VIT UNIVERSITY
(Estd. u/s 3 of UGC Act 1956)

Final Assessment Test – November 2017

Course: **CSE2002 - Theory of Computation and Compiler Design**

Class NBR(s):**0520 / 1689 / 1691 / 1692 / 1693 / 1695 / 1697 / 1698 / 1699 / 7655**

Slot: **G1+TG1**

Time: **Three Hours**

Max. Marks: **100**

PART – A (8 X 5 = 40 Marks)

Answer ALL Questions

1. Write the regular expression for the following language.

   L1 = {w| w has alternate 0s and 1s} Example. 101, 01010, 010101, 1010

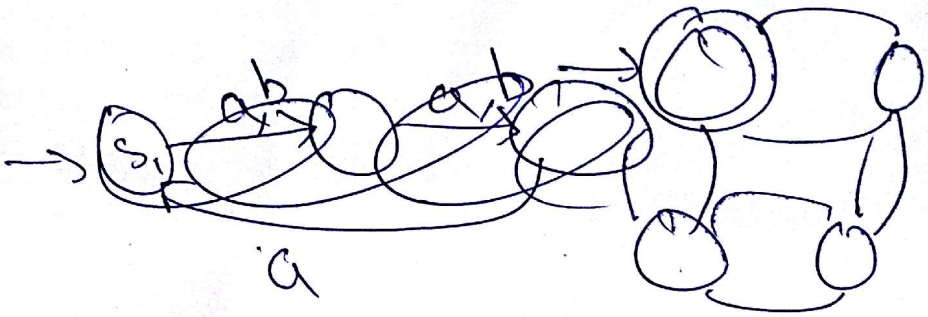
   L2= {w| every two successive 0s are followed by a 1 } Example: 0,10, 01, 110010,101001

2. Draw two DFAs D1 and D2 that accept languages L1 and L2 respectively over {a,b} such that L1 = { w| w contains even number of a's } and L2 = {w| w ends with ab }. Construct a DFA that accept the language L = {w| w contains even number of a's and ends with ab} using D1 and D2 only.

3. Construct an ε-NFA that accept the language represented by the regular expression ((ab)*+ (bb)*)aa.

4. Minimize the DFA defined by following table using Myhill-Nerode Theorem (Partition method).

	a	b
->1	2	4
2*	3	5
3	6	6
4*	5	3
5	6	6
6*	7	6
7	7	7

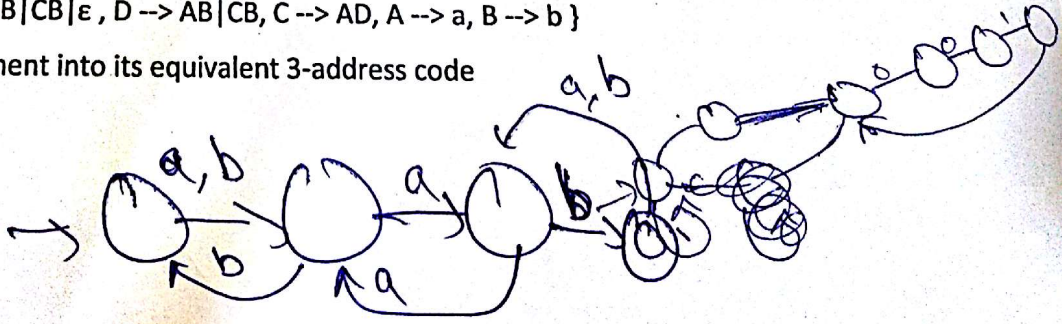


5. Convert the following CFG in to its CNF. G = { S --> AB|0B, A --> 001|ε, B --> 11A }

6. Verify whether the string 'aaabba' is belongs to the language described by the grammar G given below using CYK algorithm. G ={ S --> AB|CB|ε , D --> AB|CB, C --> AD, A --> a, B --> b }

7. Convert the following code segment into its equivalent 3-address code

```
a = b + (b /c) – (c * d);
if a > 100
    s = a;
else
    s = 100;
```



8. What are possible lists maintained for binary expressions while using backpatching? Write the task of the operations makelist(), merge(), and backpatch() used in backpatching.

PART – B (6 X 10 = 60 Marks)

Answer any SIX Questions

9. Write the primary task of each phase of a compiler. Mention what are the inputs and outputs of each phase with a suitable example. Categorize each phase into machine dependent and machine independent phase with justification. **[4+4+2]**

10. Construct the DFA directly from the regular expression a*b(a|b)*b without constructing NFA.

11. Design a PDA to accept the language L = $\{a^2b^{2n}c^n \mid n >= 1\}$. Show that the string 'aabbbbcc' is recognized by the above PDA by drawing possible instantaneous descriptions during processing the strings.

12. Create SLR parsing table for the following grammar by constructing LR(0) automata.
G = {S --> A ^ B , A --> C, B --> B+C|(B)|C, C --> id }. Show the process of parsing of the string "id^(id+id)" using SLR parsing table of G.

13. a) Write short notes on parallelizing compiler. **[5]**

    b) Find FIRST and FOLLOW of the symbols of following grammar: { S --> ABCDE, A --> a|ϵ, B --> b|ϵ, C --> c, D --> d|ϵ, E --> e|ϵ} **[5]**

14. Highlight the various issues of target code generation in a compiler. Identify the basic blocks with leaders in the following code segment. Draw the flow graph using the basic blocks of the code. **[3+4+3]**
    a) ifFalse num goto 11
    b) ifFalse counter goto 11
    c) t1 = num % 10
    d) t2 = sum + t1
    e) sum = t2
    f) t3 = num / 10
    g) num = t3
    h) t4 = counter -1
    i) counter = t4
    j) goto 1
    k) ST sum

15. Draw the state diagram and transition table for a Turing machine that increments a binary number. Assume that the input tape contains at least one non-blank symbol. For example, if the binary representation of 11 is initially on the tape ..B1011B.., then the output should be the binary representation of 12, B1100B.