Seat No.: _____ Enrolment No._____

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:
		START 100
		A DS 3
		L1 MOVER AREG, B
		ADD AREG, C
		MOVEM AREG, D
		D EQU A+1
		L2 PRINT D
		ORIGIN A-1
		C DC '5'
		ORIGIN L2+1
		STOP
		B DC '19'
		END 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.
			07

		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: 1. Code space requirements 2. Execution speed	**03**
	(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. Grammar is : E → T+ E | T – E | T T → T *V | T / V | V V → a | b | c | d String is : a - b * c + d	**07**

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. $(1^*)^*0(0\|1)^*\#$	**07**

****