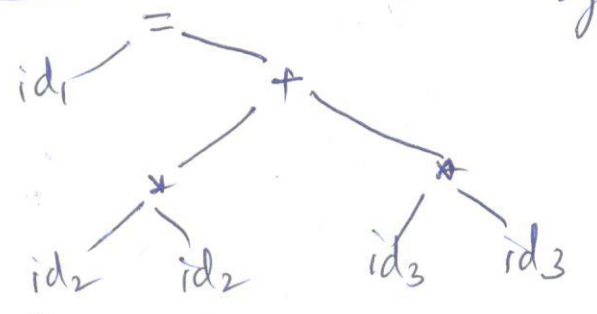
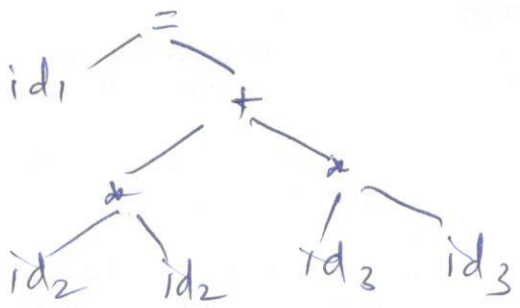
Q.No	Scheme and Solutions	Marks
1a.	input string :- bbaacabc Sequence of tokens CC C → c? means c occurs 0/1 time ∴ can start with b/a any number of times $(b/a)^*$, followed by c It accepts bbaac The second C will accept ab followed by c, abc ∴ CC will accept bbaacabc	1m 2m
b,	Val = (a*a + b*b); ↓ Lexical Analyzer → set of tokens <id,1> <=> <(> <id2> <*> <id2> <+> <id3> <*> <id3> <)>; ↓ Syntax Analysis → Parse tree / Syntax tree  	1m 1m

Q. No | Scheme and SOLUTIONS | Marks

Semantic Analysis



Symbol Table	
1	val
2	a
3	b

1m

Intermediate Code Generation

Three address code

$t_1 = id_2 * id_2$

$t_2 = id_3 * id_3$

$t_3 = t_1 + t_2$

$id_1 = t_3$

1m

Code Optimization

optimized 3AC

$t_1 = id_2 * id_2$

$t_2 = id_3 * id_3$

$id_1 = t_1 + t_2$

1m

Code Generation

Assembly lang code

```
LDA  R1, id2
MUL  R2, R1, R1
LDA  R3, id3
MUL  R4, R3, R3
ADD  R2, R2, R4
```

1m

Q.NO	Scheme and Solutions	Marks

c)

	a	b	c	f	$
S	S→adB		S→Af	S→Af	
B	B→aS	B→baB	B→ε	B→ε	B→ε
A			A→caBA	A→ε	

Follow (B) = $, c, f

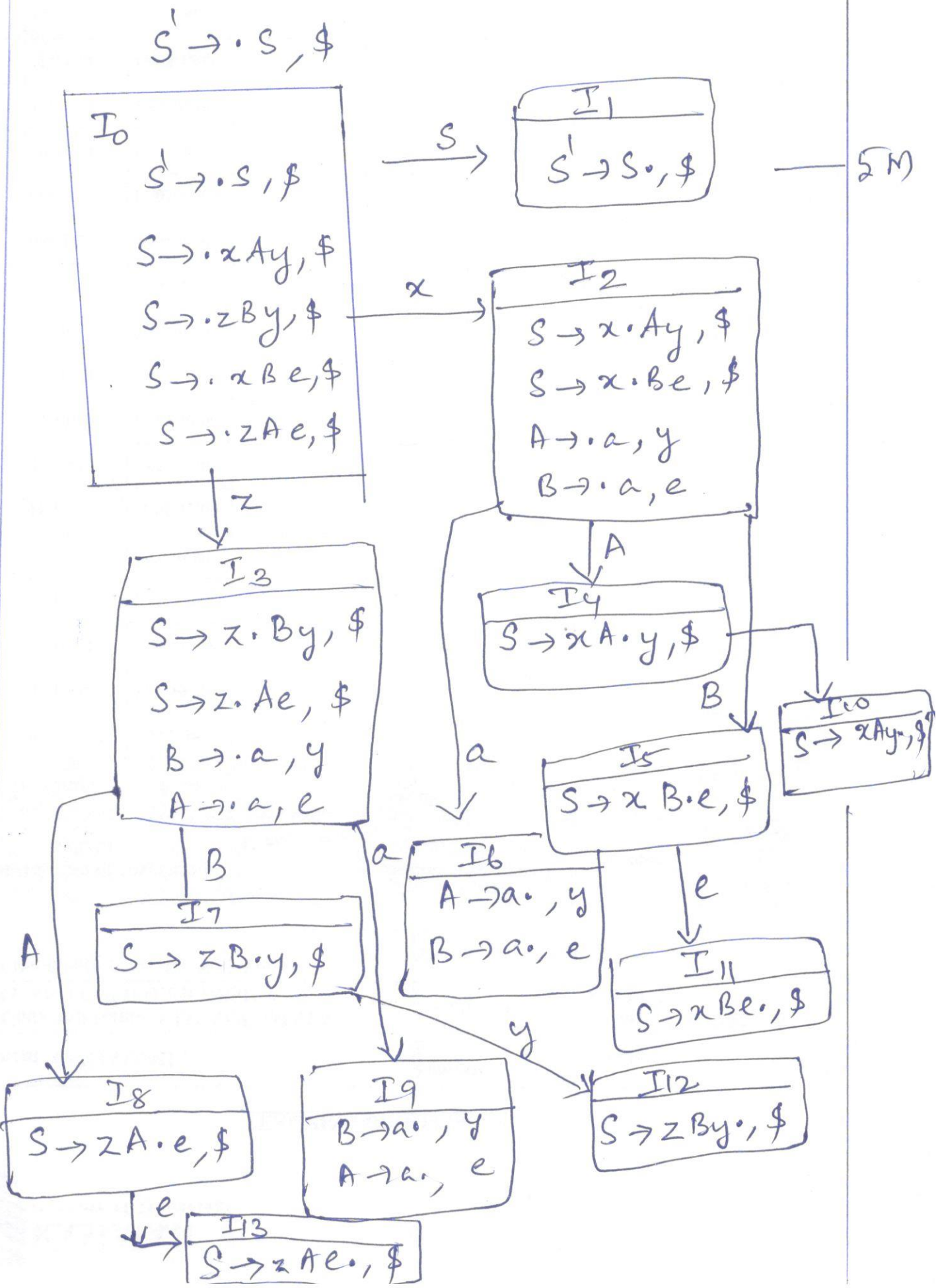
Follow (A) = f

Stack	Input	Matched
	caf$	
S $		o/p S→Af
Af $		o/p A→caBA
caBAf $		Match c
aBAf $	af$	Match a
BAf $	f $	o/p B→ε
Af $	f $	o/p A→ε
f $	f $	Match f
$	$	

Accept the input caf$

Q.NO	Scheme and Solutions	Marks

2 a, LR(1) set of items



Q.No	Scheme and Solution	Marks

2 b, i,

NT	FIRST	FOLLOW
S	b, ε	b, $
A	b, ε	b, $
B	b, ε	b, $
C	ε	b, $

— 3m (with steps)

ii,

NT	FIRST	FOLLOW
S	c, d, e	$
A	d, e, ε	d, e
B	d, e	b, d, $

— 3m (with steps)

c, i, Answer c

CFG is designed in a way that least precedence operators appears in the first definition and highest precedence operators appears in the last definition of the production grammars — 1M

Q.No	Scheme and Solutions	Marks

ii, (A) $S \to aSa \mid bS \mid \varepsilon$

If anyone definition is ε, other definitions should not start with same terminal which is there in Follow(S)

Follow(S) = \$, a ——— 1m

First(S) = a, b, ε

∴ Not LL(1)

(B) $S \to CC$, $C \to cC \mid d$ ——— 1m

It is LL(1)

(C) $A \to AA \mid (A) \mid \varepsilon$

First(A) = (, ε ——— 1m

Follow(A) = \$, (,)

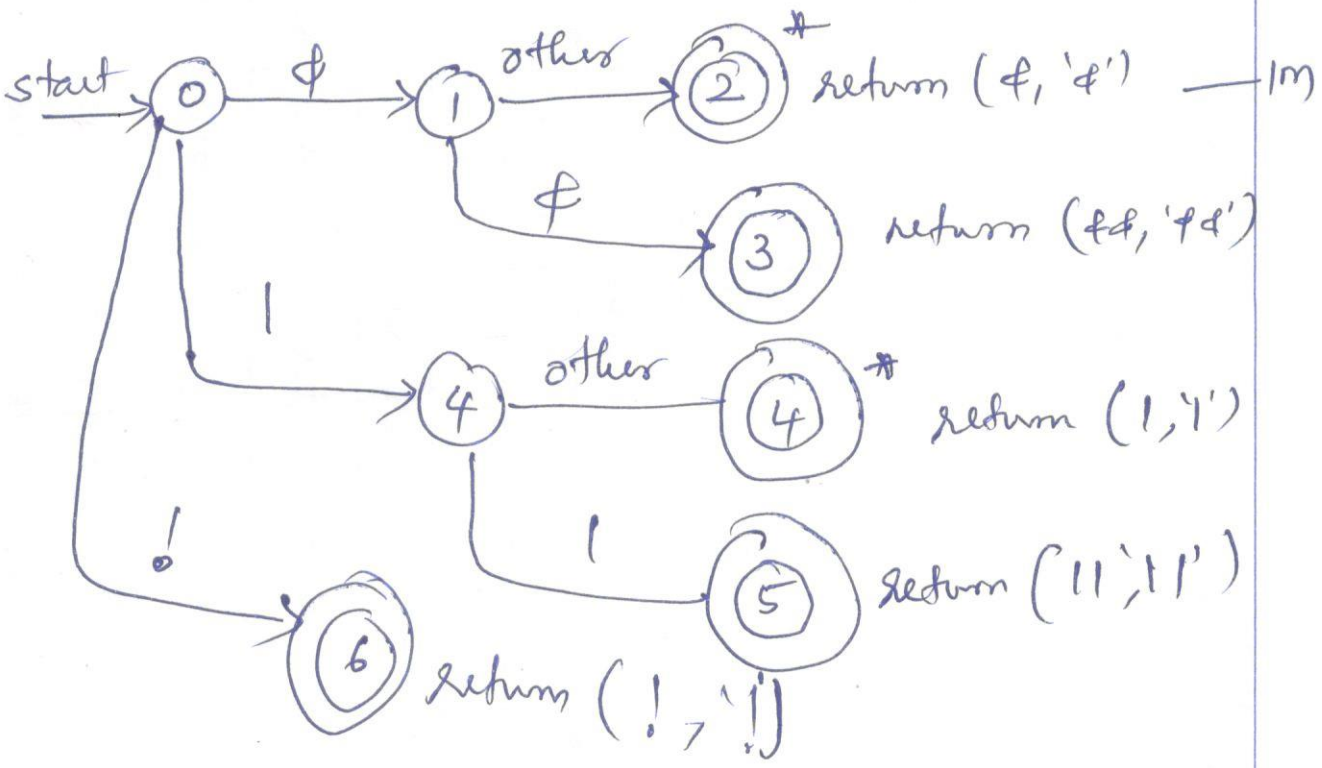
Same terminal in First and Follow

So not LL(1)

Q.NO | Scheme and Solutions

3a. All our transition diagrams are deterministic, meaning that there is never more than one edge out of a given state with a given symbol among its labels. — 3m

Transition diagrams:-



— 1m

b) ~~i → cc → (c1|c4)/(3|4|5)[1~~

i, cc → (C1|C4) L? (3|4|5) [1-8] — 1*4 =4m

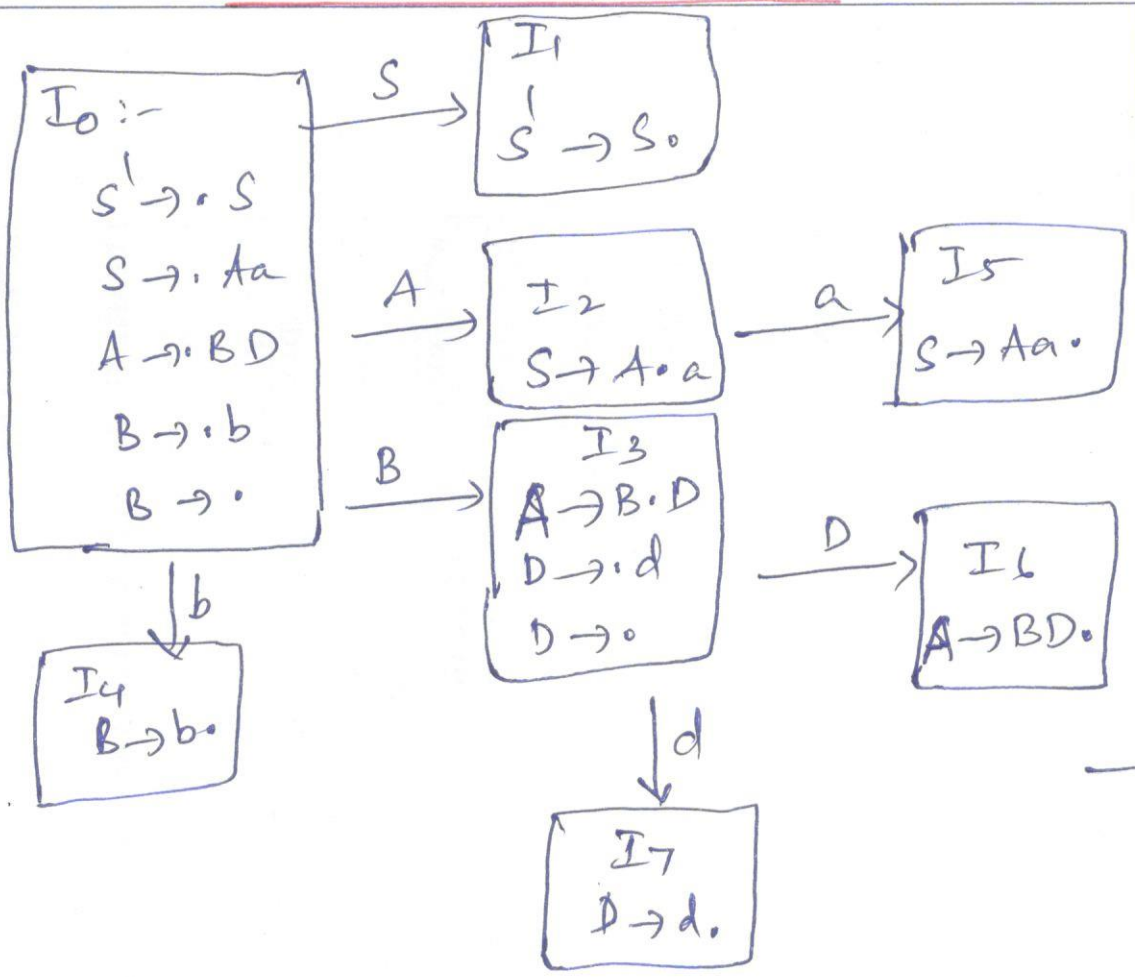
ii, . @(gmail|yahoo)/(".com"|".co.in")

iii, CRD/(4|5) 0 [1|2|5|7|8|9]
ESB/ 12 [3|4|5]

iv, (S|s)(E|e)(L|l)(E|e)(C|c)(T|t)/
(F|f)(R|r)(O|o)(M|m)/
(W|w)(H|h)(E|e)(R|r)(E|e)

Q.No	Scheme and Solutions	Marks

3c,



— 3M

States	ACTION				GOTO			
	a	b	d	$	S	A	B	D
0	r_4	s_4	r_4		1	2	3	
1				acc				
2	s_5							6
3	r_6		s_7					
4	r_3		r_3					
5				r_1				
6	r_2							
7	r_5							

— 2M

① S → Aa
② A → BD
③ B → b
④ B → ε
⑤ D → d
⑥ D → ε

FOLLOW(S) = $
FOLLOW(A) = a
FOLLOW(B) = d, a
FOLLOW(D) = a

Q.No	Scheme and Solutions			Marks
	Action	Stack	Symbols	Input
		$0		ba$
	Shift 7	$04	b	a$
	Reduce B→b	$03	B	a$
	reduce D→ε	$036	BD	a$
	Reduce A→BD	$02	A	a$
	Shift	$025	Aa	$
	Reduce S→Aa	$01	S	$

accept — 2m

write action part input matched, pop stack and advance i/p

* —— End —— *