U.S.N | | | | | | | | | |

P.E.S. College of Engineering, Mandya - 571 401
(An Autonomous Institution affiliated to VTU, Belagavi)
Sixth Semester, B.E. - Computer Science and Engineering
Semester End Examination; July / Aug. - 2022
Compiler Design

Time: 3 hrs   *Max. Marks: 100*

Course Outcomes

The Students will be able to:
CO1: Design simple lexical analyzer.
CO2: Construct simple top down parser for a given context free grammar.
CO3: Construct simple bottom up parser for a given context free grammar.
CO4: Apply different syntax directed translation schemes.
CO5: Generate intermediate and machine dependent code.

<u>**Note:**</u> *I) PART - A is compulsory. **Two** marks for each question.*
    *II) PART - B: Answer any <u>Two</u> sub questions (from a, b, c) for a Maximum of **18 marks** from each unit.*

Q. No.	Questions	Marks	BLs	COs	POs
	I : PART - A	**10**			
I a.	Define pattern and lexeme.	2	L2	CO1	PO1
b.	What is ambiguous grammar?	2	L1	CO2	PO1
c.	What is Handle?	2	L1	CO3	PO1
d.	Define syntax directed definition.	2	L1	CO4	PO1
e.	What is a flow graph?	2	L1	CO5	PO1
	II : PART - B	**90**			
	UNIT - I	**18**			
1 a.	Explain the concept of input buffering in the lexical analysis.	9	L1,2	CO1	PO1,2
b.	Give the regular expression and finite automata for the specification and recognition of identifier, constants and logical operators.	9	L5	CO1	PO1,2
c.	Briefly explain different phases of compiler taking the example statement; position = initial + rate * 60.	9	L1,2	CO1	PO1,2
	UNIT - II	**18**			
2 a.	Define left recursion, left factoring. Write an algorithm used for eliminating left recursion.	9	L1	CO2	PO1,2
b.	Write an algorithm for recursive decent parsing with an illustrative example.	9	L3	CO2	PO1,2
c.	Give algorithm for first and follow set construction and also compute first and follow for the give grammar. $E \rightarrow TE^1$; $E^1 \rightarrow +TE \mid E$; $T \rightarrow FT^1$; $T^1 \rightarrow *FT^1 \mid E$, $F \rightarrow (E) \mid id$	9	L4	CO3	PO1

UNIT - III	**18**	

3 a. Write an algorithm for shift reduce parser with example.	9	L3	CO3 PO1,2
b. consider the following grammar: $E \rightarrow E + T \mid T; \quad T \rightarrow T* F|F; \; F \rightarrow (E) \mid id$ Compute canonical collection of sets of LR(0) items.	9	L3	CO3 PO1,2
c. Write an algorithm to construct SLR parsing table and construct SLR parsing tale for the grammar: $E \rightarrow E + T \mid T; \; T \rightarrow T*F \mid F; \; F \rightarrow (E) \mid id$	9	L3	CO3 PO1,2

UNIT - IV	**18**	

4 a. Write the SDD for simple desk calculator and give annotated parse tree for 3*5+4n.	9	L3	CO4 PO1,2
b. Explain in detail different storage allocation strategies.	9	L2	CO4 PO1,2
c. Explain the following with an example: i) Inherited attribute ii) Synthesized attribute	9	L2	CO4 PO1,2

UNIT - V	**18**	

5 a. What are the basic blocks? Explain in detail DAG representation of basic blocks.	9	L1,2	CO5 PO1
b. Briefly explain the different types of intermediate codes for the expression a: $= b*-c+b*-c$.	9	L2	CO5 PO1,2
c. Explain various issues in the design of code generation.	9	L2	CO5 PO1

∗ ∗ ∗ ∗