



DUBLIN CITY UNIVERSITY

SEMESTER ONE EXAMINATIONS 2009

MODULE TITLE: Compiler Construction 1
MODULE CODE: CA448
COURSE: BSc. in Computer Applications (Software Engineering)
YEAR: 4
EXAMINERS: Dr. F. Bannister
Dr. P. Gibson
Dr. G. Hamilton, Ext no. 5017
TIME ALLOWED: 2 Hours
INSTRUCTIONS: Please answer ALL questions.
All questions carry equal marks

Requirements for this paper
Please tick (X) as appropriate

- | | |
|--------------------------|------------------------------|
| <input type="checkbox"/> | <i>Log Table</i> |
| <input type="checkbox"/> | <i>Graph Paper</i> |
| <input type="checkbox"/> | <i>Attached Answer Sheet</i> |
| <input type="checkbox"/> | <i>Statistical Tables</i> |
| <input type="checkbox"/> | <i>Floppy Disk</i> |
| <input type="checkbox"/> | <i>Actuarial Tables</i> |

THE USE OF PROGRAMMABLE OR TEXT STORING CALCULATORS IS EXPRESSLY FORBIDDEN

PLEASE DO NOT TURN OVER THIS PAGE UNTIL YOU ARE INSTRUCTED TO DO SO

Question 1**[10 marks]**

A floating point number starts with an optional minus sign, followed by zero or more decimal digits, followed by a decimal point, followed by one or more further decimal digits, followed by an optional exponent. An exponent starts with the letter E, followed by an optional minus sign, followed by one or more decimal digits. Express the syntax of floating point numbers as a regular expression.

Question 2**[10 marks]**

Construct a deterministic finite state automaton for recognising floating point numbers as described in Question 1.

Question 3**[10 marks]**

Show that the following grammar for arithmetic expressions is ambiguous:

$$E \rightarrow E + E \mid E * E \mid \sim E \mid (E) \mid \text{num}$$

Question 4**[10 marks]**

Determine whether or not the following grammar is LL(1):

$$S \rightarrow XYZ$$

$$X \rightarrow a$$

$$X \rightarrow Zb$$

$$X \rightarrow \epsilon$$

$$Y \rightarrow c$$

$$Y \rightarrow \epsilon$$

$$Z \rightarrow d$$

Question 5**[10 marks]**

Convert the following grammar into an LL(1) grammar which recognises the same language:

$$\text{Stat} \rightarrow \text{id} := \text{Exp}$$

$$\text{Stat} \rightarrow \text{id} (\text{Arglist})$$

$$\text{Arglist} \rightarrow \text{Arglist} , \text{Exp}$$

$$\text{Arglist} \rightarrow \text{Exp}$$

$$\text{Exp} \rightarrow \text{id} (\text{Arglist})$$

$$\text{Exp} \rightarrow \text{id}$$

Question 6**[10 marks]**

Determine whether or not the following grammar is LR(0):

$S \rightarrow Xb$
 $S \rightarrow Y$
 $X \rightarrow aY$
 $Y \rightarrow a$
 $Y \rightarrow aX$

Question 7**[10 marks]**

Determine whether or not the following grammar is SLR(1):

$S \rightarrow Aa$
 $S \rightarrow bAc$
 $S \rightarrow dc$
 $S \rightarrow bda$
 $A \rightarrow d$

Question 8**[10 marks]**

Determine whether or not the grammar in Question 7 is LR(1).

Question 9**[10 marks]**

A number is of the form $n_1 'E' n_2$ where n_1 and n_2 are signed integers separated by the letter 'E'. The value of such a number is $n_1 \times 10^{n_2}$. The grammar for these numbers is as follows:

$\text{Num} \rightarrow \text{SignedInt 'E' SignedInt}$
 $\text{SignedInt} \rightarrow \text{Sign Int}$
 $\text{Sign} \rightarrow +$
 $\text{Sign} \rightarrow -$
 $\text{Int} \rightarrow \text{Int Digit}$
 $\text{Int} \rightarrow \text{Digit}$

Add attributes and rules that calculate the value of a number.

Question 10**[10 marks]**

Consider the following grammar for simple expressions:

$$\begin{aligned} E &\rightarrow E + T \mid E - T \mid T \\ T &\rightarrow T * F \mid T / F \mid F \\ F &\rightarrow (E) \mid \text{id} \end{aligned}$$

Construct an abstract syntax tree for the following expression:

$$x / (x + y) * (x + y)$$

Convert this abstract syntax tree into an equivalent directed acyclic graph by eliminating all common subexpressions.