



# CBCS SCHEME

USN

18CS61

Sixth Semester B.E. Degree Examination, Jan./Feb. 2023

## System Software and Compilers

Time: 3 hrs.

Max. Marks: 100

Note: Answer any FIVE full questions, choosing ONE full question from each module.

### Module-1

- 1 a. Explain the SIC/XE machine architecture. (10 Marks)
- b. Suppose alpha is an array of 100 words. Write a sequence of instruction for SIC/XE to set all 100 elements of an array to 0. (05 Marks)
- c. What is system software? Compare system software with application software and give couple of example. (05 Marks)

OR

- 2 a. Explain the assembler directive and data structures used in assembler. (10 Marks)
- b. Write an algorithm of pass 1 of 2 – pass assembler. (05 Marks)
- c. Explain the bootstrap loader. (05 Marks)

### Module-2

- 3 a. Explain the structure of a compiler with an example. (10 Marks)
- b. List and example the applications of compiler technology. (06 Marks)
- c. Differentiate between type checking and bound checking. (04 Marks)

OR

- 4 a. Explain the role of lexical analyzer. (08 Marks)
- b. What is regular expression? Write the algebraic laws of regular expression. (06 Marks)
- c. Explain the concept of input buffering in the Lenticels analysis. (06 Marks)

### Module-3

- 5 a. Explain the different types of error recovery strategies in process. (06 Marks)
- b. Explain context free grammar and derivation. (06 Marks)
- c. Explain the top down parsing and process for the string  $id + id * id$ . Given the grammar :  
i)  $E \rightarrow E + E$   
ii)  $E \rightarrow E * E$   
iii)  $E \rightarrow (E)$   
iv)  $E \rightarrow id$  (08 Marks)

OR

- 6 a. Write the algorithm for recursive descent parser. For the following grammar write a recursive descent parser  
 $E \rightarrow T$   
 $T \rightarrow F$   
 $E \rightarrow E + T$   
 $T \rightarrow T * F$   
 $F \rightarrow (E)/id$ . (08 Marks)
- b. Is the following grammar ambiguous?  
(if – statement or if – then – else)  
 $S \rightarrow i c t S | i c t S e S | a$   
 $C \rightarrow b$ . (04 Marks)
- c. Explain bottom – up parsing, shift-reduce parsing and LL(1) grammar. (08 Marks)

**Module-4**

- 7 a. Explain the LEX specification with an example to count number vowels and consonants. (10 Marks)
- b. Explain the meta characters used in regular expression with an example. (05 Marks)
- c. Write a LEX program to count the number of scanf and printf statement and replacing them with readf and writef respectively. (05 Marks)

OR

- 8 a. Explain the YACC specification with an example. (10 Marks)
- b. Write a YACC program to accept strings of the form  $a^n b^n$  ( $n > 0$ ). (05 Marks)
- c. Discuss two types of conflict in YACC with an example. (05 Marks)

**Module-5**

- 9 a. What is syntax directed definition? Write the grammar and SDD for a simple desk calculation and show annotated Parse tree for the expression  $(3 + 4) * (5 + 6)$ . (08 Marks)
- b. What is an attribute? Explain the different types of attributes with example. (08 Marks)
- c. What is the difference between syntax tree and parse tree? (04 Marks)

OR

- 10 a. Explain the Intermediate Code Generation (ICG) and type of method used to convert ICG. (10 Marks)
- b. Explain the issues in the design of code generation. (10 Marks)

\*\*\*\*\*