Name : Sanket Chandrashekhar Harvande
Roll : 19
Sign : [signature]

Sub: Theory of Computer Science

Assignment No. : 03

Q.1 Define :-

a) Context free Grammer

⇒ A context free grammer G is a quadruple
$G = (V, \Sigma, P, S)$ where
V : set of non-terminal
Σ : set of terminal
P : set of rewriting rule or production rule.
S : The start symbol.

G is a context free grammer if each element of P is of the form
$x \rightarrow \alpha$ where $x \in V$ & $\alpha \in (V \cup \Sigma)^*$

b) Regular Grammer :-

The language accepted by finite automata be described using a set of production known as regular grammer. The production of regular grammer are of the following form.

$A \rightarrow a$
$A \rightarrow aB$
$A \rightarrow Ba$
$A \rightarrow e$

where $a \in T(\Sigma)$ & $A, B \in V$

A language generated by a regular grammer is known as regular language.

A regular grammer can be written in two forms
i) Right Linear form :-

Sanket Chandrashekhar Harvande (19)

A right linear form will have production of the given form

$A \rightarrow a$

$A \rightarrow aB$ } Variable B in $A \rightarrow aB$ is the second symbol on the right.

$A \rightarrow \epsilon$

ii) left-linear form :-

A left linear form will have production of the following form

$A \rightarrow a$

$A \rightarrow Ba$ } Variable B in $A \rightarrow Ba$ is the first symbol on the right.

$A \rightarrow \epsilon$

c) Context free language :-

Context free grammer $G = (V, \Sigma, P, S)$ associated with language $L(G) \subseteq \Sigma^*$ as follows

$$L(G) = \{ x \in \Sigma^* \mid S \stackrel{*}{\Rightarrow} x \}$$

$L(G)$: Language generated by grammer G.
x represents set of all terminals strings which can be derieved from start symbol of G.

A language L is context free language if there is a context free grammer G such that $L = L(G)$

d) Chomsky Hierarchy of grammer :-

According to Noam Chomsky, there are four types of grammers & they are as follows :

(1) Type 0 → unrestricted grammer :-

Unrestricted grammer generates recursively enumerable language (REL). The productions have no restrictions hence it is termed as unrestricted grammer.

Sanket Chandrashekhar Harvande (19)

They generate the language that are recognized by a Turing Machine (TM.)

(2) Type 1 → Context sensitive Grommer (C S G)

CSG generates context sensitive languages (CSL). They generate the languages that are recognized by linear bounded automata. (LBA).

The production must be in the form

$$\alpha A \beta \rightarrow \alpha Y \beta$$

(3) Type 2 → Context free Grammer (CFG)

CFG generates context Free language (CFL) they generate the languages that are recognized by push down automata (PDA).

(4) Type 3 → Regular Grammer :- (RG)

Regular Grammer generates regular language. They generate the Language that are recognized by finite state machine. (FSM).

Sanket Chandrashekhar Harvande (19)

Q.2 Simplify the following grammer.

$S \rightarrow ASB \mid \epsilon$
$A \rightarrow aAS \mid a \mid aA$
$B \rightarrow SbS \mid bS \mid Sb \mid b \mid aAS \mid a \mid aA \mid bb$

→

i) Remove ϵ production

$S \rightarrow ASB \mid AB$
$A \rightarrow aAS \mid a \mid aA$
$B \rightarrow SbS \mid bS \mid Sb \mid b \mid A \mid bb$

ii) Remove unit production

$S \rightarrow ASB \mid AB$
$A \rightarrow aAS \mid a \mid aA$
$B \rightarrow SbS \mid bS \mid Sb \mid b \mid aAS \mid a \mid aA \mid bb$

iii) Lemma 1 :-

$G' = (V', \Sigma, P', S)$
$V' = \{A, B, S\}$

iv) Lemma 2 :

$G'' = \{V'', \Sigma'', P'', S)$

$V'' = \{A, S, B\}$
$\Sigma'' = \{a, b, bb\}$

$P'' = S \rightarrow ASB \mid AB$
$A \rightarrow aAS \mid a \mid aA$
$B \rightarrow SbS \mid bS \mid Sb \mid b \mid bb \mid aAS \mid aA \mid b$

Sanket Chandrashekhar Harvande (19)

Q.3 Convert CFG to CNF.

$S \rightarrow aSb \mid ab$

→ for every terminal symbol introduce a new non-terminal

If $A \rightarrow B1B2 \ldots\ldots Bm$ is a rule CFG then the rule in CNF

$A \rightarrow B1D1$ $\qquad$ $D_{n-3} \rightarrow B_{n-2} D_{n-2}$

$D_1 \rightarrow B2D2$ $\qquad$ $D_{n-2} \rightarrow B_{n-1} B_n$

$D_2 \rightarrow B3D3$

CNF : step 1:-

$S \rightarrow ASB$

$S \rightarrow AB$

$A \rightarrow a$

$B \rightarrow b$

step 2 :-

$S \rightarrow AD1$

$D1 \rightarrow SB$

$S \rightarrow AB$

$A \rightarrow a$

$B \rightarrow b$

Q.4 Convert CFG to GNF.

$S \rightarrow AA \mid 0$

$A \rightarrow SS \mid 1$

→ i) Not necessary

ii) Consider, $S = A1$, $A = A2$

$A1 \rightarrow A2A2 \mid 0$ ——— ①

$A2 \rightarrow A1A1 \mid 1$ ——— ②

iii) Rule 2 is not in GNF form so,

$A2 \rightarrow A2A2A1 \mid 0A1 \mid 1$ $\qquad$ $[A1 \rightarrow A2A2 \mid 0]$

Using Lemma 2

$B1 = 0A1$, $B2 = 1$ $\qquad$ $a_1 = A2A1$

$\therefore A2 = 0A1 \mid 1$

$A2 \rightarrow 0A1Z \mid 1Z$

Sanket Chandrashekhar Harvande (19)

Z → A2A1 A1 → A2A2 | 0

Z → A2A1Z

iv) A1 → A2A2 | 0

A2 → 0A1 | 1 | 0A1Z | 1Z

Z → A2A1 | A2A1Z

A1 → 0A1A2 | A2 | 0A1ZA2 | 1ZA2 | 0

A2 → 0A1 | 1 | 0A1Z | 1Z

Z → A2A1 | A2A1Z

v) A1 → 0A1A2 | 1A2 | 0A1ZA2 | 1ZA2 | 0

A2 → 0A1 | 1 | 0A1Z | 1Z

Z → 0A1A1 | 1A1 | 0A1ZA1 | 1ZA1

Z → 0A1A1Z | 1A1Z | 0A1ZA1Z | 1ZA1Z

Now rewrite the rules by converting back A1 = S, A2 = A

S → 0SA | 1A | 0SZA | 1ZA | 0

A → 0S | 1 | 0SZ | 1Z

Z → 0SS | 1S | 0SZS | 1ZS

Z → 0SSZ | 1SZ | 0SZSZ | 1ZSZ

Q.5 Consider the grammer.

S → 0B | 1A

A → 0 | 0S | 1AA

B → 1 | 1S | 0BB

for the string 00110101 find the following left most derivation, right most derivation & parse tree.

→ Leftmost derivation:-

S → 0B [S → 0B]

→ 00BB [B → 0BB]

Sanket Chondrashekhar Harvande (19)

→ 001B [B → 1]

→ 0011S [B → 1S]

→ 00110B [S → 0B]

→ 001101S [B → 1S]

→ 0011010B [S → 0B]

→ 00110101 [B → 1]

Rightmost derivation :-

S → 0B [S → 0B]

→ 00BB [B → 0BB]

→ 00B1S [B → 1S]

→ 00B10S [S → 0B]

→ 00B101S [B → 1S]

→ 00B1010B [S → 0B]

→ 00B10101 [B → 1]

→ 00110101 [B → 1]

Parse tree