Enrollment No: _____   Exam Seat No:_____

C.U.SHAH UNIVERSITY
Summer Examination-2017

Subject Name : Translator Design

Subject Code : 4TE07TDE1   **Branch: B.Tech (CE)**

Semester : 7   **Date : 29/03/2017**   **Time : 02:30 To 05:30**   **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) Define compiler.
 b) What is a parse tree?
 c) The graph that shows basic blocks and their successor relationship is called

  i. DAG   ii. Flow Graph   iii. Control graph   iv. Hamiltonion graph

 d) To recover from an error, the operator precedence parser may

  i. insert symbols onto the stack and onto the input   iii. delete symbols from the input

  ii. delete symbols from the stack   iv. all of these

 e) In which way(s) a macro processor for assembly language can be implemented?

  i. Independent two-pass processor   iii. Expand macro calls and substitute arguments

  ii. Independent one-pass processor   iv. All of the above

 f) What do you mean by synthesized attribute?
 g) What is a pass in compiler design?
 h) List cousins of compiler.
 i) What is macro?
 j) List the main task done by linker and loader.
 k) Give the definition of cross compiler.
 l) Define handle.
 m) Translate a $*$ - (b+c) into postfix form.
 n) What do you mean by self relocation programs?

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

Q-2   **Attempt all questions**   **(14)**
 (a) Explain the phases of compiler design.
 (b) Construct an NFA using Thompson's construct for regular expression $(a \mid b)^+ a*b\#$ and convert it into an equivalent minimum state DFA.

Q-3   **Attempt all questions**
 (a) Explain the techniques of input buffering.   **(04)**

 (b) Prove that the given grammar is LL(1).   **(07)**

E → E + T | T
T → T * F | F
F → (E) | i

(c) Explain shift-reduce parsing with suitable example. **(03)**

Q-4 **Attempt all questions** **(14)**

(a) Check whether the given grammar is LR(1) or not?
S→CC
C→cC | d

(b) Give the syntax directed definition for the grammar of arithmetic expressions. Also draw the annotated parse tree for the input string 9+5*2n.

Q-5 **Attempt all questions**

(a) What is operator precedence grammar? Construct the operator precedence function for the following grammar: **(07)**
E → E + E | E * E | E ^ E | (E) | a

(b) Construct the DAG for the following basic block: **(03)**
D := B * C
E := A + B
B := B * C
A := E − D

(c) Explain various error recovery strategies in parser. **(04)**

Q-6 **Attempt all questions** **(14)**

(a) Explain the sources of code optimization in detail.

(b) Discuss the issues in the design of a code generator.

Q-7 **Attempt all questions**

(a) What is activation record? Explain in detail. **(06)**

(b) Explain the pass structure of assemblers. **(04)**

(c) Explain peephole optimization in detail. **(04)**

Q-8 **Attempt all questions** **(14)**

(a) Write quadruples, triples and indirect triples for the expression
−(a+b)*(c+d)-(a+b+c).

(b) Explain various techniques of parameter passing.