Benha University
1st Term (January 2019) Final Exam
Class: 4th Year Students (Computer Science Major)
Subject: Compiler Theory
Course Code: CSW 456

Faculty of Computers & Informatics
Date: 10/1/2019
Time: 3 Hours
Total Marks: 75 Marks
Examiner(s): Dr. Ahmed Hassan

Answer the following questions [6 questions in 2 pages]:

Question No. 1 [10 Marks]

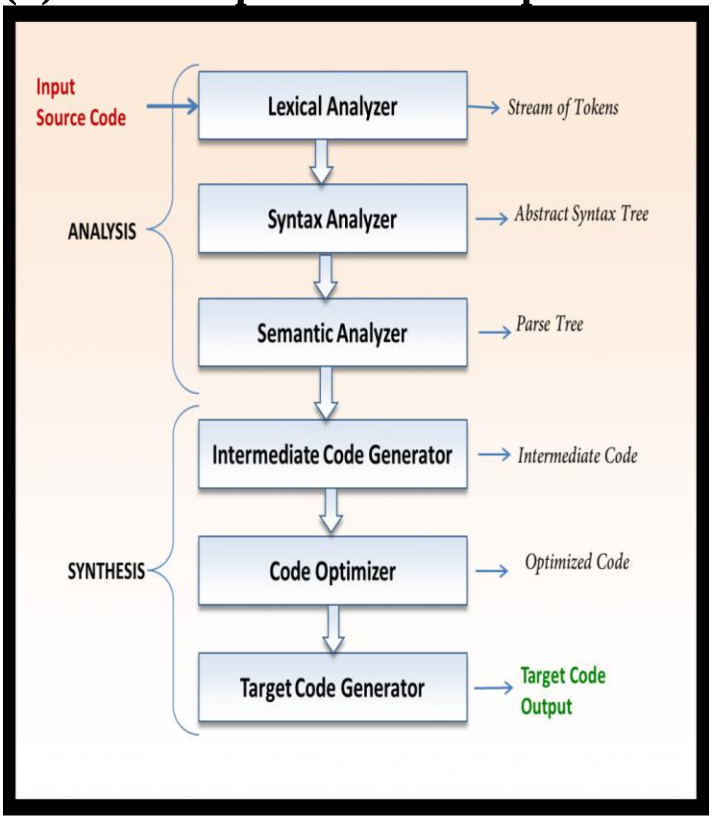
(a)Define the following:

(1) Lexemes : is a sequence of characters in the source program that matches the pattern for a token and is identified by the lexical analyzer as an instance of that token.

(2) Lexical Analyzer : read the input characters of the source program. Group them into lexemes. Produce as output a sequence of tokens for each lexeme in the source program.

(3) Parser : It takes the token produced by lexical analysis as input and generates a parse tree (or syntax tree). In this phase, token arrangements are checked against the source code grammar.

(b)Draw the phases of a compiler.

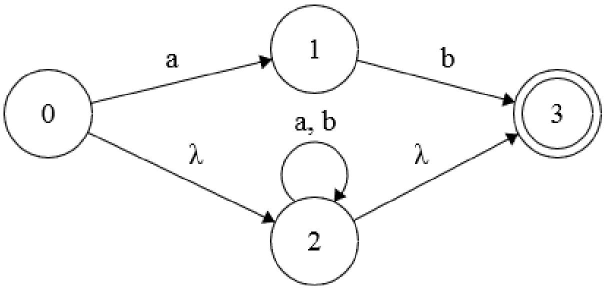


Question No. 2 **[10 Marks]**

(a) Find a *regular expression* for the language of all strings over {a,b} with odd number of "a" and ending with abb.

b*(ab*a)*b*abb OR (b + ab*a)*abb OR any one equivalent.

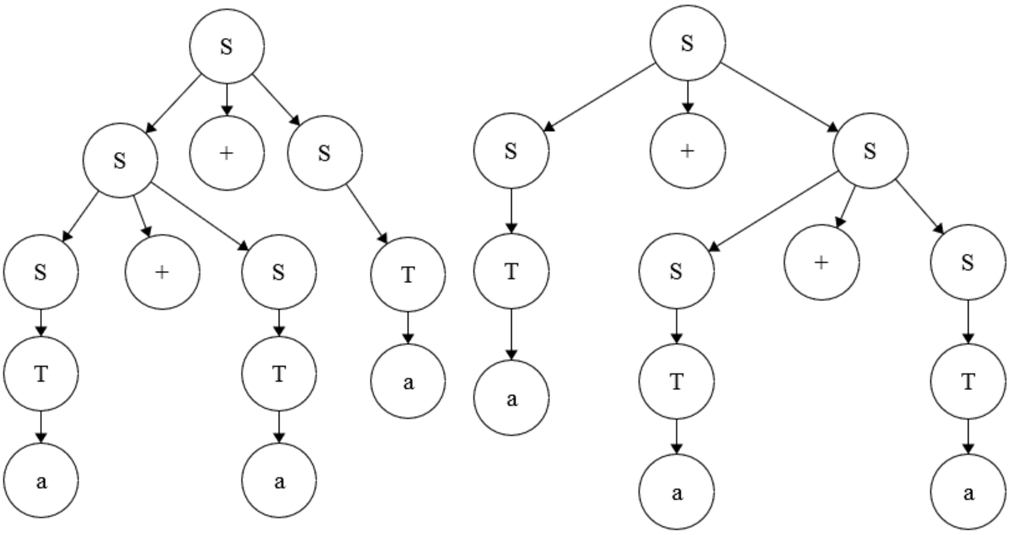
(b) convert the *regular expression* " ab + (a + b)* " to NFA.



Question No. 3 **[10 Marks]**

S → S+S | S-S | T
T → S*T | S/T | a

(a) Prove that the grammar is ambiguous.



(b) Remove the left factor then the left recursion from the grammar.

S→SA | T	S→TE	S→TE
A→+S | -S	E→AE | Ɛ	E→AE | Ɛ
T→SB | a	T→TEB | a	T→aF
B→*T | /T	A→+S | -S	F→EBF | Ɛ ⟹ F→ AEBF | Ɛ
	B→*T | /T	A→+S | -S
		B→*T | /T

Benha University
1st Term (January 2019) Final Exam
Class: 4th Year Students (Computer Science Major)
Subject: Compiler Theory
Course Code: CSW 456

Faculty of Computers & Informatics
Date: 10/1/2019
Time: 3 Hours
Total Marks: 75 Marks
Examiner(s): Dr. Ahmed Hassan

Question No. 4 [15 Marks]

A language contains three types of tokens as the following:

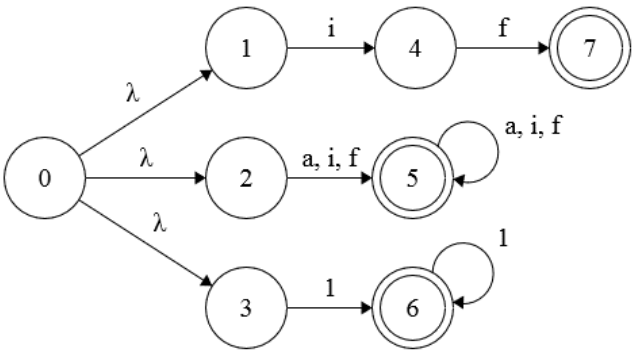
1-The first token type is the keword = { if }, **(*higher priority*)**
2-The second token type represents the identifiers which are any non-empty string over { a, i, f },
3-The third token type are the unary integers = {1, 11, 111, 1111, …}. **(*lower priority*)**

For the given language do:

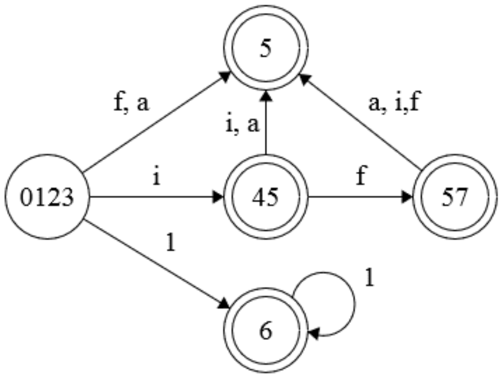
(a) Write a pattern (regular expression) to define the lexemes of each token.

if **(a|i|f)(a|i|f)*** **11***

(b) Draw an NFA scanner for your patterns from step (a).



(c) Transform the NFA scanner from step (b) into DFA scanner.



state	i	f	a	1
0123	45	5	5	6
45 ID	5	57	5	Ø
5 ID	5	5	5	Ø
57 IF	5	5	5	Ø
6 INT	Ø	Ø	Ø	6

Use the scanner from step (c) to define the tokens types and lexemes in the following input stream " ifaa111if11biif11 "

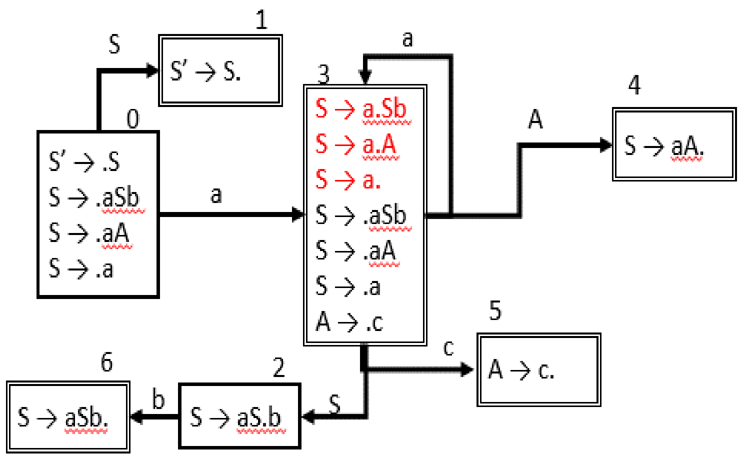
ifaa → ID
111 → INT
if → IF
11 → INT
b → Error
iif → ID
11 → INT

Question No. 5 [15 Marks]

For the following grammar:

$$S \rightarrow aSb \mid aA \mid a$$
$$A \rightarrow c$$

(a) Construct the LR(1) parser table.



LR(1)	a	b	c	$	S	A
0	S3				1	
1				acc		
2		S6				
3	S3	r3	S5	r3	2	4
4		r2		r2		
5		r4		r4		
6		r1		r1		

Non-Terminal	Follow
S	$, b
A	$, b

(b) Use the parser table to check the string: aacb

Stack	Input	Action
$0	aacb$	S3
$0a3	acb$	S3
$0a3a3	cb$	S5
$0a3a3c5	b$	r4 and goto 4
$0a3a3A4	b$	r2 and goto 2
$0a3S2	b$	S6
$0a3S2b6	$	r1 and goto 1
$0S1	$	accepted

Benha University
1st Term (January 2019) Final Exam
Class: 4th Year Students (Computer Science Major)
Subject: Compiler Theory
Course Code: CSW 456

Faculty of Computers & Informatics
Date: 10/1/2019
Time: 3 Hours
Total Marks: 75 Marks
Examiner(s): Dr. Ahmed Hassan

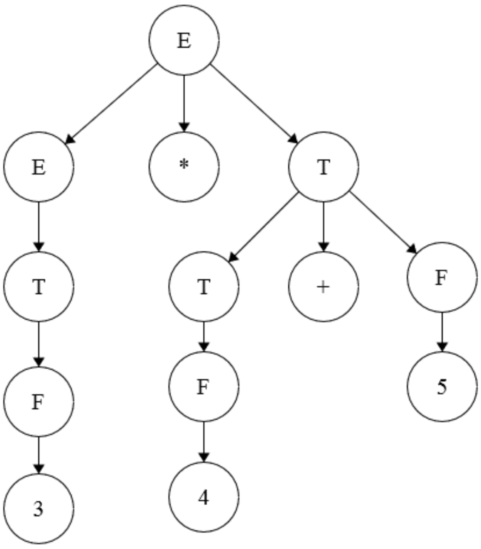
(c) Is it LR(0) grammar? Why? No, state 3 has both shift and reduce.

Question No. 6 [15 Marks]

Use the following semantic rules to generate the intermediate code for

3*4+5

And what is the result?



INDEX	VALUE
0	**Int 3**
1	**Int 4**
2	**Int 5**
3	**1 + 2**
4	**0 * 3**

T0 = 3
T1 = 4
T2 = 5
T3 = T1 + T2
T4 = T0 * T3

Result = 3*(4+5) = 3*9 = 27

GOOD LUCK,