Name:

Enrolment No:

UNIVERSITY OF PETROLEUM AND ENERGY STUDIES
End Semester Examination, May 2018

Course: Compiler Design **Semester: IV**
Program: B.Tech.-CS (All IBM specialization)
Time: 03 hrs. **Max. Marks: 100**

Instructions: Attempt all the questions.

SECTION A

S. No.		Marks	CO
Q 1	How a parser generator can be used to facilitate the construction of the front end compiler.	**4**	**CO1**
Q 2	Define the term reduction, handle and right sentential form. Explain with a suitable example.	**4**	**CO2**
Q3	Find the reduced grammar equivalent to the following CFG:- S → AC / SB A → BaSC / a B → aSB / bbC C → Bc /ad	**4**	**CO2**
Q4	Discuss the peephole optimization?	**4**	**CO4**
Q5	What is a directed acyclic graph? Discuss the steps for constructing DAG.	**4**	**CO5**

SECTION B

Q 6	Write Syntax Directed translation rules such that along with the parsing, the infix expression will be converted into postfix form for the following grammar. E → E + T E → T T → T*F T → F F → (E) F → id Illustrate the rules with a suitable example.	**10**	**CO3**

Q7	Convert the following pseudo code into 3-Address code.
	
	while (A < C and B >D) do
	{if A=1 then
	C:= C+1
	else
	While (A< =D) do
	{A:=A+3
	}
	}
		10	**CO5**

Q8	Construct a predictive parsing table for the following grammar. S → (L) / ϵ A → , SA / ϵ L → SA	**10**	**CO3**
Q9	Write quadruples, triples and indirect triples for the expression:- -((A/B) + B)* (C+(D*E)) - (A+B+C) **Or,** Create a cross compiler for EQN using following compilers (i) C compiler, written in PDP-11, producing code in PDP-11 (ii) An EQN language compiler producing code for text formatter, TROFF and written in C. Show your steps using T-diagram.	**10**	**CO1/CO5**
SECTION-C			
Q 10	Construct SLR parsing table for the following grammar and identify the problem which may encounter while parsing a string. Resolve the problem encountered by constructing the CLR parsing table. Parse ***id=id*id + id*id*** with LALR parsing table constructed for the same grammar prescribed below. G → E = E | id E → E + T | T T → T * id | id	**20**	**CO2/CO3**

Q11	Construct the basic blocks, draw the flow graph and identify the loop invariant statements for the following pseudo code.
	x =1
	i= 1
	y =1
	while(i <= n){
	x= x + A[i]
	y=2
	i=i+1
	}
	Or,
	Discuss the following terms-:
	(a) Activation Record
	(b) Handle Pruning
	(c) Leading
	(d) Symbol Table Organization
		20	**CO5**

Name: Enrollment No:	UPES

UNIVERSITY OF PETROLEUM AND ENERGY STUDIES
End Semester Examination, May 2018

Course: Compiler Design **Semester: VI**
Program: B.Tech.- CSE (All IBM specialization)
Time: 03 hrs. **Max. Marks: 100**

Instructions: Attempt all the questions.

SECTION A
(All questions are compulsory)

S.No.		Marks	CO
Q 1	Discriminate between the front end and back end of a compiler? What are the advantages of breaking up the compiler functionality into these two distinct stages	**4**	**CO1**
Q 2	Describe the role of symbol table in compiler	**4**	**CO4**
Q 3	State the problems with Top-Down Parsing.	**4**	**CO2**
Q 4	State the difference between synthesized attributes and inherited attributes.	**4**	**CO3**
Q 5	Translate the following expression into triple representation: x[i] = interest(p, n, r) + y[i] +p	**4**	**CO5**

SECTION B
(All questions are compulsory)

Q 6	Explain the role of syntax directed translation scheme in detail.	**10**	**CO3**
Q 7	List various operations that can be implemented in a symbol table.	**10**	**CO4**

Q 8	For the following C code, generate the intermediate code (Three-address code only).
	while (a > b && a <= 2*b-5)
	{
	a = a + b ;
	}
		10	**CO5**

Q 9	Create a cross compiler for EQN using following compilers (i) C compiler, written in PDP-11, producing code in PDP-11 (ii) An EQN language compiler producing code for text formatter, TROFF and written in C. Show your steps using T-diagram. **Or,** Consider the following grammar: - A → AcB | cC | C B → bB | id C → CaB | BbB | B Construct the first and follow sets for the grammar. Also design a LL(1) parsing table for the grammar.	**10**	**CO1/CO2**

SECTION C (All questions are compulsory)			
Q 10	Construct LALR(1) for the following grammar. S →B B → begin DA end D → Dd; / ϵ A → A , E / ϵ E → B / S Check the validity of the string " begin d ; end".	**20**	**CO2**
Q 11	Perform different code optimizations for the following code by first constructing Basic Blocks and flow graph (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) Handle Pruning c) Trailing d) L-Attributed SDD	**20**	**CO5/CO1/CO2/CO3/CO4**