

# C.U.SHAH UNIVERSITY

## Summer Examination-2018

**Subject Name: Translator Design**

**Subject Code: 4TE07TDE1**

**Branch: B.Tech (CE)**

**Semester: 7**

**Date: 28/03/2018**

**Time: 10:30 To 01:30**

**Marks: 70**

Instructions:

- (1) Use of Programmable calculator & any other electronic instrument is prohibited.
- (2) Instructions written on main answer book are strictly to be obeyed.
- (3) Draw neat diagrams and figures (if necessary) at right places.
- (4) Assume suitable data if needed.

- Q-1 Attempt the following questions: (14)**
- a) What is cross compiler?
  - b) Define parse tree.
  - c) Compare top down parser with bottom up parser.
  - d) Enlist properties of three address code.
  - e) Explain the task of loader in brief.
  - f) Write a regular expression for the set of strings of 0's and 1's not containing 101 as a substring.
  - g) What is peephole optimization?
  - h) Enlist any three issues in the design of code generation.
  - i) Define annotated parse tree.
  - j) What is handle?
  - k) Define operator precedence grammar.
  - l) What is Lex and YACC?
  - m) What is DAG? Draw a DAG for  $a+(b-c)*d+(b-c)*d$
  - n) Define synthesized attribute.

**Attempt any four questions from Q-2 to Q-8**

- Q-2 Attempt all questions**
- (a) Explain various storage allocation strategies in brief. (05)
  - (b) Parse the string  $id_1+id_2*id_3\#$  using shift reduce parser. (04)
  - (c) Write syntax directed definition for constructing a syntax tree for the arithmetic expression including +, - and / operators. (05)
- Q-3 Attempt all questions**
- (a) Construct NFA using Thompson's notation for following regular expression and convert it into DFA.  $(a * | b^*)ba^*\#$  (06)
  - (b) Explain error recovery strategies in parser. (04)
  - (c) What is an ambiguous grammar? Explain with suitable example. (04)
- Q-4 Attempt all questions**
- (a) Construct DFA for following regular expression without constructing NFA and minimize it.  $a^+ b (c | d) a^* b \#$  (06)



(b) Define token, pattern and lexeme. (03)

(c) Find LR(1) items for following grammar. (05)

$S \rightarrow A a \mid a A c \mid B c \mid b B a$

$A \rightarrow d$

$B \rightarrow d$

**Q-5 Attempt all questions**

(a) Explain activation record in brief. (04)

(b) Construct a predictive parser for the grammar (07)

$E \rightarrow E+T \mid T$

$T \rightarrow T * F \mid F$

$F \rightarrow F' \mid (E) \mid 0 \mid 1$

(c) What is left factoring? Explain it with suitable example. (03)

**Q-6 Attempt all questions**

(a) Write three address codes for following expression and generate final code by clearly showing register descriptors and address descriptors.  $x = a * (b - c) - d / (e + -f)$  (07)

(b) Find LR(0) items for following grammar and construct SLR parsing table. (07)

$S \rightarrow A a A b$

$S \rightarrow B b B a$

$A \rightarrow \epsilon$

$B \rightarrow \epsilon$

**Q-7 Attempt all questions**

(a) List the tasks performed by each pass of a two pass assembler. Also explain given directives for an assembler: ASSUME, EQU, EXTERN, ORIGIN (07)

(b) Explain principle sources of Code optimization in detail. (07)

**Q-8 Attempt all questions**

(a) What is nested macro? Explain with suitable example. (07)

(b) List the major steps of relocation and linking algorithms. Explain in brief. (07)

