|  |  |  |
| --- | --- | --- |
| PES University Logo.jpg | **PES University, Bangalore**  (Established under Karnataka Act No. 16 of 2013) | **UE18CS254** |
| **END SEMESTER ASSESSMENT (ESA) B. TECH IV SEMESTER- May 2020**  **Theory of Computation** UE**18CS254**  **Model Question Paper- Compiled by Prof.R.Bharathi** | | |
| Time: 3 Hrs Answer All Questions Max Marks: 100 | | |
|  | | |
| **Note: All answers must be precise and to the point.** | | |

|  |  |  |  |
| --- | --- | --- | --- |
| 1. | a) | Explain the terms Grammar and Formal Language with example.  A grammar G is defined as a quadruple G =(V, T, S, P),  where V is a finite set of objects called variables,  T is a finite set of objects called terminal symbols,  S ∈ V is a special symbol called the start variable,  P is a finite set of productions.  Set of strings, that are considered in the context of computing machines and that are not natural to humans is called Formal languages. | 4 |
| b) | Define the language accepted by a DFA and language accepted by NFA  The language accepted by a dfa M = (Q, Σ,δ, q0,F) is the set of all strings on Σ accepted by M. In formal notation,  The language L accepted by an nfa M = (Q,Σ,δ, q0,F) is defined as the set of  all strings accepted in the above sense. Formally,  In words, the language consists of all strings w for which there is a walk labeled w from the initial vertex of the transition graph to some final vertex. | 6 |
| c) | For S = {a,b}, construct DFA, that accept the sets consisting of all strings with no more than three a’s.  Break it into three cases each with an accepting state: no a’s, one a, two a’s, three a’s. A fourth a will then send the dfa into a nonaccepting trap state. | 4 |
|  | d) | Convert the -NFA given as transition table to equivalent DFA   |  |  |  |  | | --- | --- | --- | --- | | **State** | **Input = *a*** | **Input = *b*** |  | | *q*0 | {*q*0, *q*1} | {*q*1} | {} | | *q*1 | {*q*2} | {*q*1, *q*2} | {} | | \* *q*2 | {*q*0} | {*q*2} | {*q*1} |   No more new states | 6 |
|  | | | |
| 2 | a) | Construct Regular expression for each of the following:   1. L = {anbm | n >=4 and m<=3}   RE = [aaaaa\*][   1. L = { w | w ∈ {0,1}\* and |w| mod 3 =0}   RE = [(0+1) (0+1) (0+1)]\*   1. L = { w | w ∈ {a,b}\* and every a in w is immediately precede and followed by b}   RE = (b + bab)\* | 6 |
| b) | Convert the given Finite automata into equivalent regular expression    Steps:   1. q3 is a dead state, remove it     Reg Exp = (01+10)\* | 4 |
|  | c) | State and prove pumping lemma for regular languages.  Let L be an infinite regular language. Then there exists some positive  integer m such that any w ∈ L |w| ≥ m can be decomposed as, w= xyz, with |xy|<=m, and |y|>=1, such that wi = xyiz, is also in L for all i = 0, 1, 2,….  Proof:  If L is regular, there exists a dfa that recognizes it. Let such a dfa have states labeled q0, q1, q2,…, qn. Now take a string w in L such that |w| ≥. = n +1.  Since L is assumed to be infinite, this an always be done.  Consider the set of states the automaton goes through as it processes w, say  q0qiqj……..qf  Since this sequence has exactly |w| + 1 entries, at least one state must be repeated, and such a repetition must start no later than the nth move. Thus, the sequence must look like  q0qiqj……qr…….qr…..qf  indicating there must be substrings x, y, z of w such that  with |xy| ≤ n+1 = m and |y| ≥ 1. From this it immediately follows that  as well as  and so on, completing the proof of the theorem. | 6 |
|  | d) | Find Regular Grammar for the given language on {a,b}  L = (w | ( na(w) -nb(w) ) mod 3 =1} | 4 |
|  | | | |
| 3. | a) | Construct Context free grammar for the following languages,   1. L = { anbmck |k=n+m}   S aSc |T|  T bTc|  ii) L = { anbm | n m}  S T|U  T aTb |aT| a  U aUb|Ub|b | 4 |
| b) | Define instantaneous description of a pushdown automaton and context free grammar.  Answer:  The possible ways in which the npda can proceed is determined through a triplet (q,w,a), where q is the state of the control unit, w is the unread part of the input string, and u is the stack contents (with the leftmost symbol indicating the top of the stack), is called an instantaneous description of a pushdown automaton.  A grammar G = (V, T, S, P) is said to be context-free if all productions in P have  the form Ax , where A ∈ V and x ∈ (V ∪ T)\*. | 4 |
| c) | Construct npda for L ={w ∈ {a,b}\* | na(w) > nb(w)}  Example :aaaaabbb  The last transition should be  At last when is encountered, if ToS contains a, then change to final state and do not alter the contents of the stack. The npda is given by, | 6 |
| d) | Define Leftmost derivation, Ambiguous grammar and inherently ambiguous grammar.  A derivation is said to be leftmost if in each step the leftmost variable in the sentential form is replaced.  A context-free grammar G is said to be ambiguous if there exists some w ∈ L(G) that has at least two distinct derivation trees. Alternatively, ambiguity implies the existence of two or more leftmost or rightmost derivations.  If L is a context-free language for which there exists an unambiguous grammar, then L is said to be unambiguous. If every grammar that generates L is ambiguous, then the language is called inherently ambiguous. | 6 |
|  | | | |
| 4. | a) | Explain when the PDA is deterministic and define deterministic pda.  A deterministic pushdown accepter (dpda) is a pushdown automaton that never has a choice in its move.  A pushdown automaton M = (Q, Σ, Γ, δ, q0, z, F) is said to be deterministic  if it is an automaton, subject to the restrictions that, for every q ∈ Q, a Σ ∪{λ} and b ∈ Γ,  1. δ(q, a, b) contains at most one element,  2. if δ (q, λ, b) is not empty, then δ (q, c, b) must be empty for every c ∈ Σ.  The first of these conditions simply requires that for any given input symbol and any stack top, at most one move can be made.  The second condition is that when a λ-move is possible for some configuration, no input-consuming alternative is available. | 4 |
| b) | Construct an npda corresponding to the grammar  S aABB|aAA,  A aBB|a,  B bBB|A.  Answer:  First, we convert the cdf to the corresponding GNF  S aABB|aAA,  A aBB|a,  B bBB| aBB|a.  We construct npda M = ({q0, q1, q2}, {a, b}, {S, A,B,Z}, δ , q0,Z, {q2}) by the following steps:  (a) Initial: δ (q0, λ, z) = {(q1, Sz)}  (b) Process the input:  \_ For S aABB|aAA : δ (q1, a, S) = {(q1,ABB), (q1;AA)}  \_ For A aBB|a : δ (q1, a,A) = { (q1,BB), (q1, λ)}  \_ For B bBB|aBB|a : δ (q1, b,B) = { (q1;BB)}, δ (q1, a,B) = {(q1,BB), (q1, λ)}  (c) In the end: δ (q1, λ, z) = {(q2; λ)} | 6 |
| c) | Eliminate all λ -productions from  S AaB|aaB,  A λ,  B bbA| λ.  Answer:  A procedure of removing all λ -productions is as follows.  The nullable variable set is VN = {A,B}  A λ and B λ can be removed after adding the following new productions  S aB|Aa|a|aa  B bb  Finally, we have  S AaB|aaB|aB|Aa|a|aa  B bbA|bb | 4 |
| d) | Use the CYK algorithm to determine whether the string abb is in the language generated by  S AB,  A BB|a,  B AB|b.  **Answer:**  **Using the formula**    For the string abb: Firstly, we have,   |  |  |  | | --- | --- | --- | | V1,3 = {A} |  |  | | V1,2= {S,B} | V2,3 = {A} |  | | V1,1 = {A} | V2,2 = {B} | V3,3 = {B} |   Because V1,3 = {A}, S V1,3, we conclude that abb is not in the language  generated by the given grammar, i.e., abb L(G), by using the CYK algorithm. | 6 |
|  | | | |
| 5. | a) | Explain Turing machine.  A Turing machine is an automaton whose temporary storage is a tape. This tape is divided into cells, each of which is capable of holding one symbol. Associated with the tape is a read-write head that can travel right or left on the tape and that can read and write a single symbol on each move.    A Turing machine M is defined by  M = (Q,Σ,Γ,δ,q0,b ,F),  where  Q is the set of internal states,  Σ is the input alphabet  Γ is the finite set of symbols called the tape  alphabet,  δ is the transition function,  b∈Γ is a special symbol called the blank,  q0 ∈ Q is the initial state,  F ⊆ Q is the set of final states  The transition function δ is defined as  δ : Q × Γ → Q × Γ × {L,R}. | 4 |
| b) | Given two positive integers x and y in unary notation separated by a single zero. Construct a Turing machine to compute x+y.  Answer:  Unary number is made up of only one digit 0 or 1  Let x and y are two unary numbers over {1}+  Both x and y are stored on tape separated by 0 | 6 |
| c) | Define Recursive and Recursively Enumerable Languages  A language L on Σ is said to be recursive if there exists a Turing machine M that accepts L and that halts on every w in Σ+. In other words, a language is recursive if and only if there exists a membership algorithm for it.  A language L is said to be recursively enumerable if there exists a Turing machine that accepts it.  This definition implies only that there exists a Turing machine M, such that, for every w ∈ L,  with qf a final state. The definition says nothing about what happens for w not in L; it may be that the machine halts in a nonfinal state or that it never halts and goes into an infinite loop. | 6 |
|  | d) | Write notes on Post Correspondence Problem(PCP).  The PCP was first introduced by Emil Post in 1946. The problem was found to have many applications in the theory of formal languages. The goal is to prove this problem about strings to be Undecidable  Given two list of strings A and B(equal length):  A = w1, w2, …, wk  B = x1, x2, …, xk over an alphabet .  The PC problem is to determine if there is a sequence of one or more integers i1, i2, …, im such that:  w1wi1wi2…wim = x1xi1xi2…xim  (wi, xi) is called a corresponding pair. An example for Post Correspondence Problem,   |  |  |  | | --- | --- | --- | |  | A | B | | i | wi | xi | | 1 | 1 | 111 | | 2 | 10111 | 10 | | 3 | 10 | 0 |   This PCP has a solution : 2,1,1,3:  w2w1w1w3 = x2x1x1x3 =101111110 | 4 |