Mid Semester Examination (6th sem), 2022

Subject Name : → Compilers.

Subject Code : → IT 3201.

Date of Examination : → 07.03.2022.

Name : → Aniket Majhi.

Examination Roll Number : → 510819019.

Gsuite ID : → 510819019.aniket@students.iiests.ac.in.

Number of sheets uploaded : → 8

0

ID → 510819019
Page → 1
Name → Aniket Majhi

(1.) The grammer is given as,

$S \rightarrow BA$

$A \rightarrow aBA \mid \epsilon$

$B \rightarrow DC$

$C \rightarrow bDC \mid \epsilon$

$D \rightarrow [S] \mid P$

Terminals = { a, b, P, [,] }

Non Terminals = { S, B, A, D, C }

☐ Calculating FIRST of the grammer : →

(i) First of Terminals,

FIRST (a) = {a}
FIRST (b) = {b}
FIRST (P) = {P}
FIRST ([) = {[}
FIRST (]) = {]}

(ii) First of Non Terminals,

FIRST (S) = FIRST (BA)

Now, FIRST (B) = FIRST (DC)

Now, FIRST (D) = FIRST ([S]) ∪ FIRST (P)
= {[} ∪ {P}
= {[, P}

AS FIRST (D) does not contains ∈.

∴ FIRST (B) = FIRST (D) = {[, P}

Similarly, FIRST (S) = FIRST (B) = {[, P}.

Now, FIRST(A) = FIRST(aBA) ∪ FIRST(ε)

= {a} ∪ {ε}

= {a, ε}

and FIRST(C) = FIRST(bDC) ∪ FIRST(ε)

= {b} ∪ {ε}

= {b, ε}

So, we got,

FIRST(S) = {Γ, P}
FIRST(A) = {a, ε}
FIRST(B) = {Γ, P}
FIRST(C) = {b, ε}
FIRST(D) = {Γ, P}

☐ Calculating FOLLOW :→

FOLLOW(S) = {$} . [S being the start symbol].

(i) S → BA.

from the above production,
(it is of type)
A → αBβ

we can say,

Follow(B) Contains FIRST(A).

NOW, FIRST(A) = {a, ε}.

As, FIRST(A) contains ε SO,

Follow(B) also Contains FOLLOW(S).

Again, FOLLOW(A) Contains FOLLOW(S)

(ii) ~~FOLLOW(B) =~~

(ii) A → aBA
(A → aBB)
FOLLOW(B) contains FIRST(A)
and As FIRST(A) includes ε,
so, it also contains FOLLOW(A).

(iii) B → DC
FOLLOW(D) contains FIRST(C)
As FIRST(C) includes ε
so, FOLLOW(D) contains FOLLOW(B).
FOLLOW(C) contains FOLLOW(B)

(iv) C → bDC
FOLLOW(D) contains FIRST(C)
As FIRST(C) includes ε
so, FOLLOW(D) contains FOLLOW(C).

(v) D → [S]
FOLLOW(S) contains FIRST(])

→ so by combining all the above 5 points we get.

FOLLOW(S) = {$,]}
FOLLOW(A) = {$,]}
FOLLOW(B) = {a, $,]}
FOLLOW(C) = {a, $,]}
FOLLOW(D) = {b, a, $,]}

∴ Parsing Table :—

	P	a	b	[	]	$
S	S→BA			S→BA		
A		A→aBA			A→ε	A→ε
B	B→DC			B→DC		
C		C→ε	C→bDC		C→ε	C→ε
D	D→P			D→[S]		

This is the corresponding parsing table.

(2.) The benefits and drawbacks if the buffer as a whole is used except dividing it into two parts :—

N We know that the lexical analyser scans the input buffer from left to right and each symbol at a time. Now in this scene it we try to use ~~two pointers~~ double buffering technique.

There are a lot of benefits :—

(i) If we use the double buffering techniques the speed of the scanning will be high.

(ii) The double buffering techniques also reduces the overhead associated with this.

The problems of using the single buffer is,

(i) The limit of the size of the buffer will increase considerably so for

(ii) scanning the rest of the lexeme the buffer has to refilled then result ~~into~~ by overwriting of initial lexeme.

To overcome this problem the two buffers are used. to store the string.

ID - 5108.19019
Page - 1
Name - Aniket Majhi

(8.)

The regular expression of the,

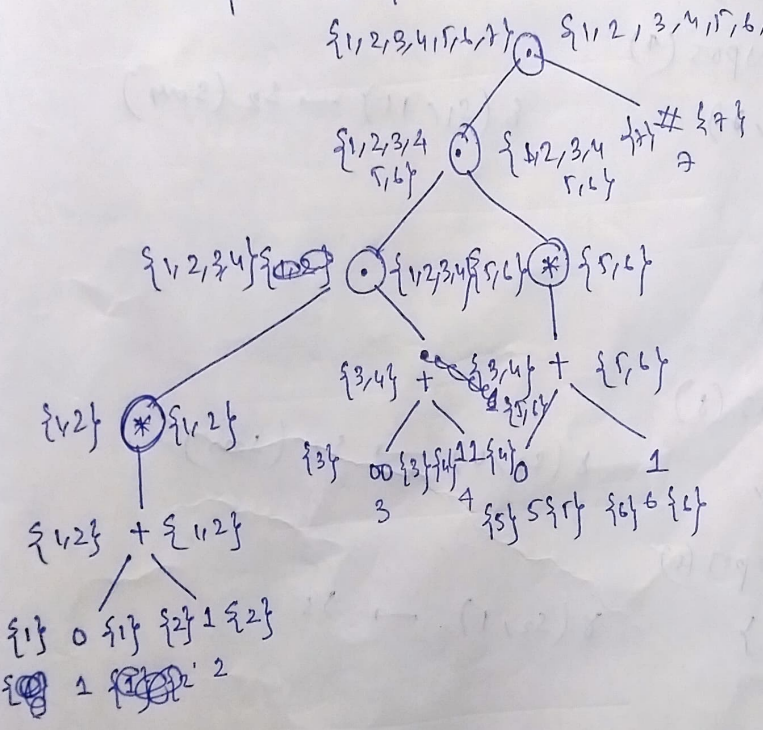
Set of languages of any combination of '0' and '1' containing atleast one double symbol is,

$$= (0+1)^*(00+11)(0+1)^*$$

The augmented regular expression is,

$$\underset{1}{(0+1)}^* \cdot \underset{3}{(00} + \underset{4}{11}) \cdot \underset{5\ 6}{(0+1)}^* \cdot \underset{7}{\#}$$

The corresponding parse tree,



∴ Followpos (1) = {1, 2, 5, 6, 7, 3, 4}

Followpos (2) = {1, 2, 5, 6, 7, 3, 4}

Followpos (3) = {5, 6, 7}

Followpos (4) = {5, 6, 7}

Followpos (5) = {5, 6}

Followpos (6) = {5, 6}

Followpos (7) = { }.

ID — 5108190[illegible]
Name - Aniket Mayekar
Page - 4

S_1 = Firstpos(root)
= {1, 2, 3, 4, 5, 6, 7}.

0: Followpos(1) ∪ Followpos(5)
= {1, 2, 3, 4, 5, 6, 7} $\delta(S_1, 0) \rightarrow S_1$

1: Followpos(2) ∪ Followpos(6)
= {1, 2, 3, 4, 5, 6, 7} $\delta(S_1, 1) \rightarrow S_1$

00: Followpos(3)
= {5, 6, 7} $\delta(S_1, 00) \rightarrow S_2$ (say)

11: Followpos(4)
= {5, 6, 7} $\delta(S_1, 11) \rightarrow S_2$ (say)

S_2 = {5, 6, 7}

0: Followpos(5)
= {5, 6} $\delta(S_2, 0) \rightarrow S_3$ (say)

1: Followpos(6)
= {5, 6} $\delta(S_2, 1) \rightarrow S_3$

S_3 = {5, 6}

0: Followpos(5) = {5, 6} = S_3
1: Followpos(6) = {5, 6} = S_3.

$\delta(S_3, 0) \rightarrow S_3$
$\delta(S_3, 1) \rightarrow S_3$

∴ The DFA.