BIRLA INSTITUTE OF TECHNOLOGY, MESRA, RANCHI
(END SEMESTER EXAMINATION MO/SP20)**

CLASS: **MCA** **SEMESTER : III**
BRANCH: **MCA** **SESSION : MO//2022**

SUBJECT: CA513 COMPILER DESIGN

TIME: **03 Hours** **FULL MARKS: 50**

INSTRUCTIONS:
1. The question paper contains 5 questions each of 10 marks and total 50 marks.
2. Attempt all questions.
3. The missing data, if any, may be assumed suitably.
***4*. Tables/Data handbook/Graph paper etc., if applicable, will be supplied to the candidates**

Q.1(a) List out all phases of Compilation process and describe their functioning in one line for each of them. [2]
Q.1(b) Differentiate between Compiler and Interpreter. [3]
Q.1(c) Design a Lexical Analyzer to recognize for loop statement. [5]
Statement will have a single condition with two variables,
Variable names used for writing condition will be of single alphabet character.
Operators used in writing the condition will belong to set {<, =, >}

Q.2(a) Write condition for a Grammar to become LL(1) [2]
Q.2(b) Given a Graamar [3]
S→ qABC
A→ a | bbbD
B → a |ebsilon
C → b | ebsilon
D → c | ebsilon
Note: S is starting symbol of the Grammar
Capital letters are Non-terminals and
Small letters are terminals
Find FIRST and FOLLOW sets for each Non terminals of the grammar.
Q.2(c) Construct LL(1) parsing table for the Grammar mentioned in Question 2.b **[5]**

Q.3(a) When a Grammar is called Augmented and why it is needed. Explain in two lines only. [2]
Q.3(b) Define LR(1) Items and explain functioning of Closure function to calculate LR(1) set of items. [3]
Q.3(c) Given a Grammar [5]
S→ L = R
S → R
L → *R
L → id
R→ L
Note: S is starting symbol of the Grammar
Capital letters are Non-terminals and
=, *, id are terminals
Construct LR(1) set of items for this Grammar and
Also construct the Canonical LR Parser for it.

Q.4(a) Define Sythesized and Inherited attributes. [2]
Q.4(b) Write short note on Backpatching [3]
Q.4(c) Write Syntax directed translation scheme for generating three address code of an assignment statement. Right hand side of the statement will be a mathematical expression which may have +, -, * and % operators and it may have parenthesized subexpression. Use suitable grammar for producing such construct. [5]

Q.5(a) While optimizing code, do we need to take care of target machine? Give two line justification [2]
Q.5(b) How common subexpressions are identified and optimized? Explain with suitable example. [3]
Q.5(c) With suitable example, explain the Peephole optimization technique. [5]

:::::22/11/2022:::::E