

**GANPAT UNIVERSITY****B. Tech SEMESTER-VII [CE-IT] EXAMINATION****NOV-DEC - 2012****CE 701 / IT 705: COMPILER DESIGN****Time: 3 Hours]****[Total Marks: 70****Instructions:**

1. Figures to the right indicate full marks
2. Each section should be written in a separate answer book
3. Be precise and to the point in your answer

**SECTION-I****Q.1**

- (A) What are phases of a compiler? Discuss the function of each phase with example in brief. (04)
- (B) Explain classes of grammar as per Chomsky Hierarchy. (04)
- (C) What are the qualities of good compiler? (02)
- (D) Construct DFA for following language over alphabet  $\{a, b\}$   $\{w \mid w \text{ starts contains equal no. of 'a' and 'b'}\}$  (02)

**Q.2**

- (A) Check using parse tree that given grammar is ambiguous or not? (03)  
 $S \rightarrow aS \mid aSB \mid X$   
 $X \rightarrow Xa \mid a$
- (B) State advantages of shift reduce parser and explain with block diagram how shift reduce parser works. (04)
- (C) Write English description for the languages generated by the following regular expressions: (02)  
 $(x \mid y)^* x(x \mid y \mid \epsilon)$
- (D) Every unambiguous grammar is LL(1). Comment on the truth/falsehood of the statement. (02)

**OR****Q.2**

- (A) Give the leftmost and rightmost derivation, parse tree for **aaabbbab** string (03)  
 $S \rightarrow aAS \mid a, A \rightarrow SbA \mid SS \mid ba$
- (B) Consider the following grammar, and show the handle of each right sentential form for the string  $(a, (a, (a,a)))$  (03)  
 $S \rightarrow (L) \mid a$   
 $L \rightarrow L,S \mid S$



- (C) Find First() and Follow() for the following grammar and check whether grammar is LL(1) or not (05)
- lexp  $\rightarrow$  atom | list  
 atom  $\rightarrow$  number | identifier  
 list  $\rightarrow$  (seq)  
 seq  $\rightarrow$  lexp seq'  
 seq'  $\rightarrow$  lexp seq' |  $\epsilon$   
 Check string **(a(b(2)))(c)** is accepted or not?

**Q.3 Attempt Any TWO**

(12)

- (A) Generate Recursive Descent Parser for the following grammar. What are the disadvantages of it.  
 $E \rightarrow TA$   
 $A \rightarrow +TA \mid \epsilon$   
 $T \rightarrow FB$   
 $B \rightarrow *FB$   
 $F \rightarrow (E) \mid id$
- (B) Consider the following grammar with terminals  
 $S \rightarrow [S X] \mid a$   
 $X \rightarrow \epsilon \mid +SY \mid Yb$   
 $Y \rightarrow \epsilon \mid -SXc$   
 Compute the First() and Follow() sets  
 Construct LL(1) parsing table for this grammar  
 Is this grammar LL(1)? why or why not?
- (C) Discuss R-R and S-R conflicts with examples for SLR and LR(1) parsers.

**SECTION-II**

**Q.4**

- (A) Generate SLR parsing table for the following grammar and show error recovery implementation for string **y \* y + yy** and show stack content and moves. (05)  
 $A \rightarrow A + A \mid A * A \mid y$
- (B) Differentiate Predictive Parser vs Shift Reduce Parser. (03)
- (C) How loop interchange improves the cache performance? (03)

**Q.5 Attempt Any TWO**

(12)

- (A) Consider the following grammar with terminals  
 $S \rightarrow a I J h$   
 $I \rightarrow I b S e \mid c$   
 $J \rightarrow K L k r \mid \epsilon$   
 $K \rightarrow d \mid \epsilon$   
 $L \rightarrow p \mid \epsilon$   
 Generate LL parsing table of the given grammar and check whether the grammar is LR(1) or not.?



- (B) Explain LR parsing algorithm with diagram. Simulate algorithm on given grammar and LR parsing table and check whether string **aaaaab** is accepted or not

$S \rightarrow AA$

$A \rightarrow aA$

$A \rightarrow b$

- (C) Construct LALR parsing table for the following grammar

$S \rightarrow Aa \mid bAc \mid BC \mid bBa$

$A \rightarrow d$

$B \rightarrow d$

**Q.6 Attempt Any TWO**

(12)

- (A) (i) Consider the following code fragment. Generate the 3AC for it.

Sum = 0;

For (I=1; I<=0; I++)

    Sun = sum + a[I] + b[I];

- (ii) Write down 3AC for following program fragment

if  $x < y$  then

    while  $z > d$  do

$a = a + b$

else

    do  $p = p + q$

    while  $e \leq f$

- (B) (i) Differentiate following loop optimization techniques with example loop fusion and loop fission with example.

- (ii) What is locality of reference? Optimize the following code fragment

Count = 0;

Result = 0;

While (count++ < 20)

{

    increment = s+count;

    result += increment;

}

- (C) (i) State and compare loop unrolling and loop jamming with example.

- (ii) Discuss various approaches to Symbol Table Organization and explain how scope information is represented in Symbol Table.

-----X-----X-----