| Name:<br>Enrollment No: | UPES |
|---|---|

**UNIVERSITY OF PETROLEUM AND ENERGY STUDIES**
**Online End Semester Examination, December 2020**

**Course: Compiler Design** **Semester: V**
**Program: B.Tech- CSE (All Branches)** **Time: 03 Hrs**
**Course Code: CSEG 3015** **Max. Marks: 100**

**Instructions: Attempt all the questions.**

**SECTION A**

1. **Each Question carries 5 Marks.**
2. **Instruction: Write short notes / Select the correct answer(s).**

| S.No. | | CO |
|---|---|---|
| Q 1 | List down the fields in the activation record. | CO4 |
| Q 2 | Describe the role of symbol table in compiler | CO4 |
| Q 3 | State the problems with Top-Down Parsing. | CO2 |
| Q 4 | State the difference between synthesized attributes and inherited attributes. | CO3 |
| Q 5 | Which of the following is the most suitable alphabet for the language L = {SHAKTI, SHIVA, RAM}<br>(a) {A, H, I, K, M, R, S, T, V, N}<br>(b) {A, H, I, K, M, R, S, T, V}<br>(c) {a, h, i, k, m, r, s, t, v}<br>(d) {a, h, k, k, m, R, S, t, v} | CO1 |
| Q 6 | DAG representation of the basic block allows us to perform the following code improving transformations:<br>(a) Eliminating local common subexpressions.<br>(b) Eliminating dead code.<br>(c) Restricting reordering of statements.<br>(d) Applying algebraic laws to reorder operands of three-address instructions. | CO5 |

**SECTION B**

1. **Each question carries 10 marks.**
2. **Instruction: Write short/brief notes.**

| | | |
|---|---|---|
| Q 7 | Translate the following expression into quadruple and triple representations:<br>x[ i ] = interest(p, n, r) + y[ i ]. | CO3 |
| Q 8 | Create a cross compiler for EQN using following compilers (i) C compiler, written in | CO1 |

| | | |
|---|---|---|
| | 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. | |
| Q 9 | Consider the following possible sets of LR(1) items in the states of a shift/reduce parser:<br>State 0:<br>[ A → a •, b ]<br>[ B → a • b, b ]<br>[ C → b a •, a ]<br>State 1:<br>[ A → b •, a ]<br>[ B → b • b, b ]<br>[ C → a b •, a ]<br>a) For the items in State 0, list any conflicts that exist and describe for what lookaheads they occur.<br>b) For the items in State 1, list any conflicts that exist and describe for what lookaheads they occur. | **CO2** |
| Q 10 | Summarize the different error recovery techniques in syntax phase. | **CO4** |
| Q11 | Consider the following grammar: -<br>A → AcB \| cC \| C<br>B → bB \| id<br>C → CaB \| BbB \| B<br>Construct the first and follow sets for the grammar. Also design a LL(1) parsing table for the grammar.<br>OR<br>Construct the operator precedence parsing table by computing leading and trailing for every non-terminal in the following grammar:<br>. S → N<br>N → V = E #<br>N → E<br>E → V<br>V → id<br>V → * E | **CO2** |
| **SECTION C**<br>**1. Attempt any of the following. Each question carries 20 marks.**<br>**2. Instruction: Write long answer.** | | |
| Q 12 | Explain the different machine independent code optimization technique and apply those on the given code segment and draw the optimized flow graph. | **CO5** |

```
 (1)     i = m-1                  (16)    t7 = 4*i
 (2)     j = n                    (17)    t8 = 4*j
 (3)     t1 = 4*n                 (18)    t9 = a[t8]
 (4)     v = a[t1]                (19)    a[t7] = t9
 (5)     i = i+1                  (20)    t10 = 4*j
 (6)     t2 = 4*i                 (21)    a[t10] = x
 (7)     t3 = a[t2]               (22)    goto (5)
 (8)     if t3<v goto (5)         (23)    t11 = 4*i
 (9)     j = j-1                  (24)    x  = a[t11]
(10)     t4 = 4*j                 (25)    t12 = 4*i
(11)     t5 = a[t4]               (26)    t13 = 4*n
(12)     if t5>v goto (9)         (27)    t14 = a[t13]
(13)     if i>=j goto (23)        (28)    a[t12] = t14
(14)     t6 = 4*i                 (29)    t15 = 4*n
(15)     x = a[t6]                (30)    a[t15] = x
```

OR

**(A)** Generate code for the following three-address statements assuming all variables are stored in memory locations.

1. x = 1
2. x = a
3. x = a + 1
4. x = a + b
5. The two statements

   (i) x = b * c

   (ii) y = a + x

**(B)** Generate code for the following sequence assuming that x, y, and z are in memory locations:

```
        if x < y goto L1
        z = 0
        goto L2
L1:     z = 1
```