

CMPN - TE Sem VI Reg C'scheme Summer 2025 14/5/2025

(3 Hours)

Total Marks: 80

- N.B:** (1) Question No. 1 is compulsory.
 (2) Attempt any three questions out of the remaining five questions.
 (3) Make suitable assumptions wherever necessary.

- Q.1.** A. With a neat diagram, explain the sequence of system programs involved from writing source code to executing a program. **5**
 B. Explain the process of loading and linking with a neat diagram. **5**
 C. Write the structure of a simple macro definition and a corresponding macro call with suitable example. **5**
 D. What are tokens, lexemes, and patterns in lexical analysis? Give examples.. **5**
- Q.2.** A. Consider the following Assembly Program: - **10**
 START 100
 READ N
 MOVER BREG, '=1'
 MOVEM BREG, TERM
 A: MULT BREG, TERM
 LTORG
 MOVER CREG, '=2'
 MOVEM BREG, '=5'
 LTORG
 N DS 1
 TERM DS 1
 END
 Generate Pass-1 and Pass-2 and show the content of Database table involved in it.
- B. What are the different types of intermediate code representations in compilers? Explain with examples. **10**
- Q.3.** A. Explain Macro calls within the Macros with appropriate example? **10**
 B. Design a Predictive Parser for the given grammar. Mention all the steps. **10**
 $E \rightarrow TQ$
 $T \rightarrow FR$
 $Q \rightarrow +TQ \mid -TQ \mid E$
 $R \rightarrow *FR \mid /FR \mid E$
 $F \rightarrow (E) \mid id$
- Q.4.** A. List and describe the data structures used in the design of a two-pass macro processor with suitable example. **10**
 B. Explain the operator precedence parser with a suitable example. **10**

- Q.5. A. Define Code Optimization. What are the different types of code optimization techniques used in compilers? Provide an example of each. 10
- B. Explain the role of a code generator in a compiler. What are the issues to be considered in the design of a code generator? 10
- Q.6. A. Discuss the concept of "Relocation" in loaders. Explain the various methods of handling relocation during the loading process with examples. 10
- B. Explain the phases of compiler with suitable example. 10
-