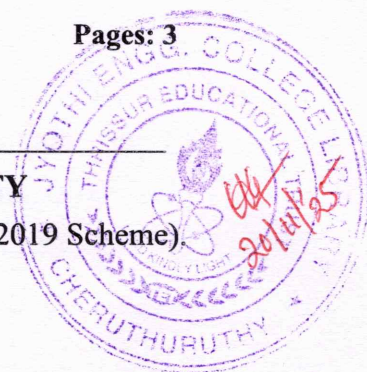


Reg No.: _____

Name: _____

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

B.Tech Degree S5 (R,S) (FT/WP)(S3 PT) Examination November 2025 (2019 Scheme)



Course Code: CST305
Course Name: SYSTEM SOFTWARE

Max. Marks: 100

Duration: 3 Hours

PART A*(Answer all questions; each question carries 3 marks)*

Marks

- | | | |
|----|---|---|
| 1 | Explain why system software must remain closely tied to the hardware architecture, whereas application software does not. Provide one example each. | 3 |
| 2 | State any three differences between the addressing mechanisms of the SIC and SIC/XE machines. | 3 |
| 3 | List the basic functions performed by a two-pass assembler. | 3 |
| 4 | Briefly explain the structure and purpose of the Header, Text, and End records of an object program. | 3 |
| 5 | What is program relocation? Briefly explain why relocation is required in assemblers. | 3 |
| 6 | Define a literal and a program block. How does each affect the way memory is organized during assembly? | 3 |
| 7 | What is relocation in loaders? Briefly explain how a relocation bit (R-bit) is used. | 3 |
| 8 | Explain automatic library search in loaders. When is it needed? | 3 |
| 9 | Describe keyword macro parameters. Give one example. | 3 |
| 10 | List any three debugging methods and state one-line description for each. | 3 |

PART B*(Answer one full question from each module, each question carries 14 marks)***Module -1**

- | | | |
|----|--|---|
| 11 | a) With the help of a neat block diagram, describe the architecture of the SIC machine. Explain how the limited register set and instruction format influence the design of system software. | 7 |
| | b) Compare all the addressing modes supported by the SIC/XE architecture. For each addressing mode, illustrate how the effective address is calculated using a small example. | 7 |

- 12 a) Explain the role of assembler directives in assembly language programming. Demonstrate how directives such as START, EQU, USE, LTORG, CSECT, and END influence the organization of memory and control flow in a program. 7
- b) A modern system uses several system software components such as assembler, loader, linker, macroprocessor, text editor, debugger, and device driver. Explain how these components interact during program development and execution, taking a small program's lifecycle as an example. 7

Module -2

- 13 a) Write and explain a short SIC/XE program to copy 10 bytes from SRC to DEST using indexed addressing. 7
- b) Explain the algorithm of Pass 1 of a two-pass assembler and describe how the key data structures (SYMTAB, OPTAB, LITAB) are updated during Pass 1. 7
- 14 a) Describe the steps involved in hand-assembling a SIC/XE instruction sequence. Illustrate how target address, opcode, and flags are computed. 7
- b) With a neat diagram, explain the overall structure of a two-pass assembler and show how input, intermediate file, and object program interact during assembly. 7

Module -3

- 15 a) Explain how control sections enable modular program linking. Illustrate with a small example showing two control sections and their external references. 7
- b) Describe the working of a one-pass assembler. What limitations of the one-pass assembler make the two-pass assembler preferable in most cases? 7
- 16 a) Explain how expressions are evaluated in assemblers. Discuss absolute vs. relative expressions with two small examples. 7
- b) With suitable diagrams, explain how program blocks are assigned addresses and how their object code is reordered in the final output. 7

Module -4

- 17 a) Describe the design of an absolute loader with a neat flow diagram. Explain why absolute loaders are rarely used today. 7
- b) Explain the two-pass algorithm of a linking loader. Highlight the role of ESTAB and external references in each pass. 7
- 18 a) Describe the differences between linking loader, linkage editor, and dynamic linking. Give suitable examples for each. 7

- b) Explain how program linking works in SIC/XE using modification records. Show how external symbols are resolved using reference numbers. 7

Module -5

- 19 a) Explain the one-pass macroprocessor algorithm. Describe how the datastructures NAMTAB (MNT), DEFTAB (MDT), and ARGTAB (ALA) are used during macro expansion. 7
- b) With examples, explain the machine-independent macroprocessor features: 7
- (i) Unique label generation
 - (ii) Concatenation of macro parameters
 - (iii) Conditional macro expansion
- 20 a) Draw and explain the general structure of a text editor. Describe how the editing buffer and viewing buffer interact. 7
- b) Explain the anatomy of a device driver. Distinguish between character device drivers and block device drivers with examples. 7
