

Seat No./Enrollment No.: _____

GUJARAT TECHNOLOGICAL UNIVERSITY
BACHELOR OF ENGINEERING – SEMESTER 6 – EXAMINATION – SUMMER 2026

Subject Code: 3160715

Date: 01/06/2026

Subject Name: System Software

Time: 10:30 AM TO 01:00 PM

Total Marks: 70

Instructions:

- 1. Attempt all questions.**
- 2. Make suitable assumptions wherever necessary.**
- 3. Figures to the right indicate full marks.**
- 4. Simple and non-programmable scientific calculators are allowed.**

- Q.1**
- (a) Explain semantic gap, specification gap and execution gap. **03**
 - (b) Explain the role of Symbol Table and Literal Table in the assembly process. **04**
 - (c) Define System Software. Explain the Life Cycle of a Source Program with a neat diagram. **07**
- Q.2**
- (a) Compare and contrast Top-Down and Bottom-Up Parsing techniques. **03**
 - (b) What is a Macro? Explain the macro expansion process with an example. **04**
 - (c) Explain various Assembler directives with suitable example. **07**
- OR
- Q.2**
- (c) Describe following data structures: OPTAB, SYMTAB, LITTAB and POOLTAB. **07**
- Q.3**
- (a) What is ambiguity in grammar? Eliminate left recursion from the following grammar:
 $A \rightarrow Aa \mid b$ **03**
 - (b) List the types of errors encountered during program execution. **04**
 - (c) Explain the flowchart of a Single-Pass Assembler. What are its advantages and limitations compared to a Two-Pass Assembler? **07**
- OR
- Q.3**
- (a) Define: Parse tree, Syntax tree, DAG **03**
 - (b) Describe the functions of a Linker. Explain the concept of Dynamic Linking with its benefits. **04**
 - (c) What is code optimization? Discuss any five techniques for code optimization used in compilers. **07**
- Q.4**
- (a) What is a self-relocating program? How does it work? **03**
 - (b) Explain basic structure of a LEX program. **04**
 - (c) Explain the phases of compilation, from source code to target code, with diagram. **07**
- OR
- Q.4**
- (a) Compare and contrast Static and Dynamic Linking. **03**
 - (b) Write a short note on the Java Virtual Machine (JVM). **04**

- (c) Given a grammar, 07
 $E \rightarrow TA,$
 $A \rightarrow +TA \mid \varepsilon$
 $T \rightarrow VB$
 $B \rightarrow *VB \mid \varepsilon$
 $V \rightarrow id \mid (E)$
Develop an LL (1) parser table and parse following string using the parsing table. id *
(id + id)

- Q.5 (a) Explain linking of overlay structured program. 03
(b) Explain use of AIF and AGO with example. 04
(c) What is interpreter? Explain pure & impure interpreters. 07

OR

- Q.5 (a) Explain Bootstrap loader. 03
(b) Difference between Procedure oriented language and Problem oriented language. 04
(c) What is program relocation? How relocation is performed by linker? Explain with example. 07

GUJARAT TECHNOLOGICAL UNIVERSITY**BE - SEMESTER-VI EXAMINATION – SUMMER 2025****Subject Code: 3160715****Date: 30-05-2025****Subject Name: System Software****Time: 10:30 AM TO 01:00 PM****Total Marks:70****Instructions:**

1. Attempt all questions.
2. Make suitable assumptions wherever necessary.
3. Figures to the right indicate full marks.
4. Simple and non-programmable scientific calculators are allowed.

		MARKS
Q.1	(a) Define following: 1)Specification Gap 2)Language Processor 3)Language Migrator	03
	(b) Differentiate Problem Oriented Languages and Procedure Oriented Languages.	04
	(c) Describe System Software Development with appropriate diagram.	07
Q.2	(a) Define Left Recursion. Remove Left Recursion from the following grammar: $A \rightarrow A x \mid A y \mid A B \mid c \mid d$ $B \rightarrow e$	03
	(b) Construct LL(1) Parsing table for the following grammar: $S \rightarrow aAC \mid Bb$ $A \rightarrow eD$ $B \rightarrow f \mid g$ $C \rightarrow h \mid i$ $D \rightarrow bE \mid \epsilon$ $E \rightarrow eD \mid dD$	04
	(c) Explain advanced assembler directives with suitable example.	07
	OR	
	(c) Consider following assembly language program: (i) Write equivalent machine code (ii) Write Intermediate code using Variant-I representation. START 200 READ A READ B MOVER BREG, A MULT BREG, B MOVEM BREG, C STOP A DS 1 B DS 1 C DS 1 END	07
Q.3	(a) Write a macro with parameters A, B, C and D. Calculate $A * B + C * D$ in AREG.	03
	(b) Explain nested macro call with example.	04

- (c) Explain expansion time loops in macro. **07**
- OR**
- Q.3** (a) Compare features of subroutine and macro. **03**
- (b) Explain positional parameter, keyword parameter and default value parameter. **04**
- (c) Explain expansion time statements AIF, AGO and ANOP. **07**
- Q.4** (a) Compare loader and linker. **03**
- (b) Explain Bootstrap loader. **04**
- (c) What is Overlay? Explain the execution of an overlay structured program. **07**
- OR**
- Q.4** (a) Explain Compile-and-Go loader. **03**
- (b) Consider an object module with three programs A, B and C, located at following addresses: **04**
- | Program | Address |
|---------|---------|
| A | 200-250 |
| B | 302-370 |
| C | 480-533 |
- The load address is 300 for the executable A, B and C to be generated. Find: i) Size of each Program ii) Load origin of each program
- iii) Relocation Factor for each Program
- (c) Explain self-relocating program. **07**
- Q.5** (a) Explain analysis phase of an assembler. **03**
- (b) Write quadruple representation for the following expression: **04**
- Result = -w * x + -y * z
- (c) Explain pure and impure interpreters. **07**
- OR**
- Q.5** (a) Compare one pass assembler and two pass assemblers. **03**
- (b) Describe static binding and dynamic binding. **04**
- (c) Describe JVM with appropriate diagram. **07**

GUJARAT TECHNOLOGICAL UNIVERSITY**BE - SEMESTER-VI (NEW) EXAMINATION – SUMMER 2024****Subject Code:3160715****Date:24-05-2024****Subject Name: System Software****Time: 10:30 AM TO 01:00 PM****Total Marks:70****Instructions:**

1. Attempt all questions.
2. Make suitable assumptions wherever necessary.
3. Figures to the right indicate full marks.
4. Simple and non-programmable scientific calculators are allowed.

		MARKS
Q.1	(a) Explain fundamental of language processing.	03
	(b) Define following terms:	04
	1) Language Migrator	
	2) Execution gap	
	3) Token	
	4) Handle	
	(c) Explain with examples - expansion time variables, expansion time Statements -AIF and AGO for macro programming.	07
Q.2	(a) Apply rule to remove left recursion from following grammar.	03
	$S \rightarrow A$	
	$A \rightarrow Ad \mid Ae \mid aB \mid aC$	
	$B \rightarrow bBC \mid f$	
	$C \rightarrow g$	
	(b) Construct LL(1) parsing table for following grammar.	04
	$S \rightarrow iCtSeS \mid iCtS \mid a$	
	$C \rightarrow b$	
	(c) Describe following data structures: OPTAB, SYMTAB, LITAB and POOLTAB	07
	OR	
	(c) Explain and compare two variants of the intermediate code generated from multi pass assembler.	07
Q.3	(a) Compare top-down and bottom-up parser.	03
	(b) Explain inherited and synthesized attributes in detail with example.	04
	(c) Consider following assembly language program: Create equivalent machine code	07

	START	101
	READ	N
	MOVER	BREG, ONE
	MOVEM	BREG, TERM
	MULT	BREG, TERM
	MOVER	CREG, TERM
	ADD	CREG, ONE
	MOVEM	CREG, TERM
	COMP	CREG, N
	BC	LE, AGAIN
	MOVEM	BREG, AGAIN
	PRINT	RESULT
	STOP	
N	DS	1
RESULT	DS	1
ONE	DC	'1'
TERM	DS	1
	END	

Instruction opcode: STOP – 00, ADD – 01, MULT – 03, MOVER – 04,

MOVEM – 05, COMP – 06, BC – 07, READ – 09, PRINT – 10, LE – 02

Assembler directives: START – 01, END – 02

Declaration statements: DC – 01, DS – 02

Register code: BREG – 02, CREG – 03

OR

- Q.3** (a) Compare and Contrast macro preprocessor and macro assembler. **03**
- (b) Explain use and field of following tables of a macro
KPD TAB, MDT, EVTAB, SSTAB **04**
- (c) Consider following assembly language program: Create equivalent machine code **07**

```

START      300
MOVER      AREG, '=5'
MOVEM      AREG,R1
AGAIN      MOVER      AREG,R1
           MOVER      CREG,R2
           ADD        CREG,='1'
           BC         ANY,TERM
           .....
           LTORG
           = '5'
           = '1'

TERM       SUB        AREG,='1'
           BC         LT,DOWN
LAST       STOP
           ORIGIN     LOOP+2
           MULT       CREG,R2
           ORIGIN     LAST+1
R1         DS         1
DOWN       EQU        LOOP
R2         DS         1
           END
           = '1'

```

- Q.4** (a) Explain DAG with example. **03**
- (b) Explain design of a linker by addressing issues of relocation and linking. **04**
- (c) Given a grammar, **07**

$E \rightarrow TA,$
 $A \rightarrow +TA \mid \varepsilon$
 $T \rightarrow VB$
 $B \rightarrow *VB \mid \varepsilon$
 $V \rightarrow id \mid (E)$

Develop an LL (1) parser table and parse following string using the parsing table. id * (id + id)

OR

- Q.4** (a) Discuss parameters for Activation Records **03**
 (b) Explain characteristics of self-relocating programs. **04**
 (c) Define Operator precedence grammar. Convert following production rules of grammar into suitable Operator precedence grammar. **07**
 $E \rightarrow EAE \mid id$
 $A \rightarrow - \mid *$
- Q.5** (a) Explain Left factoring with example. **03**
 (b) List out various Code Optimization techniques used in Compiler. Explain any two techniques with suitable example. **04**
 (c) Generate Quadruple, Triple, Indirect Triple for following expression: $ans = a + b * c / 2.0$ **07**
- OR**
- Q.5** (a) Explain the terms Binding and Binding Times. **03**
 (b) Explain pure and impure interpreters. **04**
 (c) What is Symbol table? Explain how one can organize Symbol table using Linear Data Structure? **07**
- *****

GUJARAT TECHNOLOGICAL UNIVERSITY**BE - SEMESTER-VI (NEW) EXAMINATION – SUMMER 2023****Subject Code:3160715****Date:14-07-2023****Subject Name:System Software****Time:10:30 AM TO 01:00 PM****Total Marks:70****Instructions:**

1. Attempt all questions.
2. Make suitable assumptions wherever necessary.
3. Figures to the right indicate full marks.
4. Simple and non-programmable scientific calculators are allowed.

			MARKS
Q.1	(a)	Compare user-centric view and system-centric view of system software.	03
	(b)	Enlist levels of System Software. Explain any two in detail.	04
	(c)	Define Language Processing. List various phases of Language Processor. Explain each phase in detail.	07
Q.2	(a)	Explain the causes of Large Semantic Gap.	03
	(b)	An assembly program contains the statement X EQU Y+25 Indicate how the EQU statement can be processed if a) Y is a back reference b) Y is a forward reference.	04
	(c)	Given the source program:	07
		START A L1 D L2 C B END DS MOVER ADD MOVEM EQU PRINT ORIGIN DC ORIGIN STOP DC END 100 3 AREG, B AREG, C AREG, D A+1 D A-1 '5' L2+1 '19' L1	
		a) Show the contents of the symbol table at the end of Pass I.	
		b) Explain the significance of EQU and ORIGIN statement in the program and explain how they are processed by the assembler.	
		c) Show the intermediate code generated for the program.	

OR

- | | |
|---|-----------|
| (c) Differentiate one pass and two pass assembler. Explain how forward references are handled in two pass assembler. | 07 |
|---|-----------|

- Q.3 (a)** Compare and contrast the properties of macros and subroutines with respect to following: **03**
1. Code space requirements
 2. Execution speed
- (b)** Explain use of AIF and AGO with example. **04**
- (c)** Define a macro taking starting_location and N as parameters to find summation of all N numbers stored at location starting from starting_location. The result is to be stored at starting_location. **07**
- OR**
- Q.3 (a)** Explain the use of expansion time loop. **03**
- (b)** Explain macro definition and call in detail. **04**
- (c)** Define a Macro taking A and B as parameters to compute $A=A*B+B*B+A*B$. **07**
- Q.4 (a)** Explain linking of overlay structured program. **03**
- (b)** Explain Compile-and-Go loaders with example. **04**
- (c)** Write and explain an algorithm for first pass of a linker. **07**
- OR**
- Q.4 (a)** Explain absolute loader with suitable example. **03**
- (b)** Draw the flow chart for the dynamic linking. **04**
- (c)** With algorithm and example, explain how relocation is performed by linker? **07**
- Q.5 (a)** Discuss dead code elimination method with suitable example. **03**
- (b)** Explain JVM in detail. **04**
- (c)** Define Simple Phrase and Handle. Using Handle and Simple Phrase trace the bottom up parsing algorithm. **07**
- Grammar is :
- $E \rightarrow T + E \mid T - E \mid T$
- $T \rightarrow T * V \mid T / V \mid V$
- $V \rightarrow a \mid b \mid c \mid d$
- String is : $a - b * c + d$
- OR**
- Q.5 (a)** Discuss the limitations of stack based memory allocation. **03**
- (b)** Explain classification of debuggers. **04**
- (c)** Construct optimized DFA for following Regular Expression. **07**
- $(1^*)^* 0(0|1)^* \#$

GUJARAT TECHNOLOGICAL UNIVERSITY**BE - SEMESTER-VI (NEW) EXAMINATION – SUMMER 2022****Subject Code:3160715****Date:10/06/2022****Subject Name:System Software****Time:10:30 AM TO 01:00 PM****Total Marks: 70****Instructions:**

1. Attempt all questions.
2. Make suitable assumptions wherever necessary.
3. Figures to the right indicate full marks.
4. Simple and non-programmable scientific calculators are allowed.

MARKS

- Q.1**
- | | | |
|-----|---|-----------|
| (a) | Define system software. Give difference between system software and application software. | 03 |
| (b) | Explain the user-centric view and system-centric view of system software. | 04 |
| (c) | Explain lexical, syntax and semantic analysis with example. | 07 |
- Q.2**
- | | | |
|-----|--|-----------|
| (a) | Explain the following.
1. ORIGIN 2. EQU 3. LTORG | 03 |
| (b) | Show the difference between positional parameter and keyword parameter in macro. | 04 |
| (c) | Compare variant I and variant II of intermediate code. | 07 |
- OR**
- | | | |
|-----|---------------------------|-----------|
| (c) | Given the source program: | 07 |
|-----|---------------------------|-----------|

	START	200
	MOVER	AREG, ='5'
	MOVEM	AREG, M
L1	MOVER	AREG, ='2'
	ORIGIN	L1+3
	LTORG	
NEXT	ADD	AREG, ='1'
	SUB	BREG, ='2'
	BC	LT, BACK
	LTORG	
BACK	EQU L1	
	ORIGIN	NEXT+5
	MULT	CREG, ='4'
	STOP	
X	DS	1
	END	

1. Show the content of symbol table generated at the end of pass I.
2. Show the intermediate code generated for the program.

Q.3	(a) Compare and contrast the properties of macros and subroutines with respect to following criterion. 1. Code space requirement 2. Execution speed 3. Processing requirement by assembler 4. Flexibility	03
	(b) What is program relocation? How it is performed?	04
	(c) List and explain all the tables used in macro preprocessor.	07
OR		
Q.3	(a) Demonstrate the use of AIF and AGO.	03
	(b) Explain in brief about self relocating program.	04
	(c) List and explain all the task involved in macro expansion.	07
Q.4	(a) Explain compile-and-go loaders in brief.	03
	(b) What is debugger? Explain different types of error in program.	04
	(c) What is overlay? Explain the linking of overlay structured program performed.	07
OR		
Q.4	(a) Differentiate between linker and loader.	03
	(b) Differentiate pure and impure interpreter.	04
	(c) Write and explain an algorithm for first pass of the Linker program.	07
Q.5	(a) Explain Ambiguous Grammar.	03
	(b) Eliminate left recursion from the following grammar. $S \rightarrow Aa / b$ $A \rightarrow Ac / Sd / \epsilon$	04
	(c) What is optimizing transformation? discuss various optimizing transformations.	07
OR		
Q.5	(a) Define the following. 1. Finite state automaton 2. Regular expression 3. Operator grammar	03
	(b) Explain in brief about causes of large semantic gap.	04
	(c) Explain recursive descent parsing algorithm.	07
