BTS-VI-04.18-0671 Reg. No. | | | | | | | | |

C

B.Tech. Degree VI Semester Examination April 2018

CS/IT 15-1602 COMPILER CONSTRUCTION

(*2015 Scheme*)

Time : 3 Hours Maximum Marks : 60

PART A

(Answer *ALL* questions)

(10 × 2 = 20)

I. (a) Define tokens, patterns and lexemes.

(b) Count the number of tokens in the following C statement.
/*abc*/printf("What's up %d", ++&&***a);

(c) Write the various error recovery strategies in Lexical analyzer.

(d) Explain the syntax analysis phase in a compiler.

(e) Write the algorithm for computing FOLLOW(X).

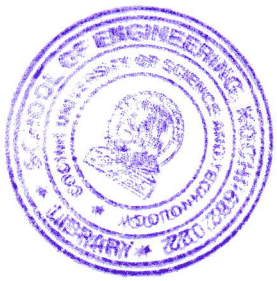
(f) What is an operator grammar? Give example.

(g) Explain the use of Directed Acyclic Graph in syntax directed definition.

(h) Explain the use of dependency graph in the translation of an input string.

(i) Write the 3 address code corresponding to the expression.
a : = b*-c+b*-c.

(j) Explain methods of translating Boolean expressions.



PART B

(4 × 10 = 40)

II. What is a compiler? Explain different phases of a compiler with a neat diagram.

OR

III. What is the role of transition diagram in the context of a lexical analyzer? Draw the transition diagram for:
(i) Relational operators (ii) Unsigned numbers.

IV. Compute the FIRST and FOLLOW of the grammar and construct the predictive parsing table.
E → T E'
E' → + T E' | ε
T → F T'
T'→ * F T' | ε
F → (E) |id

OR

V. Discuss the working of an LR purser.

VI. With an example explain the bottom-up evaluation of S attributed definition.

OR

VII. Explain different storage allocation strategies.

VIII. Explain the principal sources of optimization in a code. Illustrate with example.

OR

IX. What are the different issues in the design of a code generator?

***