Seat No.: _____ Enrolment No._____

GUJARAT TECHNOLOGICAL UNIVERSITY

B. E. - SEMESTER – VI • EXAMINATION – WINTER 2012

Subject code: 160706 **Date: 08/01/2013**

Subject Name: System Programming

Time: 02.30 pm - 05.00 pm **Total Marks: 70**

Instructions:

1. **Attempt any five questions.**
2. **Make suitable assumptions wherever necessary.**
3. **Figures to the right indicate full marks.**

Q.1 **(a)** (i) Explain lexical analysis of a language processor. **07**

(ii) Construct DFA for following regular expression:

(a* | b*)a*ab #

(b) (i) Consider following grammar **07**

S -> aSbS | bSaS | epsilon

Derive the string abab. Draw corresponding parse tree.

Are these rules ambiguous? Justify.

(ii) Write regular definitions for producing real numbers of programming language 'C'.

Q.2 **(a)** Write unambiguous production rules to produce arithmetic expression consisting of +, -, *, / , ^ (exponent), id. **07**

Use them for parsing id ^ id ^ id * id + id / id using shift - reduce parser (Naïve bottom up parsing). Also lists limitation(s) of the method.

(b) (i) Remove left recursion from following production rules: **07**

A -> AaB | x

B -> BCb | y

C -> Cc | epsilon

(ii) Explain left factoring by giving example.

OR

(b) By giving suitable example, illustrate working of operator precedence parser. **07**

Q.3 **(a)** Explain working of LL (1) parser. Parse following string: **07**

id + id - (id * id)

(b) Explain use of various data structures (tables) needed in PASS I of the assembler. Also give details of their fields. **07**

Explain various suitable data structures for the symbol table.

OR

Q.3 **(a)** Explain following : **07**

(i) Assembler directives (any three)

(ii) Loader

(b) Consider following assembly program. Show (i) Contents of Symbol Table (ii) intermediate codes using Variant I representation (iii) corresponding machine codes **07**

START 100

```
READ A
READ B
READ C
MOVER AREG, A
ADD AREG, B
ADD AREG, C
MULT AREG, C
MOVEM AREG, RESULT
PRINT RESULT
STOP
A        DS    1
B        DS    1
C        DS    1
RESULT   DS    1
END
```

Instruction opcodes:
READ – 09, MOVER – 04, MOVEM – 05, ADD – 01,
MULT – 03, PRINT – 10, STOP – 00
Assembler-directive codes: START – 01, END - 02
Register code: AREG – 01

Q.4 **(a)** (i) Define a macro taking A and B as parameters to compute A = A * B + B * B + A + B **07**
(ii) Explain positional parameters, keyword parameters and default value parameters for macros.

(b) Explain advanced macro programming facilities. Give examples. **07**

OR

Q.4 **(a)** (i) Define a macro taking starting_location and N as parameters to find summation of all N numbers stored at locations starting from starting_location. The result is to be stored at starting_location. **07**
(ii) Illustrate expansion of nested macro calls by giving example.

Q.4 **(b)** List and explain tasks involved in macro expansion. **07**

Q.5 **(a)** (i) Write Three address codes and triple representation for x = x * y * z + x * y + y * z **07**
(ii) Explain self relocating program and overlay structured program.

(b) (i) List and compare various techniques for dynamic memory allocation. **07**
(ii) Give example(s) of errors detected by first three phases of compiler.

OR

Q.5 **(a)** Explain various optimizing transformations of a compiler by giving suitable examples. **07**

(b) Explain design of a linker by addressing issues of relocation and linking. **07**

****