American University of Beirut, Faculty of Arts and Sciences
Department of Mathematics

Math 274
Final Exam

time: 2 hrs Fall 1996-97

Name: _____ #id: _____

G. Jalloul

I. Describe how lex generates lexical analyzers for regular expressions. Illustrate this on the regular expression a (alb)*.

II. Consider the following grammar

Decl → Type id , List;
List → id, List | ε
Type → integer | real

1. Is the given grammar LL(1)? Prove or disprove. If it is not LL(1) modify it so that it becomes LL(1).

2. Is the given grammar operator precedence? Explain. If it is not suggest modifications that will make it operator precedence.

3. Identify programming language constructs that have been included in grammars as a result of using operator precedence parsing technique.

III. Consider the following augmented grammar for a form of relational expressions:

1 R' → R
2 R → E<E
3 R → E>E
4 E → digit
5 E → R

and the following subset of the collection of LR(0) items

I_0 : R → .R
R → .E<E
R → .E>E
E → .digit
E → .R

I_1 : goto (I_0 , R)
R → R.
R → R.

I_2 : goto (I_0 , E)
R → E.<E
R → E.<E

I_3 : goto (I_0 , digit)
E → digit.

I_4 : goto (I_2 , <)

I_5 : goto (I_2 , >)

I_6 : goto (I_4 , E)
R → E<E.
R → E.<E
R → E.>E

I_7 : goto (I_5 , E)
R → E>E.
R → E.<E
R → E.>E

1. Find the states I_4 and I_5

2. Find the follow sets of the non terminal symbols of the grammar

3. Is the given grammar SLR? Explain

4. Define the following entries of the SLR parse table to give < a higher precedence than >, make < right associative and > left associative. Explain

action [I_6 , <] =

action [I_6 , >] =

action [I_7 , <] =

action [I_7 , >] =

5. Modify the given set of items to form the canonical collection of LR (1). Is the grammar LR(1)

I_0 : R → . R
R → . E<E
R → . E>E
E → . digit
E → . R

I_1 : goto (I_0 , R)
R → R.
R → R .

I_2 : goto (I_0 , E)
R → E . <E
R → E . <E

I_3 : goto (I_0 , digit)
E → digit .

I_5 : goto (I_2 , >)

I_6 : goto (I_4 , E)
R → E<E .
R → E . < E
R → E . > E

I_7 : goto (I_5 , E)
R → E > E .
R → E . < E
R → E . > E

L_4 : goto (I_2 , <)

6. Find the LALR(1) collection of items. Is the grammar LALR

7. What is the advantage of using the LALR parsing technique over the canonical LR , knowing that the LR technique is more powerful.

8. Compare the LL and LR parsing techniques in terms of their parsing approach, ease of implementation and the set of grammars they recognize

IV. Consider the following grammar for boolean expressions:

E → E and E
| E or E
| not (E)
| true
| false

1. Give a syntax directed definition that returns the value of E in the attribute val. Is val a synthesized or inherited attribute? Explain

2. Write a yacc program that will evaluate boolean expressions- generated by the above grammar -- with the following precedence rules:

not	highest precedence and is right associative
and	second precedence and is left associative
or	least precedence and is left associative

3. What are the steps and corresponding instructions need to produce an executable object code equivalent to a yacc program

V.

1. The case statement of Pascal seems to be a redundant construct since the effect of executing a case statement can be programmed using an if statement. Give a justification for why including the case in the grammar might be beneficial.

2. What is meant by the Front and Back Ends of a compiler?

3. What is meant by a cross compiler? Give an example

4. Assume that you have access to a Pascal compiler, that runs on intel 486 and produces intel 486 code. Explain how you might obtain move your compiler for SPL – without writing one – to an intel 486.