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

CLASS: BTECH | **SEMESTER: VI**
BRANCH: CS | **SESSION: SP/2025**

SUBJECT: CS333 COMPILER DESIGN

TIME: 2 HOURS | **FULL MARKS: 25**

INSTRUCTIONS:
1. The total marks of the questions are 25.
2. Candidates attempt for all 25 marks.
3. Before attempting the question paper, be sure that you have got the correct question paper.
4. The missing data, if any, may be assumed suitably.
5. Tables/Data hand book/Graph paper etc. to be supplied to the candidates in the examination hall.

				CO	BL
Q1	(a)	Identify at least 4 most relevant cousins of a compiler from the following list. Describe the function of each cousin chosen by you in one line. matlab, network simulator, preprocessor, numpy, assembler, loader, sql optimizer, linker, operating system	[2]	CO-1	Understand
Q1	(b)	Give one example of an error in a C program that is a lexical error (make sure the error is NOT a syntax error or a semantic error). Examine the following statement, "*During parsing, both terminal and nonterminal symbols appear on the stack of a top down parser and also on the stack of a bottom up parser*". Write whether the statement is valid or not with brief justifications.	[3]	CO-2	Understand
Q2	(a)	Consider two tokens specified by the regular expressions below in the order given. **<u>Token specification</u>** a+b+ Token 1 a*b* Token 2 <u>Write answers for the questions.</u> (i)A string of length 2 that is recognized as Token 2 but not as Token 1. (ii) All strings of length 4 that are recognized as tokens by both the regexes.	[2]	CO-3	Analyse
Q2	(b)	Given the following 2 definitions, write regular expressions for unsigned integers, unsigned octal numbers (start with 0 followed by one or more digits from 0 to 7), and unsigned hexadecimal numbers (start with 0x or 0X followed by any hexadecimal digit). Do not add any more definitions. base8 [0-7] letter [a-fA-F]	[3]	CO3, CO5	Apply
Q3	(a)	Examine the operators and their properties given below, which may be different than that of C / C++. The terminals to be used are: { > >= == != && | | id ( ) }. Note that parentheses '(' and ')' have the highest precedence.	[2]	CO-3	Understand, Apply

Operator	Precedence	Associativity
> >=	10	Left
== !=	9	Right
&& | |	8	Left

(i) You have to write an unambiguous grammar that satisfies all the properties given.
(ii) Consider the input string: id == id == id && id > id | | id >= id .
Construct a parse tree for the above string according to the operator properties given.

PTO

Q3 (b) Consider the grammar G given below with start nonterminal E and 2 terminals {id *}. [3] CO-3 Analyse

$E \rightarrow E * id \mid id * E \mid id$

(i) Write the associativity of operator *. (ii) Consider the string w = " id $*_1$ id $*_2$ id $*_3$ id" where $*_1$,$*_2$ and $*_3$ are the operator * only, but have been numbered for convenience. Using G, find all distinct parse trees of w; you are required to retain the labels of * in the parse trees.

(iii) Can you find a parse tree for w that can not be generated using G?

Q4 (a) Consider grammar G, given below, with S as start symbol and 6 terminals {id num ( ) , ; }, which is a simplified form of a function call statement. [2] CO-3 Understand, Apply

CFG named G: $S \rightarrow id\ (A);$ $A \rightarrow A\ ,\ T \mid T$ $T \rightarrow num \mid id$

Write answers for the questions.

(i) Construct FIRST( ) and FOLLOW ( ) for all nonterminals.

(ii) Without changing G, show the LL(1) table entry only for the row corresponding to A.

(b) In case G of Q4(a) is not a LL(1) grammar, make minimal changes to G to make it LL(1). Show the new LL(1) table for all nonterminals, except S. [3] CO3 Analyse

Q5 (a) Consider a state t of some SLR(1) parser, that has four LR(0) items in it as given below; where A, B, C, D and E are nonterminals, b and d are terminals and the remaining are grammar symbols. [2] CO-3 Analyse

1. $A \rightarrow \alpha \bullet bB$ 2. $C \rightarrow \delta \bullet$ 3. $D \rightarrow \beta \bullet dC$ 4. $E \rightarrow \eta \bullet$

What relation(s) must be TRUE among the grammar symbols to ensure that state t has no conflicts (neither shift-reduce nor reduce-reduce)?

Q5 (b) Consider the following declaration grammar, with Dlist as the original start symbol and D' as the new start symbol, 5 terminals { id , int ; * }, 6 nonterminals, and rule 8 as the only null production. [3] CO-3 Apply, Analyse

1. $D' \rightarrow Dlist$ 2. $Dlist \rightarrow Dlist\ D$ 3. $Dlist \rightarrow D$ 4. $D \rightarrow T\ L;$

5. $L \rightarrow L\ ,\ P\ id$ 6. $L \rightarrow P\ id$ 7. $P \rightarrow *\ P$ 8. $P \rightarrow \varepsilon$

9. $T \rightarrow int$

The kernel item of start state 0 is {$D' \rightarrow \bullet Dlist$}. Consider state 2 which is goto(0, T).

(i) Write all the LR(0) items of state 2 and label each item as a kernel item, a closure item or a complete item.

(ii) Using the LR(0) items above, construct the *action* and *goto* table entries for state 2 only. Do not construct any other state of the SLR(1) automaton but choose convenient state numbers and indicate their relation to state 2 in order to populate the table entries.

:::::27/02/2025 E:::::