# SPECIAL SUPPLE B.TECH/CSE/7TH SEM/CSEN 4101/2018

# COMPILER CONSTRUCTION
# (CSEN 4101)

**Time Allotted : 3 hrs** **Full Marks : 70**

***Figures out of the right margin indicate full marks.***

***Candidates are required to answer Group A and***
***any 5 (five) from Group B to E, taking at least one from each group.***

***Candidates are required to give answer in their own words as far as practicable.***

## Group – A
## (Multiple Choice Type Questions)

1. Choose the correct alternative for the following: **10 × 1 = 10**

(i) A given grammar is not LL(1) if the parsing table of a grammar contains
(a) duplicate entry of same production (b) any Є-entry
(c) more than one production rule (d) any blank field.

(ii) The peephole optimization
(a) is applied to a small part of the code
(b) can be used to optimize intermediate code
(c) can be applied to a portion of the code that is not contiguous
(d) all of these.

(iii) Which is used to keep track of currently active activations?
(a) Control stack (b) Activation
(c) Execution (d) Symbol.

(iv) In operator precedence parsing, precedence relations are defined
(a) for all pair of non-terminals (b) for all pair of terminals
(c) to delimit the handle (d) none of the mentioned.

(v) We have the grammar E→E + p | E × p | p. The handles in the right-sentential form of the reduction for a sentence n + n × n are
(a) p, p + p and p + p × p (b) p, E + p and E × p
(c) p, E + p and E + E × p (d) p, E + p and E + p × p.

(vi) The grammar P → Pq1|Pq2|r1|r2 is
(a) a CFG
(b) left recursive and has a common left factor
(c) left recursive
(d) a grammar with common left factor.

(vii) {S, A, B} is the non-terminal alphabet and {a, b} is the terminal alphabet of the CFG. S is the start symbol. The set of production rules are given below,
S→aB  S→bA  B→b  A→a B→bS  A→aS  B→aB  B A→bAA
Which string is accepted by the given grammar?
(a) aabbbb  (b) abbbba  (c) aabbab  (d) aaaabb.

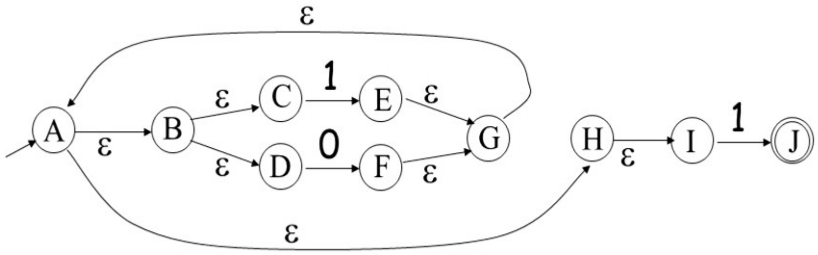
(viii) Optimization(s) connected with x : = x + 0 is/are
(a) Peephole and algebraic
(b) Reduction in strength and algebraic
(c) Peephole only
(d) Loop and peephole.

(ix) An S-attributed grammar can be evaluated ____________
(a) top down  (b) bottom up
(c) both (a) and (b)  (d) none.

(x) Which of the following is not an intermediate code form?
(a) Postfix notation  (b) Syntax trees
(c) Three address codes  (d) Quadruples.

## Group – B

2. (a) Convert following NFA to equivalent DFA.



(b) What are the front end and back end of a compiler?

(c) Draw a DFA that recognizes floating point numbers.

(d) What language does the following regular expression generate?
(0|1)*0(0|1)(0|1)

**6 + 2 + 2 + 2 = 12**

3. (a) Construct the NFA for the regular expression (p* |q*)* pqq (p|q)*.

(b) What are the different phases of compilation? How the following statement is translated via different phases of compilation?
**a = b * c + d;**

**4 + 8 = 12**

## Group – C

4. (a) List the various error recovery strategies for a parser.

(b) Define 'Handle Pruning' in bottom-up parsing.

(c) Eliminate left recursion from the following grammar.
S → Aa | b
A → Ac | Sd | f

(d) Simplify the following grammar.
S → aA | aBB, A → aAA | ε, B → bB | bbC, C → B

**3 + 2 + 3 + 4 = 12**

5. (a) Construct the LL(1) parsing table for the following grammar.
exp → term exp'
exp' → addop term exp' | λ
addop → + | -
term → factor term'
term' → mulop factor term' |λ
mulop → *
factor → ( exp ) | num

(b) Check whether the following grammar is SLR(1) or not. Explain in detail.
A' → A A → (A) | a

(c) What is a symbol table?

**6 + 4 + 2 = 12**

## Group – D

6. (a) Write three address code for the following expression.
P = Q* – (R–S) + Q* – (R–S)

(b) Distinguish between quadruples, triples and indirect triples for the expression in 6(a).

(c) Construct syntax tree and DAG for the expression provided in 6(a).

**2 + 6 + (2 + 2) = 12**

7. (a) Consider the following SDT.
S → xxW { print("1"); }
S → y { print("2"); }

W → Sz { print("3"); }

If an SR parser carries out the translations specified, immediately after reducing with rules of grammar, what is the result carrying out the above translations on an input string "**$x^4yz^2$**"? Draw the parse tree for it.

(b) Define Synthesized and inherited attributes with suitable examples.

(c) Differentiate between S-attributed definition and L-attributed definition.

**(4 + 2) + 4 + 2 = 12**

## Group – E

8. (a) Translate the following code into machine code and show the register and address descriptors while the instructions are generated. Assume that two registers are available.

```
t := a – b
u := a – c
v := t + u
d := v + u
```

(b) Write a short note on
(i) Peephole optimization (ii) Basic blocks and DAG.

**6 + (3 + 3) = 12**

9. (a) Explain the following techniques of code optimization with suitable examples:
(i) Copy propagation (ii) Dead code elimination
(iii) Common sub-expression elimination

(b) Identify basic blocks in the following code segment

```
main( )
{
    int  i = 0, n = 10;
    int  a[10];
    while( i <=(n-1))
    {
       a[ i ] = i * i ;
        i = i + 1;
    }
    return ;
}
```

Draw flow graph for the above basic blocks.
Construct DAG for the above basic blocks.

**(3 × 2) + (2 + 2 + 2) = 12**