

Code No: 136AQ

JAWAHARLAL NEHRU TECHNOLOGICAL UNIVERSITY HYDERABAD

B. Tech III Year II Semester Examinations, December - 2019

COMPILER DESIGN

(Common to CSE, IT)

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) What is Language Processor? [2]
- b) Differentiate between pass one and pass two of a compiler. [3]
- c) Define context free grammar. [2]
- d) What do you mean by ambiguous grammar? Give example. [3]
- e) Convert the expression $a = b * -c + b * -c$ into three Address statements. [2]
- f) What is an attribute? Explain different types of attribute? [3]
- g) List the different storage allocation strategies. [2]
- h) What are the sub problems in register allocation strategies? [3]
- i) Give the applications of DAG. [2]
- j) What is the role of flow graph optimizing a compiler? [3]

PART - B (50 Marks)

2. With a neat diagram explain the phases of a compiler in detail. [10]
- OR
3. Give an example to design an automaton from a given regular expression. [10]
4. Test whether the grammar is LL(1) or not and construct a predictive parsing table for it
 $S \rightarrow iEtS | iEtSeS | a \quad E \rightarrow b$ [10]
- OR
5. Construct an LALR parsing table for the following grammar: [10]
 $E \rightarrow E+T | T \quad T \rightarrow T*F | F \quad F \rightarrow (E) | id$
6. Write syntax directed definition for flow of control statements. [10]
- OR
7. Write the three address code sequence for the $d := (a-b) + (a-c) + (a-c)$. [10]
8. Give example to explain peephole optimization technique in detail. [10]
- OR
9. Write in detail about the issues in the design of code generator. [10]
10. With an example explain common sub expression and dead code elimination methods in detail. [10]
- OR
11. What is DAG and flowgraph? Explain their role in compilation process. [10]