Seat No.: _____ Enrolment No._____

GUJARAT TECHNOLOGICAL UNIVERSITY

M. E. Sem. – IInd - Examination – June/July- 2011

Subject code: 1720202

Subject Name: Design of Language Processors

Date:24/06/2011 **Time: 10:30 am – 01:00 pm**

Total Marks: 70

Instructions:

1. **Attempt all questions.**
2. **Make suitable assumptions wherever necessary.**
3. **Figures to the right indicate full marks.**

Q.1 **(a)** Which data structures can be used to implement symbol table? Discuss them with pros and cons of each. **07**

(b) Write a regular expression for **07**

1. The set of strings of 0's and 1's not containing 101 as a substring.
2. The set of strings with even number of a's followed by odd number of b's.

Q.2 **(a)** Define token, pattern and lexeme. Give two examples of each to differentiate them. **07**

(b) Construct NFA for following regular expression and convert it into DFA. **07**

$a^+ (a | b) b^* a \#$

OR

(b) Construct DFA for following regular expression without constructing NFA and optimize the same. **07**

$(a | b)^* a (a | b) \#$

Q.3 **(a)** Define LL(1) grammar. Give example of each case where grammar is LL(1) and not LL(1). Justify the same. **07**

(b) Construct a CLR parsing table for following grammar. **07**

$S \rightarrow Ba \mid bBc \mid dc \mid bda$

$B \rightarrow d$

OR

Q.3 **(a)** What is called handle? Justify that parsing a string using SR parser is same as rightmost derivation in reverse, if a proper handle is selected. Give appropriate example. **07**

(b) Construct a SLR parsing table for following grammar. **07**

$S \rightarrow xAy \mid xBy \mid xAz$

$A \rightarrow aS \mid b$

$B \rightarrow b$

Q.4 **(a)** Write three address code for following expression and generate final code by clearly showing register descriptors and address descriptors. **07**

$x = a / (b + c) - d * (e + f)$

(b) List different code improving transformations and explain any three of them. **07**

OR

Q.4 **(a)** Explain peephole optimization with its characteristics and examples. **07**

(b) What is DAG? How does it differ from syntax tree? Draw syntax tree and DAG for the following expression **07**

$$a + a * (b - c) + (b - c) * d$$

Also write syntax directed definition to construct DAG of above expression.

Q.5 **(a)** 1. Define synthesized attribute and inherited attribute. Give example for each justifying the definition. **07**

2. What is called annotated parse tree? Explain with example.

(b) Explain synthesis and analysis phase of assembler. Also discuss pass structure of assembler. **07**

OR

Q.5 **(a)** List the major steps of relocation and linking algorithms. Explain in brief. **07**

(b) Explain macro call and macro expansion. How do macros differ from subroutine? **07**

****