

GUJARAT TECHNOLOGICAL UNIVERSITY**B E Sem-VI Examination May 2011****Subject code: 160706****Subject Name: System Programming****Date: 24/05/2011****Time: 10.30 am – 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.

- Q.1** (a) Explain following. **04**
 (i) Execution Gap (iii) Phases of Compiler
 (ii) Interpreter (iv) Language Migrator
- (b) Explain memory allocation in block structured language. **03**
- (c) Given following expression, $x = -a * b + -a * b$ **07**
 (i) Write three address code for the expression.
 (ii) Optimize the three address code if it is possible to do so.
 (iii) Give triple implementation for the three address code of the expression.
- Q.2** (a) Given a grammar, **07**
 $E \rightarrow TA, \quad A \rightarrow +TA \mid \epsilon$
 $T \rightarrow VB \quad B \rightarrow *VB \mid \epsilon$
 $V \rightarrow id \mid (E)$
 Develop an LL(1) parser table and parse following string using the parsing table.
 $id * (id + id)$
- (b) (i) Convert given regular expression to DFA. The expression is $(a \mid b)^* abb\#$ **05**
 (ii) Write regular expressions of a given language. The language consists of all strings of a 's and b 's which ends with a and does not contain bb . **02**
- OR**
- (b) What is bottom up parser? Explain operator precedence parser. Let a grammar for a language is $E \rightarrow E+E \mid E * E \mid id$. Check validity of following string using stack based operator precedence parser. $id * id + id * id$ **07**
- Q.3** (a) Explain assembly scheme with suitable example **07**
- (b) Let us consider a two pass assembler and assume that each instruction is one word. Given an assembly program and code for Mnemonics. **07**

	START 101	Mnemonics	CODE
	READ A	STOP	00
	READ B	MULT	03
	MOVER BREG, A	MOVER	04
	MULT BREG, B	MOVEM	05
	MOVEM BREG, D	READ	09
	STOP	DS	02
A	DS 1	START	01
B	DS 1	END	02
D	DS 1	Ordinal number of BREG is 2	
	END		

- (i) Show content of symbol table at the end of pass-one of an assembler.
- (ii) Write intermediate code representation of the assembly program. Use variant-II of intermediate code representation.

OR

- Q.3** (a) Explain the role of Mnemonic Opcode Table, Symbol Table, Literal Table and POOL Table in assembling process of assembly language program. **07**
(b) Compare single pass assembler and two pass assembler. Explain two pass assembler in detail with suitable example. **07**
- Q.4** (a) (i) Write a macro that moves n numbers from the first operand to the second operand, where n is specified as third operand of the macro. **07**
(ii) Write a macro which takes B, C, and D as parameters and calculates $B * C + C * D$.
(b) Compare the features of subroutine and macros with respect to following. **07**
(i) Execution speed
(ii) Processing requirement by assembler.
(iii) Flexibility and generality

OR

- Q.4** (a) Explain advanced macro facilities with suitable example. **07**
(b) Explain design specification tasks for macro preprocessor with suitable example. **07**
- Q.5** (a) Describe in detail how relocation and linking is performed. **07**
(b) Explain following terms with suitable example. **07**
(i) Expansion time variable (iii) Semantic Expansion
(ii) Positional parameter (iv) Macro Preprocessor

OR

- Q.5** (a) (i) Write a brief note on MS-DOS linker. **03**
(ii) What is Overlay? Explain the execution of an overlay structured program. **04**
(b) Do as directed **07**
(i) A macro is a unit of specification for _____ through expansion. (*Fill in the blank*)
(ii) Macro definition is enclosed between a _____ statement and a _____ statement. (*Fill in the blank*)
(iii) A _____ is designed to hold the names of all macro defined in a program. (*Fill in the blank*)
(iv) An _____ counter is maintained to count the number of nested macro calls. (*Fill in the blank*)
(v) A lexical expansion is typically employed to replace occurrence of formal parameters by corresponding actual parameters. (*State True/False*)
(vi) Default specification of parameters is useful in situations where a parameter has the different value in most calls. (*State True/False*)
(vii) Macro definition table is maintained to hold value of sequencing symbols. (*State True/False*)
