Name: Enrolment No:	UPES UNIVERSITY WITH A PURPOSE

UNIVERSITY OF PETROLEUM AND ENERGY STUDIES
End Semester Examination, May 2019

Program: B.Tech. – CS - All IBM specialization + CL **Semester: VI**
Course : Compiler Design **Time 03 hrs.**
Course Code: CSEG 326 **Max. Marks: 100**

Instructions: Attempt all questions, however internal choice is mentioned.

SECTION A

S. No.		Marks	CO
Q 1	Define lex. Discuss the three parts of a lex program?	4	CO1
Q 2	Explain different ways of representing three-address code.	4	CO3
Q 3	Describe the concepts of sentinel? Define handle pruning.	4	CO2
Q 4	Comprehend the comparison between synthesized attributes and inherited attributes.	4	CO3
Q 5	Describe the symbol table storage allocation information.	4	CO5
	SECTION B		
Q 6	Explain in detail about the compiler construction tools.	10	CO1
Q 7	What is handle pruning? Explain with the help of the grammar S → SS+ / SS* / a and input string aaa*a++. Give a bottom up parser of the given input string.	10	CO2
Q 8	Write quadruple representation for the following:- a + a * (b – c) + (b-c) * d	10	CO4
Q 9	Explain the concept of Syntax Directed Definition (SDD). Consider the CFG given below:- S → EN E → E + T | E – T | T T → T * F | T / F | F F → (E) | digit N → ; a) Obtain the SDD for the above grammar. b) Construct the parse tree, syntax tree and annotated parse tree for the input string 5*6 +7; Or, Identify the following grammar is an LR(1) grammar and construct LALR parsing table.	10	CO3/ CO4

	S → Aa | bAc | dC | bda A→ d. Parse the input string **bdc**. Using table generated by you.		
	SECTION-C		
Q 10	Design the operator precedence parser and operator precedence graph for the grammar given below. E → E + E | E – E | E * E | E / E | E ^ E | (E) | -E | id	**20**	**CO3**
Q 11	Perform different code optimizations for the following code by first constructing Basic Blocks and flow graph and identify loop invariant statement. (1) PROD := 0 (2) I: = 1 (3) T_1:= 4*I (4) T_2 := **addr**(A) - 4 (5) T_3 := $T_2[T_1]$ (6) T_4:= **addr**(B) - 4 (7) T_5:= $T_4[T_1]$ (8) T_6:= T_3*T_5 (9) PROD:= PROD+T_6 (10) I:= I+1 (11) If I ≤ 20 goto (3) Or, Define the following terms: (a) DAG (b) Leading (c) Activation Record (d) Peep hole optimization	**20**	**CO5**

Name: Enrolment No:	UPES UNIVERSITY WITH A PURPOSE

UNIVERSITY OF PETROLEUM AND ENERGY STUDIES
End Semester Examination, May 2019

Program: B.Tech. – CS - All IBM specialization + CL **Semester: VI**
Course : Compiler Design **Time 03 hrs.**
Course Code: CSEG 326 **Max. Marks: 100**

Instructions: Attempt all questions, however internal choice is mentioned.

SECTION A

S. No.		Marks	CO
Q 1	Define lex. Discuss the three parts of a lex program?	**4**	**CO1**
Q 2	Discuss the concepts of sentinel? Define handle pruning.	**4**	**CO2**
Q 3	Define Syntax directed definition.	**4**	**CO3**
Q 4	Explain different ways of representing three-address code.	**4**	**CO4**
Q 5	Explain the fields in an Activation record.	**4**	**CO5**

SECTION B

Q 6	Write three-address code for the following program segment.
	sum = 0;
	for (i=0; i<=10; i++)
	sum = sum + a[i];
		10	**CO4**

Q 7	Eliminate left recursion and left factoring (if present) from the following grammar:- S → aB / aC / Sd / Se B → bBc / f C → g	**10**	**CO2**
Q 8	What is handle pruning? Explain with the help of the grammar S → SS+ / SS* / a and input string aaa*a++. Give a bottom up parser of the given input string.	**10**	**CO2**
Q 9	List out the properties of optimizing compilers? Or, Give the SLR parsing table for the grammar. S → L = R / R L → * R / id R → L	**10**	**CO1/ CO2**

SECTION-C			
Q 10	Analyze the following grammar is an LR(1) grammar and construct LALR parsing table. S → Aa | bAc | dC | bda A→ d. Parse the input string **bdc**. Using table generated by you.	**20**	**CO2**

Q 11	Perform different code optimizations for the following code by first constructing Basic Blocks, flow graph, and identify loop invariant statement.
	
	X=1;
	
	I=1;
	
	Y=1;
	
	While (I <=n)
	
	{ X = X+ A[i];
	
	Y = 2;
	
	I = I +1;
	
	}
	
	Or,
	
	Define the following terms:
	
	(e) DAG
	
	(f) Trailing
	
	(g) Symbol Table Organization
	
	(h) Peep hole optimization
		20	**CO5**