|  |  |  |
| --- | --- | --- |
|  | **PES University, Bangalore**  (Established under Karnataka Act No. 16 of 2013) | UE18CS254 |
| **END SEMESTER ASSESSMENT (ESA) Model QP**    **UE18CS254- Theory of computation** | | |
| Time: 3 Hrs Answer All Questions Max Marks: 100 | | |

|  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| 1. | a) | What is a regular language?Give four differences between deterministic finite automaton(DFA) and non deterministic finite automaton(NFA).  Language(L) is called regular iff there exists some deterministic finite automata that accepts language(L).   |  |  | | --- | --- | | Distinct next state(DFA) | Different possible next states(NFA) | | DFA cannot use empty string transition | NFA can use empty string transition | | DFA requires more space | NFA requires less space | | Backtracking is not allowed | Backtracking is allowed | | 5 |
| b) | Construct a DFA to accept a's and b's starting with ab. | 5 |
| c) | Construct an NFA for a\*+(ab)\*  Sol in notes | 5 |
| d) | Convert the following NFA to its equivalent DFA.    Let Q’ be a new set of states of the Deterministic Finite Automata (DFA).  Add transitions of start state q0 to the transition table T’.   |  |  |  | | --- | --- | --- | | **State / Alphabet** | **a** | **b** | | →**q0** | q0 | {q0, q1} |   New state present in state Q’ is {q0, q1}.  Add transitions for set of states {q0, q1} to the transition table T’.  New state present in state Q’ is {q0, q1, q2}.  Add transitions for set of states {q0, q1, q2} to the transition table T’.  Since no new states are left to be added in the transition table T’, so we stop.  States containing q2 as its component are treated as final states of the DFA.    Finally, Transition table for Deterministic Finite Automata (DFA) is-   |  |  |  | | --- | --- | --- | | →**q0** | q0 | {q0, q1} | | **{q0, q1}** | q0 | \*{q0, q1, q2} | | **\*{q0, q1, q2}** | q0 | \*{q0, q1, q2} | | 5 |
|  | | | |
| 2. | a) | Obtain regular expression for the following:  i)Language consisting of strings of a’s and b’s that begin with at least two a’s and end with an even number of b’s.  aa(a)\*.(bb)\*  ii)Language consisting of strings of a’s and b’s whose length is a multiple of 3.  ((a+b)(a+b)(a+b))\* | 6 |
| b) | Convert the following regular expression to NFA.  (1+(01))\*00(1+(10))\*  sol in notes | 5 |
| c) | Describe the language of the following regular expression:  strings of a’s and b’s with no consecutive b’s.  Sol:(b+𝞴)(a(a+𝞴)\*(b+𝞴)\*)(a+𝞴)\* | 4 |
| d) | Using pumping lemma show that the languageL={|n⩾0} is not regular.  Let L be regular.  x=,|x|=2n ,x=aaaaabbbbb, |uv|⩽n and |v|⩾1  u∈L for i=0,1,2,3,....  If i=o then one a is less than b’s. Therefore the language is not regular. | 5 |
|  | | | |
| 3. | a) | Construct grammar for the following regular language:  i)Binary strings in which every 0 is followed by 11.  Sol:S→11S|011S|𝞴  ii)Set of all even palindromes over {a,b}.  Sol:S→aSa|bSb|𝞴 | 5 |
| b) | Given the following grammar,  S → a / abSb / aAb  A → bS / aAAb  Show that the grammar is ambiguous by generating two derivations for abababb. | 5 |
| c) | Use the CYK algorithm to determine whether or not the given string baaba belongs to the grammar. S → AB / BC  A → BA / a  B → CC / b  C → AB / a | 8 |
| d) | Convert the following grammar into Griebach Normal Form.  S→abS|baS|𝞴  Sol:S→aBS|bAS|𝞴  B→b  A→a | 2 |
|  | | | |
| 4. | a) | Construct Push Down Automata that accepts the language L over Σ={a,b} :  L={|n≥1}  Sol:from notes | 6 |
| b) | State and prove pumping lemma for context free languages.  Sol:  Pumping Lemma for CFL states that for any Context Free Language L, it is possible to find two substrings that can be ‘pumped’ any number of times and still be in the same language. For any language L, we break its strings into five parts and pump second and fourth substring.  Pumping Lemma, here also, is used as a tool to prove that a language is not CFL. Because, if any one string does not satisfy its conditions, then the language is not CFL.  Thus, if L is a CFL, there exists an integer n, such that for all x ∈ L with |x| ≥ n, there exists u, v, w, x, y ∈ Σ∗, such that x = uvwxy, and  (1) |vwx| ≤ n  (2) |vx| ≥ 1  (3) for all i ≥ 0: uwy ∈ L | 6 |
| c) | Convert the following CFG to PDA.  S→aSA|bSb|𝞴  Sol  i)Convert to GNF  S→aSA|bSB|𝞴  A→a  B→b  S→aSA|aA|bSB|bB|𝞴  diagram from notes | 8 |
|  |  |  |
|  | | | |
| 5. | a) | Construct a Turing Machine for language L = {0n1n2n | n≥1} | 8 |
| b) | Explain cantor’s diagonalization.    First we assume that the given set R is enumerable.Our objective is to set up a contradiction to prove that our assumption is wrong.From diagram each row is as real number and each column is a decimal place.Consider the left-to-right diagonal which contains the 1st decimal place of the first real number and so on.If we read this diagonal as a real number ,we get a number.We now complement it to invert the elements of the diagonal.This new number differs from every enumerated real number in the corresponding diagonal element.We contradict our assumption | 6 |
| c) | Explain universal turing machine. | 6 |
| d) |  |  |