1

	FINALTERM EXAMINATION SPRING 2006 CS606 - COMPILER CONSTRUCTION (Session - 1)	Marks: 75 Time: 120min

StudentID/LoginID: _____

Student Name: _____

Center Name/Code: _____

Exam Date: Wednesday, August 23, 2006

Please read the following instructions carefully before attempting any of the questions:

1. This examination is closed book, closed notes, closed neighbors.
2. Do not ask any questions about the contents of this examination from anyone.
   a. If you think that there is something wrong with any of the questions, attempt it to the best of your understanding.
   b. If you believe that some essential piece of information is missing, make an appropriate assumption and use it to solve the problem.
3. Some of the examination consists of multiple-choice questions. Choose only one choice as your answer.
   a. If you believe that two (or more) of the choices are correct for a particular question, choose the best one.
   b. On the other hand, if you believe that all of the choices provided for a particular question are wrong then select the one that appears to you as being the least wrong.

****WARNING: Please note that Virtual University takes serious action against unfair means. Anyone found involved in cheating will get an `F` grade in this course.**

For Teacher's use only											
Question Marks	1	2	3	4	5	6	7	8			Total

Question No: 1 (Marks: 2) - Please choose one

Circle the choice that is a regular expression for the set of all strings over the alphabet {a} that an even number of a's.

- aa*
- (aa)*
- aa*a
- a(aa)*

Question No: 2 (Marks: 2) - Please choose one

A shift reduce parser performs reductions in the reverse order specified by a left-most

- False
- True
- None of these

Question No: 3 (Marks: 2) - Please choose one

Consider the context-free grammar with terminal symbols a, b, c,
symbols A and B where A is the start symbol and productions

$A \rightarrow BAB \mid a$

$B \rightarrow b \mid c \mid \varepsilon$

Which of the following strings is not in the language of the grammar?

- abb
- bcacb
- bbccab
- bcab
- bcacba

Question No: 4 (Marks: 2) - Please choose one

Dotted items ($T \rightarrow \alpha \bullet \beta$) record which part of a token has already been
There are two kinds of basic items:**shift items**and**reduce** .
integer→ (• [0-9]) +

-This is a shift item.

- False
- True
- None of these

Question No: 5 (Marks: 2) - Please choose one

A grammar with a FIRST/FIRST conflict can be made LL(1) by *only* applying**left factoring** that is, no substitution and left-recursion removal are needed.

- False
- True
- None of these

Question No: 6 (Marks: 20)

Consider the grammar

$S \rightarrow$ **a** B C **d** | **d** C B **e**
$B \rightarrow$ **b** B | ε
$C \rightarrow$ **c a** | **a c** | ε

(a) What are the terminals, non-terminals, and the start symbol for the grammar? [02

(b) Give the parse tree for the input string**abbcad**. [02 Points]

(c) Compute the First sets and Follow sets for each of the nonterminals in the grammar.
[04 Points]

(d) Construct an LL(1) parsing table for the grammar. [12 Points]

Question No: 7 (Marks: 30)

Consider the following grammar over the terminals +, – (the negation operator) and *id*.

4

$S \rightarrow E$

$E \rightarrow E+E \mid \quad -E \mid id$

Below is a partial DFA for this grammar.

(a) Complete the above DFA. You need to do the following: [21 points]

- Complete state 0 by performing closure on the item listed.
- Fill in all elements of states 1 and 5, and the lookahead items in states 3, 4 and 6.
- Fill in the missing transition labels on all edges.
- Write the necessary "reduce by ..." labels on states.

(b) For each state with a conflict, list the state, the lookahead token, and the type of conflict (i.e. shift-reduce conflict, or reduce-reduce conflict). [4 points]

c) Show that the given grammar is ambiguous (Hint: use the parse tree) [5 points]

```
E → E + T | T
T → T + E | id
```

Question No: 8 (Marks: 15)

Add semantic rules to the following grammar to compute the `rm`, whose value is the rightmost terminal in the string we parsed. For example, if the string parsed *zxyxy*, `S.rm` would be *y*.

Note: subscripts in the grammar below are only to distinguish multiple instances of the same

5

nonterminal.

```
S → A            {S.rm =

A → A1 x y
  | B A1 y
  | C

B → B1 z
  | x

C → w C1
  | y C1
  | z
```