CBCS SCHEME

USN | | | | | | | | | | |

BCS613C/BIS613C

# Sixth Semester B.E./B.Tech. Degree Examination, Dec.2025/Jan.2026
## Compiler Design

Time: 3 hrs. Max. Marks: 100

***Note: 1. Answer any FIVE full questions, choosing ONE full question from each module.***
***2. M : Marks , L: Bloom's level , C: Course outcomes.***

| | | Module – 1 | M | L | C |
|---|---|---|---|---|---|
| Q.1 | a. | Explain different phases of compiler with example<br>position = initial + rate * 60 | 10 | L2 | CO1 |
| | b. | Describe an overview of different software productivity tools. | 6 | L2 | CO1 |
| | c. | How program translations used in compiler. Justify with types. | 4 | L3 | CO1 |
| | | **OR** | | | |
| Q.2 | a. | What is Complier? Explain a language processing system. | 10 | L1 | CO1 |
| | b. | Describe applications of compiler technology with respect to<br>i) Implementation of high level programming languages<br>ii) Optimization for computer architecture | 6 | L2 | CO1 |
| | c. | Write intermediate code for "do i = i + 1 ; while (a [i] < v) ;" | 4 | L3 | CO1 |
| | | **Module – 2** | | | |
| Q.3 | a. | Explain input buffering strategy used in Lexical analysis. | 8 | L1 | CO2 |
| | b. | Construct the transition diagram to recognize the tokens given below<br>i) Relational operation<br>ii) Unsigned number | 8 | L2 | CO2 |
| | c. | Eliminate left recursion of the following grammars.<br>i) L → L, id / id<br>ii) rfactor → rfactor * \| rprimary | 4 | L3 | CO2 |
| | | **OR** | | | |
| Q.4 | a. | Explain two ways to handle Reserved words and identifiers. | 8 | L1 | CO2 |
| | b. | Discuss different error recovery strategies. | 8 | L2 | CO2 |
| | c. | Write four operations on language with example. | 4 | L3 | CO2 |
| | | **Module – 3** | | | |
| Q.5 | a. | Construct predictive parsing table for following grammar based on LL(1) of top down approach. Also verify id * id is valid or not.<br>E → E + T \| T<br>T → T * F \| F<br>F → (E) \| id | 10 | L3 | CO3 |
| | b. | With predictive parsing table prove that following is not LL(1) grammar.<br>S → iEtSS' \| a<br>S' → eS \| E<br>E → b | 10 | L3 | CO3 |

| | | | | | |
|---|---|---|---|---|---|
| | | **OR** | | | |
| **Q.6** | **a.** | Explain the strategy of shift reduce parser and construct configurations of shift reduce parser on input $id_1 * id_2$ by applying suitable grammar, $E \rightarrow T, T \rightarrow T * F \mid F, F \rightarrow id$. | **10** | **L3** | **CO3** |
| | **b.** | Write First and Follow for the following grammars :<br>i) $S \rightarrow Aa\ Ab \mid Bb\ Ba$<br>$A \rightarrow \in$<br>$B \rightarrow \in$<br>ii) $S \rightarrow a\ Bh$<br>$B \rightarrow cC$<br>$C \rightarrow bC \mid \in$<br>$E \rightarrow g \mid \in$<br>$F \rightarrow f \mid \in$ | **10** | **L3** | **CO3** |
| | | **Module – 4** | | | |
| **Q.7** | **a.** | Show that the following grammar in SLR (1)<br>$S \rightarrow S\ A \mid A$<br>$A \rightarrow a$ | **10** | **L3** | **CO3** |
| | **b.** | What is inherited and synthesized attributes? Write SDD for simple desk calculator based on bottom up approach. | **10** | **L2** | **CO4** |
| | | **OR** | | | |
| **Q.8** | **a.** | Construct set of LR (0) items and SLR parsing table for<br>$E \rightarrow E + n \mid n$ | **10** | **L3** | **CO3** |
| | **b.** | Discuss S attributes and L attributes. Write annotated parse tree for 3 * 5 using grammar suitable for top down parsing. | **10** | **L2** | **CO4** |
| | | **Module – 5** | | | |
| **Q.9** | **a.** | What is a three address code? Explain the different ways of representing three address codes with examples. | **10** | **L1** | **CO5** |
| | **b.** | Build Directed Acyclic Graph (DAG) and steps for constructing the DAG along with SDD for the given expression<br>$a + a * (b - c) + (b - c) * d$ | **10** | **L2** | **CO5** |
| | | **OR** | | | |
| **Q.10** | **a.** | Explain the issues in the design of code generation. | **10** | **L1** | **CO5** |
| | **b.** | Discuss a simple target machine model along with different kinds of instructions and addressing modes. | **10** | **L2** | **CO5** |

* * * * *