National Institute of Technology, Tiruchirapppalli - 15
Department of Computer Science and Engineering
CYCLE TEST 2
CSPC41 – Principles of Compiler Design

Course/Department **: B.Tech./CSE** Batch : **2018-2022**
Semester/Section **: VII** Session **: JUL/2021**
Date and Time **: 12-11-2021 & 10.00 AM – 11.00 AM** Marks **: 15**

Answer ALL Questions with proper steps and justification.
Draw diagrams wherever necessary.

1. Construct the CALR parsing table for the following grammar by using LR(1) items. **(3)**
   $S \rightarrow CB|BC$
   $C \rightarrow Cad|d$
   $B \rightarrow BaC|a$

2. Write the three-address code for the following code fragment. Identify the basic blocks in the resultant three-address code. **(4)**

```
for (i = 0; i < n; i++)
    count[(arr[i] / exp) % 10]++;
for (i = 1; i < 10; i++)
    count[i] += count[i - 1];
for (i = n - 1; i >= 0; i--) {
    output[count[(arr[i] / exp) % 10] - 1] = arr[i];
    count[(arr[i] / exp) % 10]--;
}
for (i = 0; i < n; i++)
    arr[i] = output[i];
```

3. Give the three-address code in triples representation for the expression x := b[i] + c ^ d * t - 2. **(2)**

4. Generate the three-address code for the following pseudocode using backpatching. Show the annotated tree in computing goto destinations. **(3)**

```
while i < j and j < n  or k < n do
        c = 1
        if c < j then
                x = x + c
        else
                x = 2 * c
        c = c + 1
        j = j + 1
        k = k * 2
end
a = x + n
```

5. Write target code in assembly language (hypothetical machine) for the following three-address code fragment and find its cost. You can assume that variable *a* is already loaded to register R0 by some statements in the code before line 101, whereas registers for remaining variables are not allocated yet. In order to perform an arithmetic operation, at least one of the operands should be in register. **(3)**
   101: *x* := *a* + *x*
   102: *y* := *b* * *2*
   103: *a* := *x* - *y*