

Code No: 115DR**JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD****B. Tech III Year I Semester Examinations, March - 2017****AUTOMATA AND COMPILER DESIGN****(Information Technology)****Time: 3 hours****Max. Marks: 75****Note:** This question paper contains two parts A and B.

Part A is compulsory which carries 25 marks. Answer all questions in Part A. Part B consists of 5 Units. Answer any one full question from each unit. Each question carries 10 marks and may have a, b, c as sub questions.

PART - A**(25 Marks)**

- 1.a) Define e-closure with example. [2]
- b) What are the issues in lexical analysis? [3]
- c) Define 'Handle Pruning' in bottom-up parsing. [2]
- d) Define annotated parse tree. [3]
- e) What is type expression? [2]
- f) What is meant by structural equivalence? [3]
- g) What is a Basic Block? [2]
- h) What is an activation record for a procedure? [3]
- i) What is meant by assembly language code? [2]
- j) Define DAG with example. [3]

PART - B**(50 Marks)**

2. Explain various phases of compiler and trace it with the program segment
 $x = a + b * 60$. [10]

OR

- 3.a) Draw NFA for the regular expression for ab^*/ab .
- b) What is Left Recursion? Eliminate left recursion from the following grammar:
 $A \rightarrow Ac/Aad/bd/c$. [5+5]

4. Construct canonical parsing table for the grammar given below [10]
 $S \rightarrow Aa/bAc/bBa$
 $A \rightarrow d$
 $B \rightarrow d$

OR

- 5.a) Explain in detail about syntax directed translation.
- b) Write the syntax directed translation for declarations. [5+5]
6. Explain the specification of a simple type checker. [10]

OR

- 7.a) Consider following grammar:
 $E \rightarrow \text{num}.\text{num}/\text{literal}/\text{num}/E\%E/E+E/E/E/*E/E[E]$
Construct semantic rules to find type of expression.
- b) Give an algorithm to test the equivalence of C types. [5+5]

8. Describe the method to obtain faster access to nonlocals. [10]

OR

9. Explain different principles source of optimization technique with suitable examples. [10]

10.a) Explain the issues in design of code generator.

b) Explain simple code generator with suitable example. [5+5]

OR

11. Explain in detail register allocation and assignment. [10]

---ooOoo---