

BIRLA INSTITUTE OF TECHNOLOGY, MESRA, RANCHI  
(MID SEMESTER EXAMINATION)

CLASS: BE  
BRANCH: CSE

SEMESTER: VI  
SESSION : SP/2020

SUBJECT : CS6105 COMPILER DESIGN

TIME: 1.5 HOURS

FULL MARKS: 25

**INSTRUCTIONS:**

1. The total marks of the questions are 30.
  2. Candidates may attempt for all 30 marks.
  3. In those cases where the marks obtained exceed 25 marks, the excess will be ignored.
  4. Before attempting the question paper, be sure that you have got the correct question paper.
  5. The missing data, if any, may be assumed suitably.
- 

- Q1 (a) Why programming is not preferred in low-level language? Differentiate between compiler and interpreter. [2]  
(b) Highlight some *important features* of any standard compiler. What are the *challenges* in designing compiler? [3]
- Q2 (a) Explain the need of *dividing* the compilation process into various phases and discuss each of the phases. [3]  
(b) Can *lexical analysis phase* be preferably separated from parsing? Give the reason. What issues are faced to design lexical analyzer and how these are resolved? Discuss with suitable examples. [2]
- Q3 (a) Explain why a system may have several compilers but normally has a single linker. [2]  
(b) Define parser. Can a C-compiler detect any error for the *following* statement at Lexical analysis phase? Justify your answer. `int 13x, *p;` [3]
- Q4 (a) List the *demerits* of top down parser with examples. [2]  
(b) Explain the importance of finding FIRST and FOLLOW sets for constructing LL(*k*) parser;  $k \geq 1$ . [3]
- Q5 Design LL(1) parser (table construction is essential) for the grammar (G) given below (without removing immediate left-recursion).  $E \rightarrow E+T, E \rightarrow T, T \rightarrow id$ . Is the grammar LL(1)? Draw conclusion about the grammar G from the designed LL(1) table. Discuss how to detect and recover errors by LL(1) parser. [5]
- Q6 (a) Discuss the *secondary tasks* of lexical analyzer Give the Lex specification for removing multi-line comment used in C-program. [3]  
(b) Explain briefly various *conflicts* that occur during *shift-reduce* parsing. [2]

:::::: 26/02/2020 :::::M