrow \langle \text{sample doc} \rangle \langle \text{sample doc} \rangle \dots \langle \text{sample doc} \rangle \\ = \{01, 0011, \dots\}$$

$$= \{0^n 1^n ; n \geq 0\}$$

Start variable specific by default *name* first rule or $\text{*var*$ buffer.

মাত্র context free grammar দ্বারা language প্রেরণ করা যাবে। মাত্র context free language. (CFL)

Rule - Go merge करता थायः

A → OA1

A → ØA2

A → OA3

A → OA1 | ØA2 | OA3

<Sentence> → <noun, phrase> <Verb phrase> &

<noun phrase> → <complex noun> | <complex> <prep>

<verb> → <complex> | <complex> <prep>

<complex> noun → <ART> <NOUN>

<prep> → <preposition> <complex noun>

<complex> → <verb> | <verb> <noun phrase>

<ART> → a an | the

<NOUN> → boy | girl | flower

<verb> → touches | likes | sees.

<preposition> → with.

variable डिग्री & <> (या अंग डिग्री)

Variables → 10

terminals → 26

Start variable →

$\langle \text{sentence} \rangle$

$\langle \text{sentence} \rangle \rightarrow \langle \text{noun phrase} \rangle \langle \text{verb phrase} \rangle$

→ $\langle \text{complex noun} \rangle \langle \text{complex} \rangle$ (Repetitive structure)

→ $\langle \text{ART} \rangle \langle \text{NOUN} \rangle \langle \text{verb} \rangle \langle \text{noun phrase} \rangle$

→ the girl likes $\langle \text{complex noun} \rangle$ (Adjective)

→ the girl likes $\langle \text{ART} \rangle \langle \text{NOUN} \rangle$ (Adjective)

→ the girl likes the flower. (Additional factor)

elsewhere more

Approach → [Top down: start variable for string A.
bottom up: string for start variable A.]

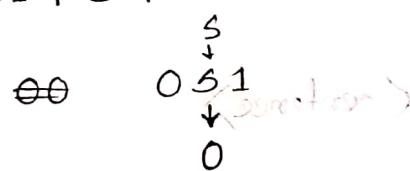
• $S \rightarrow 0S1 | \epsilon$

$L(S) = \{01, 0011, 000111, \dots\}$

এই language কিম্বা 001 কে describe করা যায় না।

But ~~for~~ ক্ষয়ে পাইলে $S \rightarrow 0$ নথিটির rule include
 ক্ষয়ে 001 পাওয়া যাবে,

$$S \rightarrow 0s1 \in T^*$$



$\langle \text{empty}, 001 \rangle \langle \text{empty}, \text{root} \rangle \rightarrow \langle \text{empty} \rangle$

formal definition:

$\langle \text{empty}, \text{root} \rangle \langle \text{empty} \rangle \langle \text{empty}, \text{left} \rangle \langle \text{empty}, \text{right} \rangle \rightarrow$

4-tuple: $\langle \text{empty}, \text{root} \rangle \text{ left } \text{ right } \text{ left } \text{ right } \rightarrow$

(V, Σ, R, S)

set of variables/
non terminals.

$\Sigma = \text{Set of terminals}$

$R = \text{Set of Rules.}$

$S = S \in V \text{ (Start variable)}$

$S \rightarrow aNb | Ns | \Theta \rightarrow$ ~~for terminal aN~~

1. $V = \{S\}$

2. $\Sigma = \{a, b, \Theta\}$

Θ ক্ষয়ে যাবে এবং aN ক্ষয়ে যাবে এবং Ns ক্ষয়ে যাবে।

G1: $a - C$
 $b - D$

$S \Rightarrow aSb$

$\Rightarrow aSbSb$

$\Rightarrow aaaSbSb$

$\Rightarrow aaSbSSb$

$\Rightarrow aaSbSSb$

$\Rightarrow aabaaSbabbb$

$\Rightarrow aabababb$

$\Rightarrow ((())()) \rightarrow$ So, it's grammar or balanced parenthesis

1. G1. $\langle \text{EXPR} \rangle \rightarrow \langle \text{EXPR} \rangle + \langle \text{TERM} \rangle$

2. $\langle \text{TERM} \rangle \rightarrow \langle \text{TERM} \rangle \times \langle \text{FACTOR} \rangle \mid \langle \text{FACTOR} \rangle$

3. $\langle \text{FACTOR} \rangle \rightarrow (\langle \text{EXPR} \rangle) \mid a$

variable: {+, ×, (), a}

$S = \{ \langle \text{EXPR} \rangle \}$

$V = \{ \langle \text{EXPR} \rangle, \langle \text{TERM} \rangle, \langle \text{FACTOR} \rangle \}$

$$\begin{aligned} B &= E + T \\ &= \overbrace{E+T} + T \\ &= \overbrace{E+T+T+T} \\ &= T+T+T+T \dots \quad (B \rightarrow T) \end{aligned}$$

E অন্তিমের ত এবং প্রাপ্তি,

$$T \Rightarrow T \times F$$

$$\begin{aligned} &\Rightarrow \overbrace{T \times F} \times F \\ &\Rightarrow \overbrace{T \times F \times F \times F} \\ &\Rightarrow F \times F \times F \times F \quad T \rightarrow F \text{ হলো } \end{aligned}$$

T অন্তিমের F এবং প্রাপ্তি,

$$*(\alpha + \alpha \alpha) / (\alpha \alpha \alpha) + (\alpha \alpha \alpha) \times (\alpha \alpha \alpha) \leftarrow (\alpha \alpha \alpha) \cdot \alpha$$



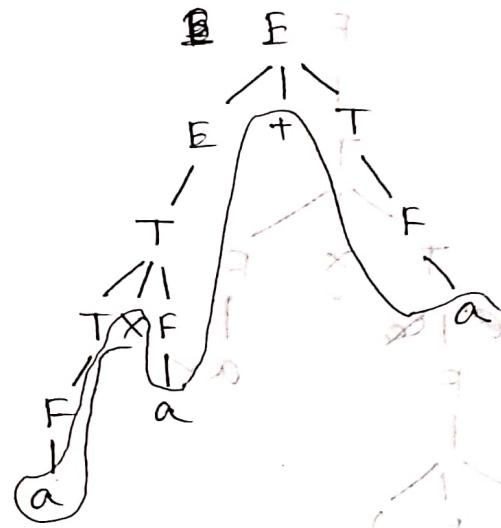
$$E \rightarrow B+T | T$$

$$T \rightarrow T \times F | F$$

$$F \rightarrow (E) | \alpha$$

$$\{(\alpha \alpha \alpha), (\alpha \alpha \alpha), (\alpha \alpha \alpha)\} = \alpha$$

* $a + \alpha a$



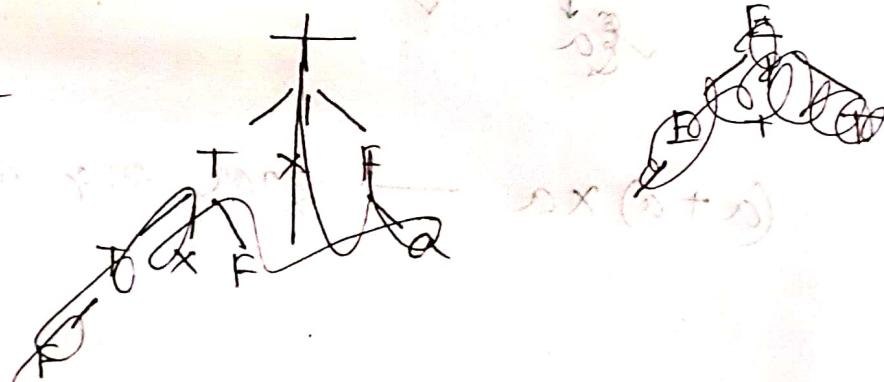
$\alpha a a + a$

$\therefore \alpha a + a$

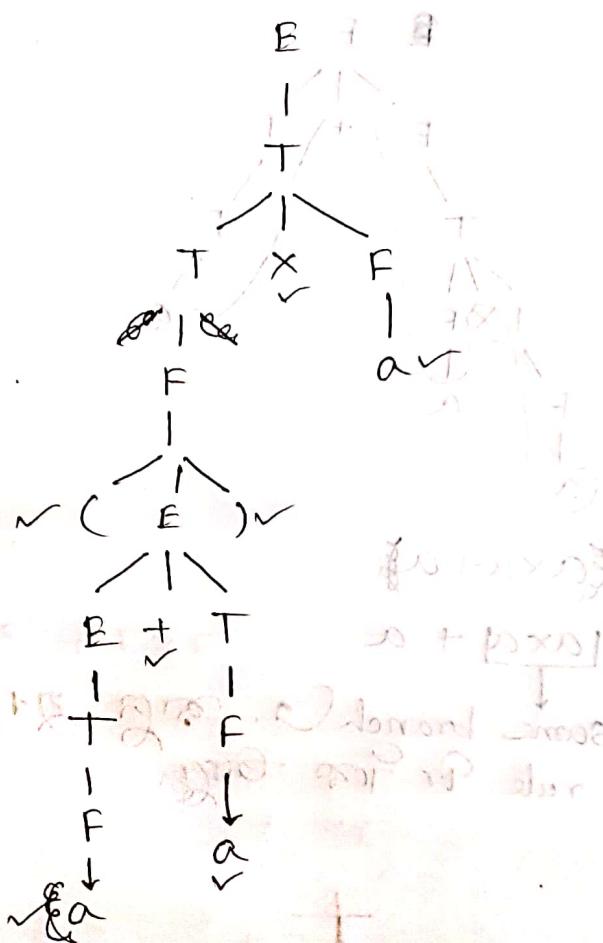
same branch \hookrightarrow rule or goes to α

So, C rule or goes to α

* $(a + a) \alpha a$



$$(a+a) \times a$$



18 September, 2019

$$E \rightarrow E + T \mid T$$

$$T \rightarrow T \times F \mid F$$

$$F \rightarrow (E) \mid a$$

$$a + axa$$

$$E \Rightarrow E + T$$

$$\Rightarrow T + T$$

$$\Rightarrow F + T \times F$$

$$\Rightarrow F + F \times F$$

$$\Rightarrow a + axa$$

E টা প্রজন্মের step দ্বারা পর a+axa হয়েছে। এই কথাটা কোনো অভিযোগ নেওয়া হলো, $a \in E \xrightarrow{*} a + axa$.

1. Palindrome of 0 and 1.

এখন grammar দ্বারা অন্তর্ভুক্ত হবে,

কোনো কোটির ক্ষেত্রেই হিসাবে same নম্বৰ থাকবে

২৬।

$S \rightarrow 0S0$

$S \rightarrow 1S1$

$S \rightarrow 0$

$S \rightarrow 1$

$S \rightarrow \epsilon \quad \therefore S = 0\epsilon|0|1|0S0|1S1$

କେବି ଗ୍ରାମ୍ପର ଗ୍ରହଣ 0, 1, 00, 11, ϵ

ଏଥେମାଟିକ୍ ଅପ୍ପାନ୍ତରେ accept ହେଲେ।

ଯାହାର ଏବେବିହାରୀ palindrome.

କେବି ଏବେବିହାରୀ generate ହେଲେ।

011 010110

$S \rightarrow 0S0$

$\rightarrow 01S10 \quad (S \rightarrow 1S1)$

$\rightarrow 011S110 \quad (S \rightarrow 1S1)$

$\rightarrow 0110S0110 \quad (S \rightarrow 0S0)$

$\rightarrow 011010110 \quad (S \rightarrow 1)$

2. $L = \{0^n 1^n \mid n \geq 0\} \cup \{1^n 0^n \mid n \geq 0\}$

ଯାହାର, ଲାଭତର ସାମେ ଅନ୍ୟରେ 01 ହେଲେ।

— ଅନ୍ୟରେ same ଅନ୍ୟରେ 1.0 ହାବାନ୍ତରେ।

$S_1 \rightarrow 0 S_1 1$

$S_1 \rightarrow \epsilon$

$S_2 \rightarrow 1 S_2 0$

$S_2 \rightarrow \epsilon$

କେବି ଫୁଲିବା ଉନିଯନ୍ ଏବିତିରେ।

$$S \rightarrow S_1 | S_2$$

$$S_1 \rightarrow 0S_1 | \epsilon$$

$$S_2 \rightarrow 1S_2 | \epsilon$$

3. $L = \{0^n 1^{2n} \mid n \geq 0\}$

$$S \rightarrow 0S11 | \epsilon$$

माना, $n \geq 1$ - इसे first String कहेंगे 011.

$$\therefore S \rightarrow 0A11$$

$$A \rightarrow \epsilon | OA11$$



4. $L = \{0^n 1^m \mid m, n \geq 0, 2n \leq m \leq 3n\}$

if, $n=0 \quad m=0$

$n=1 \quad m=2, 3 \quad 011, 0111$

$n=2 \quad m=4, 5, 6 \quad 001111, 0011111, 00111111$

$$\begin{aligned} S &\rightarrow \epsilon \\ S &\rightarrow 0S11 \\ S &\rightarrow 0S111 \end{aligned}$$

ϵ accept করিব,
011, 0111 accept করিব।

001111 2nd rule উন্তু প্রয়োগ করিবে এবং মাত্র।

001111 2nd rule করিবে,
3rd rule করিবে।

$$5. L = \{0^n 1^m \mid n, m \geq 0, n \neq m\}$$

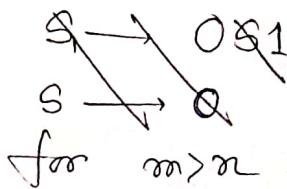
আপ্স, ইফি, $n > m$

অথবা $m > n$.

এখন, $00^n 1^n$ $n > m$ করিবে satisfy করিব।

$0^n 1^m$ $m > n$ করিবে satisfy করিব।

$\therefore n > m$ কর অস্ট্র।



$$S_1 \rightarrow 0S_1 | 0E$$

$$E = 0^n 1^n$$

$$S_2 \rightarrow 0S_2 1 | E1$$

$$E = 0^n 1^n$$

$$B \rightarrow 0E1 | \epsilon$$

$$\therefore S = S_1 | S_2$$

$$S_1 = 0S_1 | 0E$$

$$S_2 = S_2 1 | E1$$

$$E1 = 0E1 | \epsilon$$

(যদি start variable এ নেমা থাকুন,
তাহলে start প্রয়োগ কর এবং ক্ষেত্রে
ক্ষেত্রে নেওয়া হবে।)

19 September, 2019

$$6. L = \{ \omega | \omega \in \{a, b\}^* \text{ and } n_a(\omega) = n_b(\omega) \}$$

সেখন a, b এর সমিক্ষণ শব্দ, Σ লিপি a, b
সমিক্ষণ শব্দগুলি কী?

$$S \rightarrow aSb$$

$$S \rightarrow bSa$$

$$S \rightarrow \epsilon$$

$$ab \quad ba$$

abbabbab

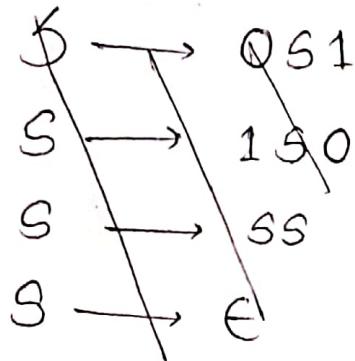
S

$$\begin{aligned} S &\rightarrow aSb \\ &\rightarrow abSab \\ &\rightarrow abasbab \\ &\rightarrow abaasbab \\ &\rightarrow ababbabb \\ &\rightarrow ababbab \end{aligned}$$

But এই গ্রাম্যে $a babba \rightarrow$ গ্রাম্য
accept করে না। কারণ
এই গ্রাম্যে শব্দগুলি সমান লেন্থ
হলেই অক্ষর।

$$\therefore S \rightarrow aSb \mid bSa \mid \epsilon \mid ss.$$

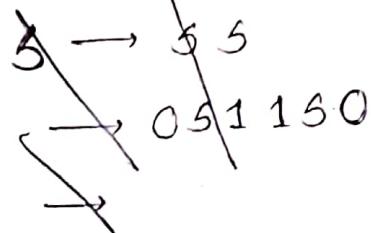
7. $L = \{ w | w \in \{0,1\}^* \text{ and even length} \}$



~~0011011010~~

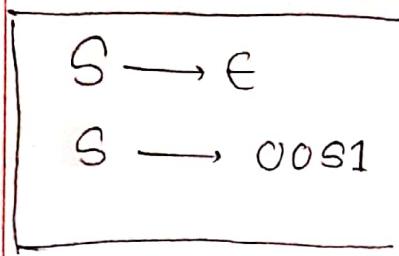
Initial Cases:

ϵ
01 / 10 / 00 / 11
0



$S \rightarrow \epsilon / 0\$1 / 1\$0 / 00 / 1\1

8. $L = \{ 0^{2n} 1^n \mid n \geq 0 \}$



$n \rightarrow 2 \therefore 000011$

9. $L = \{ a^n b^m c^k \mid n, m, k \geq 0 \text{ and } n = m+k \}$

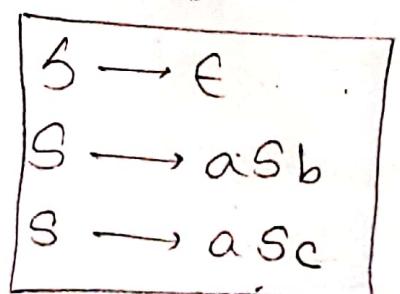
ϵ ,

ac / ab $\dots \rightarrow n=1$

aabc / aabb / aacc $n=2$

~~aaa
aaab~~

$n=3$



But কোনো rule দ্বাৰা $a^m b c^k$ গুৰুতে প্ৰিয়। $a^m b c^k$ গুৰুতে প্ৰিয়।
 So. কোনো language কোনো rule দ্বাৰা নহ'ল।

$$S \rightarrow \epsilon$$

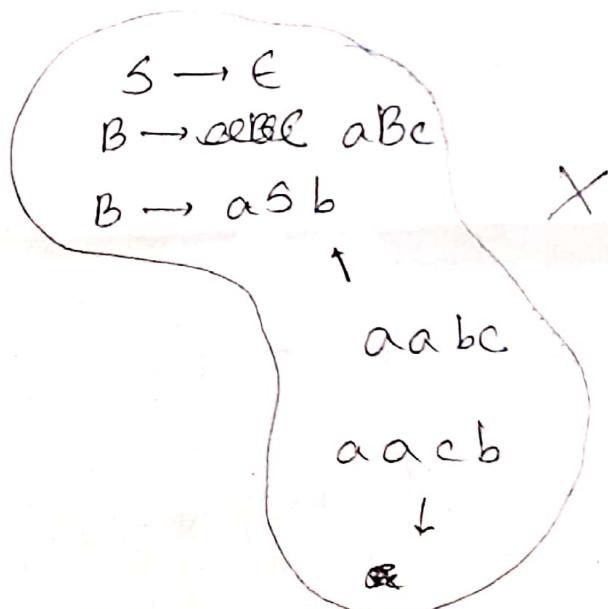
প্ৰথম হুগলি - অসমীয়া - মাত্ৰ এইটো হ'ল এ কোনো rule দ্বাৰা নহ'ল।

$$S \rightarrow \epsilon$$

$$\begin{array}{c} B \rightarrow aBc \\ B \rightarrow aSb \end{array}$$

$$B \rightarrow aBc \mid A$$

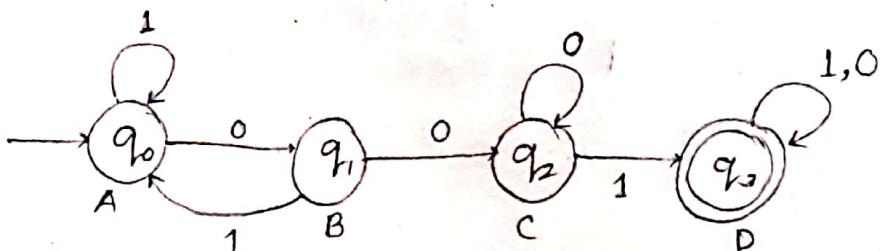
$$A \rightarrow aAb \mid \epsilon$$



8. * প্ৰয়োগিৰ DFA দ্বাৰা CFG টো convert কৰিবলৈ পৰিয়,

$$S(q_i, a) = q_j$$

$$R_i \rightarrow aR_j$$



$A \rightarrow OB | 1A$

$B \rightarrow OC | 1A$

$C \rightarrow OC | 1D$

$D \rightarrow OD | 1D | \epsilon$

6. $L = \{ \omega | \omega \in \{a, b\}^*, n_a(\omega) = n_b(\omega) \}$

Ambiguity

They are flying planes.

- একজন বিমান চলছে
বিমান একজন বিমান

$E \rightarrow E+E | E^*E | a$

$a + a^*a$

$E \Rightarrow E + E$

$\downarrow + \downarrow$
 $E + E^*E$

\downarrow
 $a + a^*a$

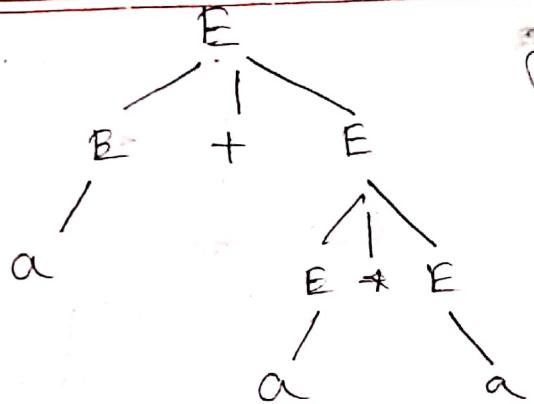
$E \rightarrow E + E$

$\longrightarrow a + E$

$\longrightarrow a + E^*E$

$\longrightarrow a + a^*a$

ইইটির দুইভ্রান্তির derivation.



But parse tree
পরিমূল সমান.
So, derivation different

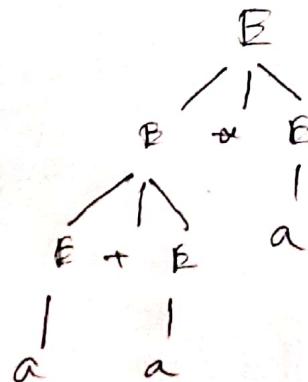
Parse tree
পরিমূল সমান
নয়।

Parse tree different কোনও ক্ষেত্রে:

ক্ষেত্রে সেখানে প্রথম নির্ণয়ের ফলোর
derivation ব্যবস্থা এবং দ্বিতীয় ব্যবস্থা এবং প্রথম:

1. left most derivation
2. right most derivation.

$$\begin{aligned}
 E &\rightarrow E^* E \\
 &\rightarrow (E+E)^* E \\
 &\rightarrow a + a E^* E \\
 &\rightarrow a + a + E \\
 &\rightarrow a + a - a
 \end{aligned}$$



Ques

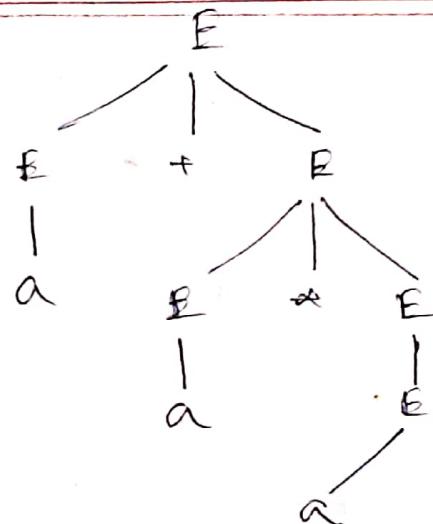
$$E \rightarrow E + E$$

$$\rightarrow a + E$$

$$\rightarrow a + E^* E$$

$$\rightarrow a + a^* E$$

$$\rightarrow a + a^* a$$



ज्ञान ग्रन्थ

Same, leftmost derivation But parse tree different.

~~ज्ञान ग्रन्थ~~, ~~ज्ञान ग्रन्थ~~ different ~~कृत~~ ambiguity ~~कृत~~

25 September, 2019

$$E \longrightarrow E^* E \mid (E) \mid E+E \mid I$$

$$I \longrightarrow ab \mid IO$$

derive,

$$E \xrightarrow{*_{IO}} a^*(a+b00)$$

$$E \xrightarrow{*_{IO}} a^*(a+b00) \quad \text{not correct.} \quad \text{(-)}$$

\hookrightarrow

$$E \longrightarrow E+E$$

$$\longrightarrow I^* E$$

$$\longrightarrow a^* E$$

$$\longrightarrow a^*(E)$$

$$\longrightarrow a^*(E+E)$$

$$\longrightarrow a^*(a+b00)$$

$$\longrightarrow a^*(a+IO)$$

$$\longrightarrow a^*(a+b00)$$

$$\longrightarrow a^*(a+b00)$$

$$\longrightarrow a^*(a+b00)$$

$$E \longrightarrow E^* E$$

$$\longrightarrow E^* (E)$$

$$\longrightarrow E^* (E+E)$$

$$\longrightarrow E^* (E+I)$$

$$\longrightarrow E^* (E+IO)$$

$$\longrightarrow E^* (E+IO)$$

$$\longrightarrow E^* (E+b00)$$

$$\longrightarrow E^* (E+a00)$$

$$\longrightarrow E^* (E+a00)$$

$$\longrightarrow E^* (E+a00)$$

Computer language for describe various type of grammar. Like as, Regular expression, Ambiguity.

Chaitin's algorithm

কোনো grammar ambiguous হিসেবে দেখায় কিন্তু এটা অসমীয়া
 language এর কোনো Grammar String কে যদি কুইচের leftmost
 derivation করলে দ্বিগুণ পরিষ, তাহলে একটো grammar বা
 ab ambiguous.

কোনো grammar ambiguous হিসেবে দেখায় কিন্তু non ambiguous
 grammar লাওয়া যাবে। But কোনো grammar কে কুইচে
 সম্ভব না হিবল করা, তাহলে এটা inherently ambiguous.

$$* L = \{a^i b^j c^k \mid i=j \text{ or } i=k\}$$

a কোর b কোর মান হালেন অপ্রয়োগ, L এতে কোর আড়তে

a কোর, a কোর c

-কুইচ aabbc. অসমীয়া grammar এর কুইচে approach কে ব্যবহার করে
 describe করতে, চাইছো, -কুইচ

ab হালেন করে কুইচে পরিষ,

ac - " " "

-কুইচ approach, তাই ambiguous grammar কুইচে,

CNF (Chomsky Normal Form)

CNF → grammar করা সহজ ও সর্পিল মডেল,

গুরুত্বপূর্ণ নীতি ব্যবহার করে গ্রাম্য রূপ তৈরি করা।

So, নীতি সহজ ও সর্পিল রূপে প্রক্রিয়া।

A context free grammar is in Chomsky Normal Form if every rule is of the normal form.

$$A \rightarrow BC$$

$$A \rightarrow a$$

$S \rightarrow \epsilon$ if S is Start Variable

$$S \rightarrow ASA \mid aB$$

$$A \rightarrow B \mid S$$

$$B \rightarrow b \mid \epsilon$$

i) first (i) ensure অক্ষত রূপ সময়ের উপর
Start variable এ থাকে।

1. $S_0 \rightarrow S$ → তাই S Start variable হিসাব।

2. $S \rightarrow ASA + aB$

3. $A \rightarrow B \mid S$

4. $B \rightarrow b \mid \epsilon$

ii) Empty rule কে অক্ষত রূপে তাই দ্বিতীয় ক্ষেত্রে B ক্ষেত্রে
এই ক্ষেত্রে ক্ষেত্রে B এর জন্মগত ϵ একটি নির্দেশ, then
extra rule add করে নিন্দেশ।

1.2

$S_0 \rightarrow S$ (unit rule এর ফল হলো $S_0 \rightarrow S$) | ab.

$S \rightarrow ASA | a$ (B এর কান্দগাছ empty রয়েছে) | ab.

$A \rightarrow B | S | \epsilon$ (B এর কান্দগাছ ϵ রয়েছে এবং এটির মাঝে
empty লাইনার আছে)

$B \rightarrow b$

প্রথম: অন্তর $A = \epsilon$ করুন,

$S_0 \rightarrow S$

$S \rightarrow ASA | ab | a | S | AS | SA$

$A \rightarrow B | S$

$B \rightarrow b$

$SA \rightarrow A$

$S \rightarrow A$

F step 3: unit rule এর ফল হলো অন্তর করুন:

$S_0 \rightarrow S$

$S \rightarrow S$

(unit rule এর ফল হলো একটি শব্দ)

$A \rightarrow B$

- কান্দকীর single variable এর

$A \rightarrow S$.

terminal এর)

$S_0 \rightarrow \$ | ASA | ab | a | SA | AS | \$$

Q $S \rightarrow ASA | ab | a | SA | AS | \$$

$\$ | ASA$

$A \rightarrow \$ | S | b$

$B \rightarrow b$

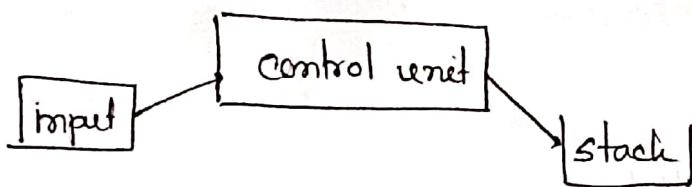
$\$ | S$

$\$ | b$

2 October, 2019

$$L = \{0^n 1^n \mid n \geq 0\}$$

- formal definition
- Designing li.



NFA \rightarrow 5 tuples. : 1. Q

2. $\Sigma_\epsilon = \Sigma \cup \{\epsilon\}$

3. $S: \{q_0, q_1, q_2\}$

4. q_0 :

5. $F:$

PDA. 1. Q

2. Σ_ϵ

3. $S \cup T_\epsilon \rightarrow$ Stack (এবং অর্থে input alphabet)

4. S

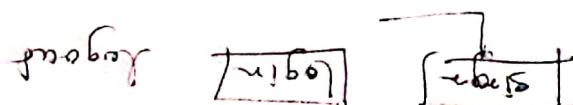
5. q_0

6. F

$$Q \times \Sigma \rightarrow Q$$

Q state এবং, ক্ষেত্র alphabet, সিঙ্গুলার, অধিক্ষেত্রে সম্পর্ক

state এবং স্টেক



b state, min

for NFA to DFA conversion

1. Given state \cup Q

2. Given input Σ & δ

3. Stack \cup initial stack

$$\forall Q \in P(Q \times \Sigma) \rightarrow P(Q)$$

$$P(Q \times \Sigma \times T_e) \rightarrow P(Q \times T_e)$$

$$Q = \{q_1, q_2\}$$

$$\delta(q_1, a) \rightarrow q_j$$

$$\Sigma = \{a, b\}$$

$$\delta((q_1, a), a) \rightarrow (q_2, \epsilon)$$

$$T_e = \{a\}$$

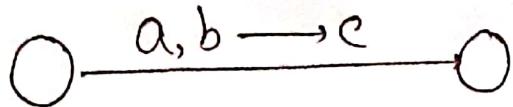
माना है state \cup प्रारंभिक, देखते हैं a input
प्रिवाया तो उसके next State \cup मान्दे। Then
उसे State \cup अकाशीनीर Stack \cup मा दाखल
करेंगे और उसके next state \cup मान्दे।

जब Stack empty हो जाएगा तो देखते हैं अग

देखते हैं input \cup देखते हैं जब Stack empty हो जाएगा तो देखते हैं flag stack

जब Stack empty हो जाएगा तो pop करते हैं अपने Stack flag तो लगा

जब Stack empty हो जाएगा तो Stack छोड़।



a input
b stack top.

a input প্রেরি, b stack এ রাখবে, stack
c কে পুরা করিব।

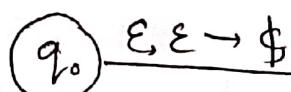
যদি, a = ε

b = ε

c = b

হলো

stack এ রাখবে b.



0 input, stack এ \$ = 0 :- stack top
এ রাখবে 0.

0 input, stack এ 0 :- stack top
এ 0.

1 input, stack এ 0, . : stack top
এ, ε. ক্ষেত্রে রাখবে (প্রক্রিয়া)

1, 0 রাখবে, ε রাখবে।

input ε, stack এ dollar. রাখবে

input 0, রাখবে, stack এ 0, 0, 0

- রাখবে stack এ ε রাখবে।

$$L = \{0^n 1^n \mid n \geq 0\}$$

$$1. Q = \{q_0, q_1, q_2, q_3\}$$

$$2. \Sigma_e = \{0, 1\}$$

$$3. T = \{0, \$\}$$

$$4. S =$$

$$5. q_0 = q_0$$

$$6. QF = \{q_0, q_3\}$$

4. S:

input	0	1	ϵ
stack	0, \$, ϵ		
q_0			
q_1			
q_2			
q_3			

input	0	1	ϵ
stack	0, \$, ϵ	0, \$	ϵ
q_0			
q_1			
q_2			
q_3			

input	0	ϵ	$\$$	ϵ	ϵ	0	ϵ	$\$$	ϵ	$(q_1, \$)$
stack	0	ϵ	(q_1, ϵ)							
	q_0	q_1	q_2	q_3						

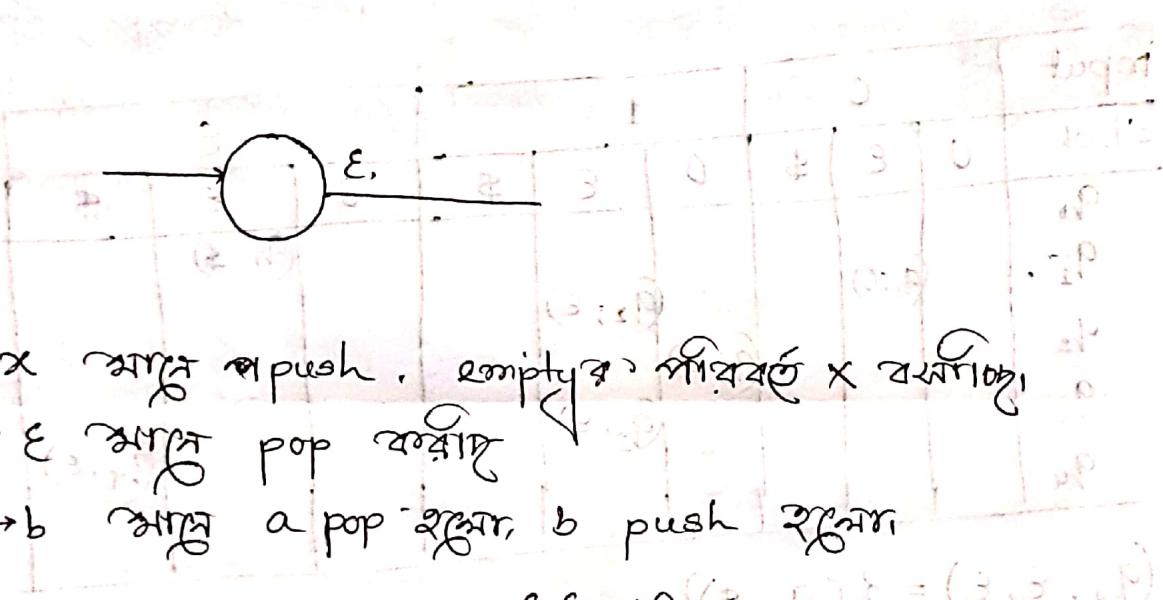
- Element are stored in reverse order, $\{a_i, 0\} \subset \text{transient stack}$, $\{a_i, 0\} \subset \text{transient stack}$
- Element are stored in reverse order, $\{a_i, 0\} \subset \text{transient stack}$, $\{a_i, 0\} \subset \text{transient stack}$
- Element are stored in reverse order, $\{a_i, 0\} \subset \text{transient stack}$, $\{a_i, 0\} \subset \text{transient stack}$
- Element are stored in reverse order, $\{a_i, 0\} \subset \text{transient stack}$, $\{a_i, 0\} \subset \text{transient stack}$
- Element are stored in reverse order, $\{a_i, 0\} \subset \text{transient stack}$, $\{a_i, 0\} \subset \text{transient stack}$
- Element are stored in reverse order, $\{a_i, 0\} \subset \text{transient stack}$, $\{a_i, 0\} \subset \text{transient stack}$
- Element are stored in reverse order, $\{a_i, 0\} \subset \text{transient stack}$, $\{a_i, 0\} \subset \text{transient stack}$
- Element are stored in reverse order, $\{a_i, 0\} \subset \text{transient stack}$, $\{a_i, 0\} \subset \text{transient stack}$

3 October, 2019

$$L = \{0^n 1^{2n} \mid n \geq 0\}$$

zero শুধু push কর্যালৈ হলো । পেছে pop কর্যালৈ এবলি zero
কিনা বাবে । পেছে stack কর্যালৈ modification কর্যালৈ।

0010011 input দিয়ে, 1st বাবে । পাইমাস পথ - গোল
zero পেছে push কর্যালৈ কৰ।



$\epsilon \rightarrow x$ মানে push . empty র পীঁথ বলুন x কর্যালৈ।

$a \rightarrow \epsilon$ মানে pop কর্যালৈ।

$a \rightarrow b$ মানে a pop কর্যালৈ, b push কর্যালৈ।

$$0, \epsilon \rightarrow 0 \quad ((\epsilon, \epsilon)) = (\epsilon, \epsilon, \epsilon)$$

$$((\epsilon, \epsilon)) = (\epsilon, \epsilon, \epsilon)$$

q_3 শুধু পেছে already হলো । পেছে দিয়ে আসো কেবল হলো

1 পেছে আসো, q_3 থেকে 1 দিয়ে q_2 শুধু কেবল যায় আসো। তাহলু
কেবল 1 দিয়ে q_3 শুধু আসতে পারে। এজেন্ট কেবল হলো ।
দিয়ে q_3 শুধু আসতে পারে।

$$1. Q = \{q_1, q_2, q_3, q_4, q_5\}$$

$$2. \Sigma_\epsilon = \{0, 1\}$$

$$3. T_\epsilon = \{\$, \epsilon\}$$

$$4. S = \{q_0\}$$

$$5. q_0 = q_0$$

$$6. F = \{q_1, q_4\}$$

8

input	0			1			ϵ		
stack	0	ϵ	$\$$	0	ϵ	$\$$	0	ϵ	$\$$
q_0									
q_1		$(q_1, 0)$			(q_2, ϵ)			$(q_1, \$)$	
q_2				(q_3, ϵ)					
q_3					(q_2, ϵ)				
q_4								(q_4, ϵ)	

$$1. \delta(q_0, \epsilon, \epsilon) = \{(q_1, \$)\}$$

$$\delta(q_1, 0, \epsilon) = \{(q_1, 0)\}$$

$$\delta(q_2, 1, \epsilon) = \{(q_2, \epsilon)\}$$

$$\delta(q_3, \epsilon, \epsilon) = \{(q_4, \epsilon)\}$$

$$\delta(q_4, \epsilon, \epsilon) = \{(q_4, \epsilon)\}$$

$$L = \{w, w^R \mid w \in \{0, 1\}^*\}$$

ওয়ার্ড w যা অবশ্যই, w^R আবশ্যিক reverse

কেন-এভাই 0011100 , 0011 push করলে, 1 পর

আবশ্যিক 1 match করলে, 1 পর

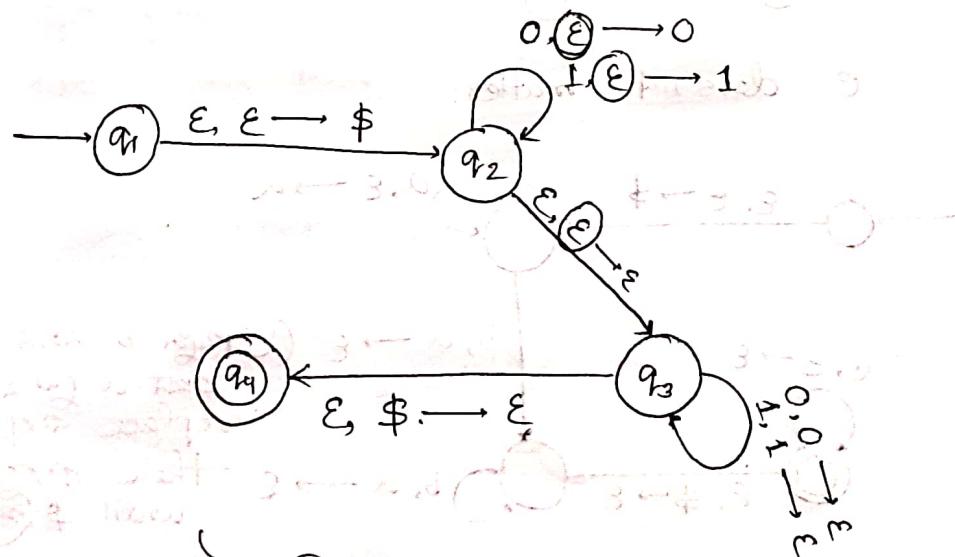
pop করলে, But problem

হচ্ছে, হেন আবশ্যিক w^R আবশ্যিক করলে পরিষ্কার না

- তাই অত্যধিক String read করায় অবশ্য দ্বিতীয় নির্মাণ করে নির্মাণ w . - কাউন্ট w^R করিবে w এর সাথে পরিষ্কার কর

এ অবশ্যে,

0 1 0 1 1 0 1 0



আবশ্যিক প্রয়োগ শৈলী \$ অবশ্যই নির্দিষ্ট, \$ প্রয়োগ করলে তাই নেই.

তাই 0 ব্যবহার করলে, 1 প্রয়োগ same করা করতাম,

অত্যধিক Stack push করার পরে ε push করতাম, input করে, stack করে ε, 0, ε অবশ্যই, then মুদ্রণ 0 input করো,

for input

সেখা stack কিম্বা ০-মানের অস্থির এ ক্ষয়ে প্রিয়ে,

প্রত্যক্ষম: stack এ empty প্রেমে, input কি মান ০, ১
অর্হ অস্থির ০, ১ অন্তর্ভুক্ত কোথা ওই state রেখ
অস্থির, যখন input কি এ প্রেম সীমিত তখন অব্য state

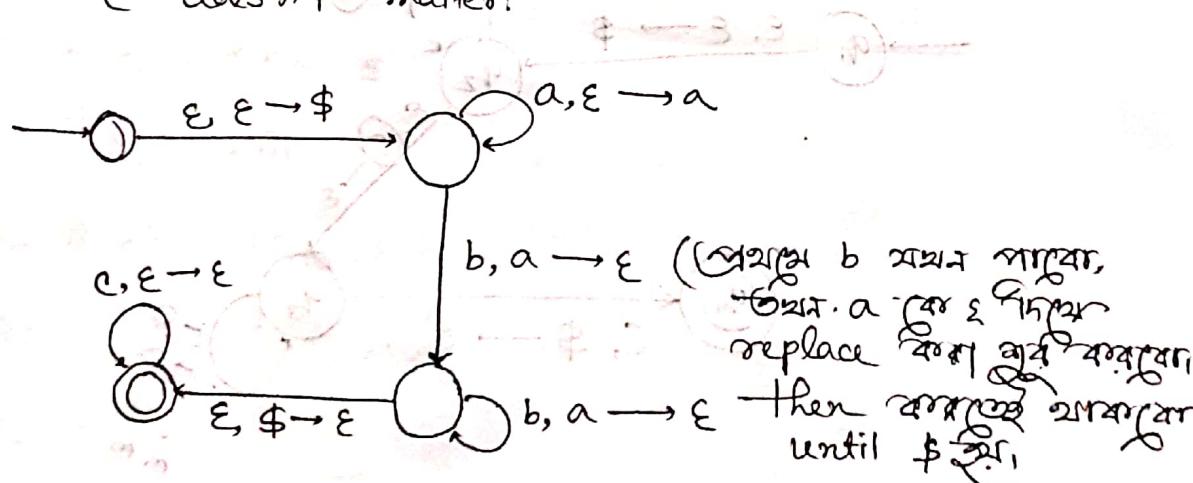
এ দিগন্তে pop করা শুরু বল্বো, আরু আরু এর define কর

বল্বো String কে জায়গামূলক প্রেমে নিখন

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

এভেজে, $i=j$ কি অর্জ:

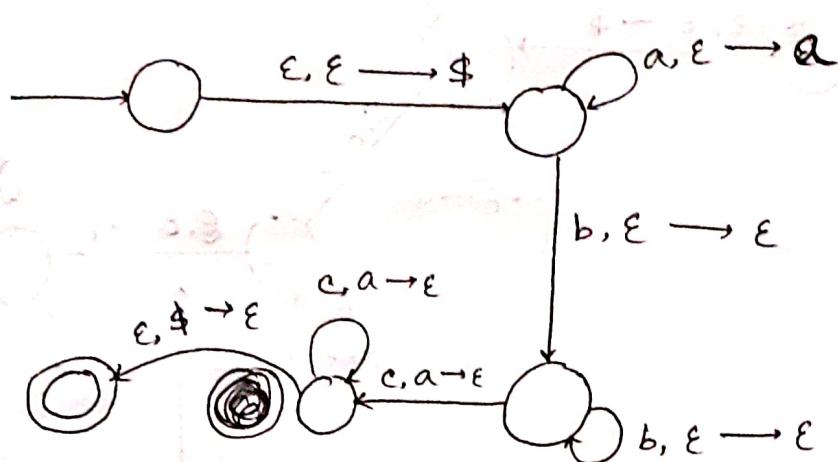
c doesn't matter



$b, a \rightarrow \epsilon$ (প্রথম b শব্দে পার্য্য,
তখন a কে ϵ প্রিয়ে
replace করে শুরু বল্বো,
then অন্তর্ভুক্ত প্রাপ্ত পর
until $\$$ আসে)

09) এখন, a, b, c গুলি consider করো।

Eq

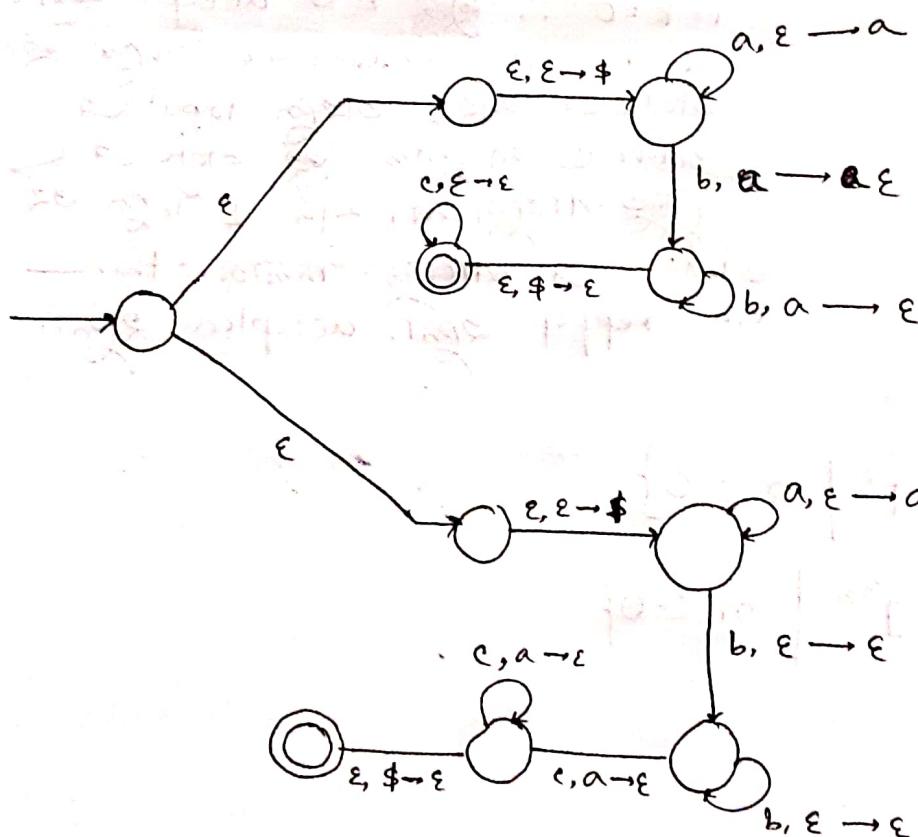


Stack

বাই

read

Next



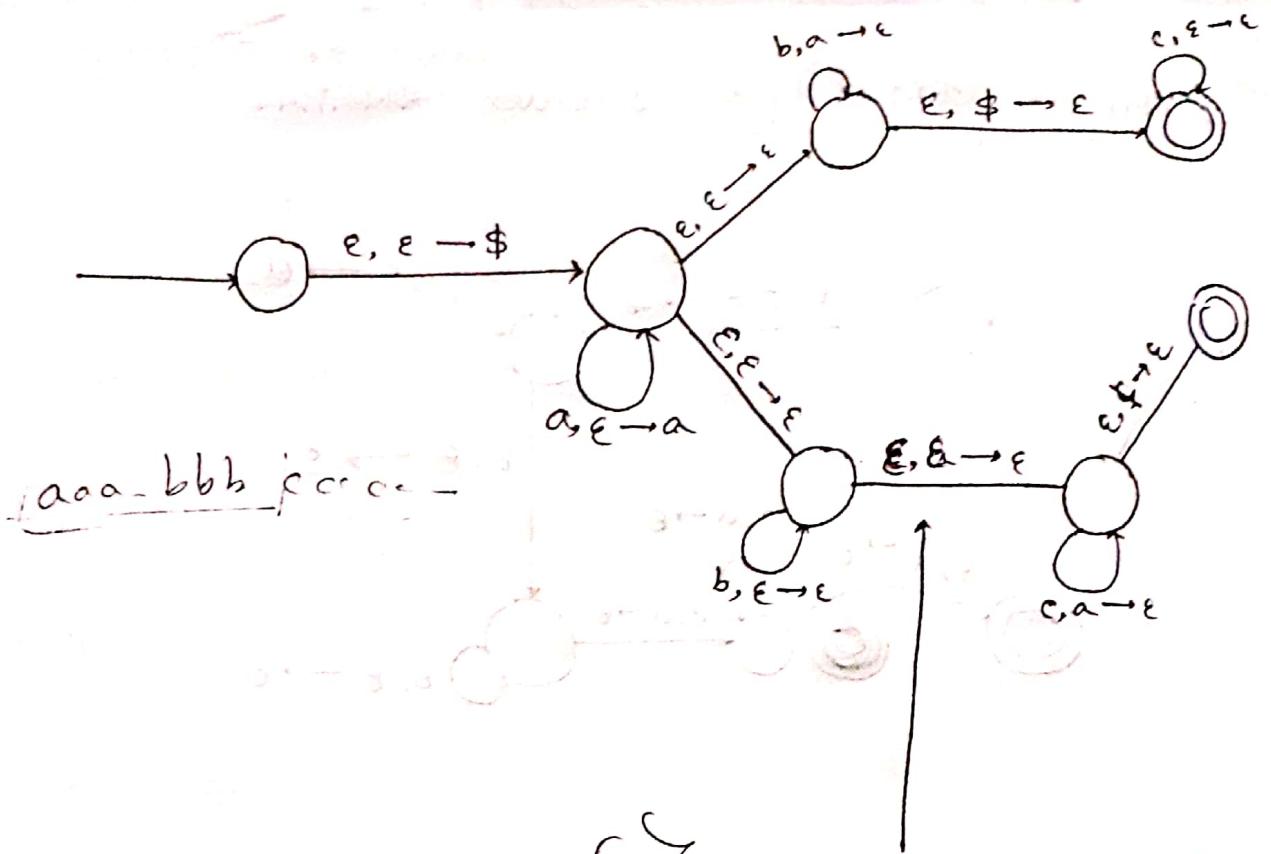
Q

Ans:

Q:

Ans:

a



aaa-bbb-ccc-

এই ফেয়ারিয়াল, $b,a \rightarrow \epsilon$ র জন্য
কথ্য নির্বাচন হওয়া এবং ক্ষয়তি $i=k$ হলো.
 $i=k=0 \therefore$ প্রথম b গুলি accept করা
হয়। But, $a,b,c \rightarrow \epsilon$ র জন্য তা
state 4- শীর্ষে তাহলে input
পরিসর প্রদেশ দ্বারা state 4
ক্ষেত্র প্রস্তুত না থারে এ নির্বাচন দ্বা
রা state 4- প্রস্তুত নির্ভর। Then
 $c,a \rightarrow \epsilon$ র জন্য accepted হওয়া

Task

$$L = \{0^{2n} 1^n \mid n \geq 0\}$$

$$L = \{0^{2n} 1^{3n} \mid n \geq 0\}$$

09 October, 2019

Equivalence with CFG.

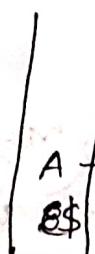
$$A \rightarrow 0A1 | \$\#$$

Stack \sqcup rules \Rightarrow push অবগুর ফর্মের symbol, এবং input মিন কোডু তেমনী পাই Stack \sqcup . তখন Grammar String read করতে হবে।

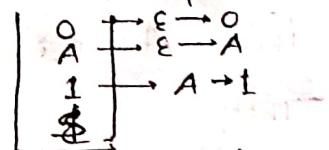
Stack \sqcup first \hookrightarrow start variable \sqcup rule for string করণ।

$$\begin{aligned} A &\rightarrow 0A1 | B \\ B &\rightarrow aBb | \$\# \end{aligned}$$

$0 a\# b 1$ টা
generate করত।



Start Variable টা
 $0A1 \sqcup$ এবং B কোডু
replace করত। যাৰে, So

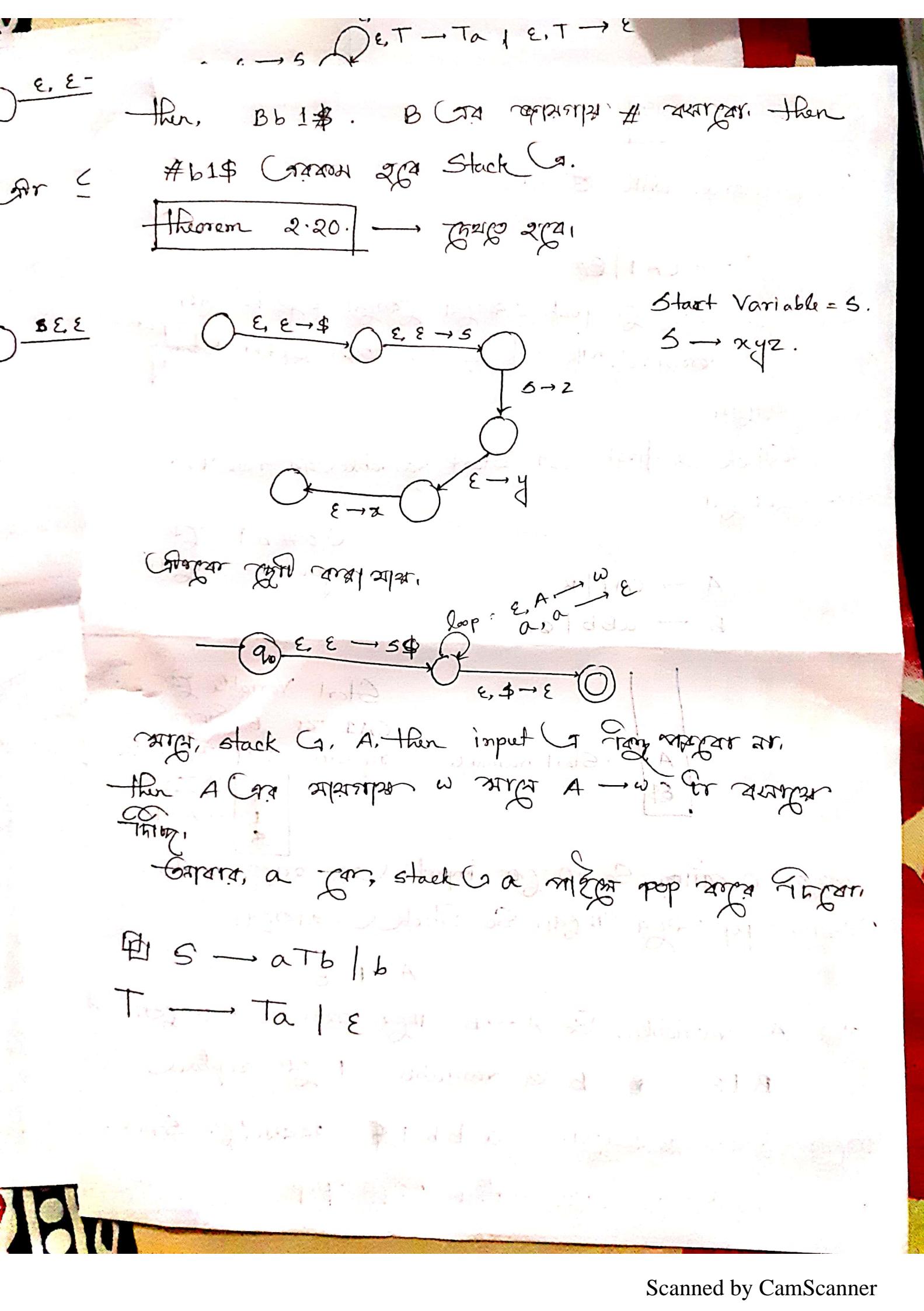


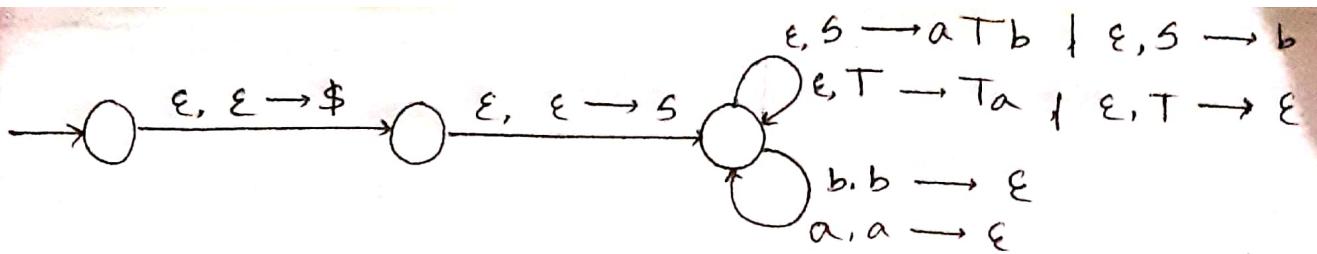
পথম 0 আছি, So, 0 টা $\#$ input দেওয়া হোক্যু মাচ
অবগুর pop কোডু ফিলে, So Stack \sqcup থাকলৈ,

$$A \rightarrow 1 \#.$$

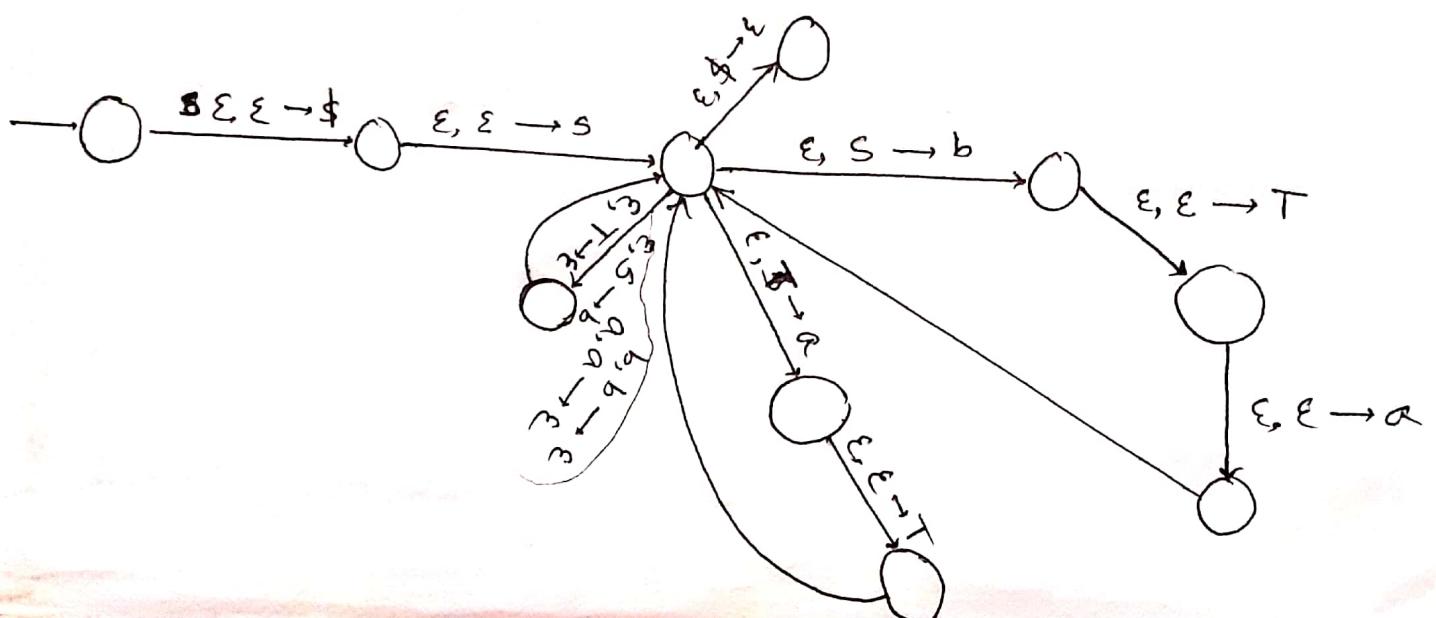
এখন, A , variable. So $A \rightarrow B$ কোডু replace কোডু,

$B 1 \#$. $\#$ B বা variable. B কোডু replace
কোডু aBb কোডু, $aBb 1 \#$ পথম টুকু Stack.
a কোডু input দেওয়া হোক্যু মাচক কোডু pop.





Or Shortcut form. *किंवा ग्रन्थातील याची वर्णन.*



$S \rightarrow aSb$

$S \rightarrow SS$

$S \rightarrow \epsilon$

