

Reg No.: _____

Name: _____

APJ ABDUL KALAM TECHNOLOGICAL UNIVERSITY

Sixth semester B.Tech examinations (S), September 2020

Course Code: CS304**Course Name: COMPILER DESIGN**

Max. Marks: 100

Duration: 3 Hours

PART A*Answer all questions, each carries 3 marks.*

Marks

- 1 State the role of lexical analyzer. Identify the lexemes and their corresponding tokens in the following statement: `printf ("Simple Interest=%f\n", si);` (3)
- 2 Explain any three tools that help a programmer in building a compiler efficiently. (3)
- 3 Eliminate the ambiguity from the given grammar (3)
- $E \rightarrow E * E \mid E - E \mid E ^ E \mid E / E \mid E + E \mid (E) \mid id.$**
- The associativity of the operators is as given below. The operators are listed in the decreasing order of precedence.
- (i) ()
 - (ii) / and + are right associative
 - (iii) ^ is left associative.
 - (iv) * and - are left associative
- 4 For what type of grammar, recursive descent parser cannot be constructed? (3)
- Show the steps involved in recursive descent parsing with backtracking for the string *cad* with the given grammar: $S \rightarrow cAd$ $A \rightarrow ab \mid a$

PART B*Answer any two full questions, each carries 9 marks.*

- 5 a) Trace the output after each phase of the compiler for the assignment statement: (6)
- $a = b + c * 10$** , if variables given are of float type.
- b) Show that the following grammar is ambiguous. (3)
- $bexpr \rightarrow bexpr \text{ OR } bterm \mid bterm$
- $bterm \rightarrow bterm \text{ AND } bfactor \mid bfactor$
- $bfactor \rightarrow \text{NOT } bfactor \mid (bexpr) \mid \text{TRUE} \mid \text{FALSE}$

- 6 a) Left factor the following grammar and then obtain LL(1) parsing table (6)
 $E \rightarrow T+E \mid T$
 $T \rightarrow \text{float} \mid \text{float} * T \mid (E)$
- b) What is the relevance of input buffering in lexical analysis? (3)
- 7 a) Write Non-recursive predictive parsing algorithm. (5)
- b) Write regular expressions for the following languages: (4)
- i) All strings over the English alphabet that contain the five vowels in order.
- ii) All strings of a's and b's that do not contain the subsequence *abb*.

PART C

Answer all questions, each carries 3 marks.

- 8 What is handle pruning? Indicate the handles in the reduction of the right sentential form $S S+ a *$ to the start symbol using the grammar below: (3)
 $S \rightarrow S S+ \mid S S * \mid a$
- 9 What are viable prefixes? For the given grammar $S \rightarrow 0 S 1 \mid 0 1$ write all the viable prefixes for the string 00001111 (3)
- 10 Give the S-attributed SDD of a simple desk calculator and show annotated parse tree for the expression $(3+4)*(5+6)$. (3)
- 11 Write a translation scheme for performing type checking of statements. (3)

PART D

Answer any two full questions, each carries 9 marks.

- 12 a) Construct canonical collection of LR(1) items for the following grammar: (5)
 $S \rightarrow AA$
 $A \rightarrow Aa \mid b$
- b) Differentiate between S-attributed and L-attributed definitions with suitable examples. (4)
- 13 a) Write the SDD for a simple type declaration and draw the annotated parse tree for the declaration **float a, b, c**. (5)
- b) Construct SLR parsing table for the grammar $A \rightarrow a \mid (A)$. (4)
- 14 a) Using operator precedence relations, parse the string $\text{id} + (\text{id} * \text{id})$. (5)
- b) Construct DAG for the expression $(a/10 + (b - 10))*(a/10 + (b - 10))$. Also write the sequence of instructions used for the DAG construction. (4)

PART E

Answer any four full questions, each carries 10 marks.

- 15 a) Using necessary figure, illustrate how the caller and callee cooperate in (6)

managing various tasks in stack allocation strategy when a procedure is activated.

- b) Explain copy propagation with an example. (4)
- 16 a) Write SDD to produce three-address code for Boolean expressions and obtain the three-address code for the statement given below: (6)
- ```
while a < b do
 if c < d then
 x = y + z
 else
 x = y - z
```
- b) Explain common sub expression elimination with an example. (4)
- 17 a) Identify any four issues in the design of a Code Generator. (6)
- b) Write the three address code sequence for the statement  $x=y*z + y*-z$ . Also give its triple representation. (4)
- 18 Write the code generation algorithm. Using this algorithm generate code sequence for the expression  $x = (a - b) + (a + c)$ . (10)
- 19 a) With suitable example of a basic block, explain the code-improving transformations of a basic block. (6)
- b) Describe the various fields in an activation record. (4)
- 20 a) Explain the 3 representations of three-address code statements. (6)
- b) What is static allocation strategy? What are its limitations? (4)

\*\*\*\*