B.E / B.Tech (Part--time) DEGREE END SEMESTER EXAMINATIONS, NOV / DEC 2011

INFORMATION TECHNOLOGY

FOURTH SEMESTER

IT9251 –FORMAL LANGUAGES AND AUTOMATA

REGULATIONS 2008

**Time : 3 hrs** **Max Mark : 100**

**Answer ALL Questions**

**Part – A ( 10x2 = 20Marks**

1. Prove that 1+2+.....+n = n (n+1) by induction.

2

2. Define DFA and NFA. Give examples.
3. What is a regular expression? Write a regular expression for variables in C.
4. Write the closure properties of regular languages.
5. What is an ambiguous grammar? Give an example.
6. Define Deterministic Pushdown Automata.
7. Distinguish between context free grammars and context free languages.
8. Write the applications of Turing machines.
9. Define LL(1) grammar.
10. What is a bottom up parser?

**Part – B ( 5x 16 = 80 Marks)**

11 (i) Explain the procedure used for converting to a DFA from NFA. (8)

(ii) What is Epsilon closure? (8)
Explain the method of computing Epsilon closure.

12.a(i) Explain the procedure used to convert DFA's to regular expressions. (6)

(ii) Show that every language defined by a regular expression is also defined by a finite automation (10)

**(OR)**

b(i) State and explain the Pumping lemma for regular languages. (8)

(ii) Explain the techniques used for minimization of automata. (8)

13.a(i) Write a grammar for simpler expression and assignment statements in C. Draw the parse trees for it. Explain the left most derivation using an example. (16)

**(OR)**

b. Define Push Down Automation. Explain it. Design a PDA for checking whether a given string is a palindrome. State the applications of PDA. (16)

14.a(i) Explain the production rules, epsilon and unit productions with examples. (6)
(ii) Put the following grammar in Chomsky Normal Form. (10)

$S \rightarrow ASB / \in$
$A \rightarrow aAS/a$
$B \rightarrow Sbs/A/bb$

**(OR)**

b (i) Explain Turing Machines. (8)
(ii) Construct a Turing machine to check palindromes. (8)

15.a(i) Explain the top down parsing algorithm. (8)
(ii) Explain the procedures used to find FIRST( ) and FOLLOW ( ) (8)

**(OR)**

b (i) Explain LR (0) grammar with an example. (8)
(ii) Explain the implementation of a bottom-up parser. (8)