UNIVERSITY *of* LIMERICK

OLLSCOIL LUIMNIGH

COLLEGE *of* INFORMATICS *and* ELECTRONICS

Department of Computer Science
and Information Systems

End-of-Semester Assessment Paper

Academic Year:	**2005/06**	Semester:	**Spring**
Module Title:	**P.L.T.**	Module Code:	**CS4158**
Duration of Exam:	**2½ Hours**	Percent of Total Marks:	**70**
Lecturer(s):	**Jim Buckley**	Paper marked out of :	**70**

Instructions to Candidates:

- Answer question 1.
- Answer 2 of the remaining 3 questions.
- Question 1 carries 30 marks
- The other questions carry 20 marks.

Q1. a) Form 'LEX' regular expressions for:

- A variable name that must be made up of one of more letters, followed by zero, one or more numbers;
- A variable name, starting with a letter, composed of letters and numbers, but with no two numbers appearing consecutively. So the expression should allow y5t7r but should not allow y5t43f.

5 Marks

b) Draw a transducer for each of the regular expression constructed in 'a' above, and from them create transition tables

5 Marks

c) Evaluate and comment on the following C statements with respect to operator priority, associativity, fixity and arity:

- int a=-b=-3; // where a = 10, b = 4
- --j++;
- data=*ptr+y;

5 Marks

d) Classify the following 2 grammars as either Context Free Grammars, Context Sensitive Grammars or Regular Grammars (Explaining your reasons)

Grammar 1	Grammar 2
E -> aB	E -> gH
B -> dG	H -> RT
G -> c	R -> k
G -> λ	T -> p

5 Marks

e) Explain the term 'configuration' in the context of bottom-up parsing

5 Marks

f) Explain how LEX and YACC have a common understanding of terminal vocabulary

5 Marks

Q2. a) Describe, by building an LR(0) finite state automata for the following grammar, why it is LR(1)

S->E$
E->E+T
E->T
T->T*p

6 Marks

b) Build the LR(1) finite state automata for the grammar, showing how the problem has been resolved

10 Marks

c) Briefly describe the structure of a parser with respect to its component parts, their roles and their interrelationships

4 Marks

Q3. a) Describe, using examples, two conditions that conflict with a grammar being LL(1) and, for each condition, describe how they can be overcome.

8 Marks

b) Using the LL(1) grammar below, calculate the predict set for each production:

E -> (E)
E -> V H
E -> + E
V -> id C
C -> (E)
C -> λ
H -> +E
H -> λ

10 Marks

c) Use this predict set to form a LL(1) Parse table for the grammar.

2 Marks

Q4. a) Create a Post Production system that generates language instances which:

- Are made up of 1's and 0's
- Have even numbers of 0's
- Odd numbers of 1's

The 0's and 1's in each language instance can be in any order. So, for example, valid language instances include: 11001, 10101 and 11110110001. Language instances exclude: 110000, 100011 and 11101100.

8 Marks

b) Illustrate how you would create the language instance 11110110001 with this Production system

8 Marks

c) Describe Chomsky's core contributions to compiler theory

4 Marks