Compiler Design (CS348)

Mid Semester Exam- March 2022

Open Ended Questions

Note: i) Figures in bracket () indicate marks

ii) Assume suitable data wherever necessary, mention your assumptions clearly.

1. Describe the role of lexical analysis in the design of a compiler. (3marks)What does the following regular expressions mean (2marks*2=4marks). Construct NFAs for the same by following Thompson's Construction rules.(4 marks*2=8 marks)
   i. (a|b)*a(a|b)abb
   ii. (0|1)0*1*01
2. Is it possible to design a compiler without distinct lexical analysis phase? Justify your answer with reference. (3marks) What does the following regular expressions mean (2marks*2=4marks). Construct NFAs for the same by following Thompson's Construction rules.(4 marks*2=8 marks)
   i. (a|b)*abb(a|b)*
   ii. (a*|b*)*
3. Design an algorithm to recognize Lex-lookahead patterns of the form r1|r2 (5 marks), where r1 and r2 are regular expressions. Show how your algorithm works on the following inputs. (5mark*2 =10marks)
   i. (abcd | abc)/d
   ii. (a|ab) / ba
4. Write down the regular expressions for the following (1mark*5 =5marks). Construct the NFA for any two regular expressions of the above by following Thompson's Construction rules.(5 marks *2=10 marks)
   i. To check correct syntax for the E-mail address.
   ii. To check an IP address for correct syntax, i.e. four groups of 1-3 digits separated by dots(.).
   iii. Any decimal number that is a multiple of 5
   iv. Binary strings such as a '0' is always followed by a '1'
   v. User defined Function names of C language
5. Construct a finite automata which will accept strings of 0's and 1's containing odd number of 0's and even number of 1's. (5 marks). Convert that NFA to DFA(5 marks) and obtain minimized DFA for the same by using partition method(5 marks).

6. What does the following regular expressions mean? (1mark*5 =5marks) Construct the NFA for any two regular expressions of the above by following Thompson's Construction rules.(5 marks *2=10 marks)
   (a) a(ab)*a
   (b) [0-7][0-7]*
   (c) (0|1)*0(0|1) (0|1) (0|1)
   (d) (a|b)*abb
   (e) (a|b)*a(a|b)(a|b)(a|b)

7. A)For the statement d=a + b* c % b + b /c, write all the compilation phases with input and output to each phase (6 marks)

   B) Consider the given grammar
   S -> qQRr
   Q-> x | ϵ
   R-> t | ϵ
   i. State whether the given grammar is LL(1) grammar. If so, give reason (1 mark)
   ii. Compute FIRST & FOLLOW (3 marks)
   iii. Construct Predictive Parsing Table (3 marks)
   iv. Stack implementation for the string : qxtr$ (2 marks)

8. A)With respect to parsing explain the following terminologies through examples
   1)Ambigious Grammar 2) Left Recursion 3) Precedence & Associativity 4) Role of Parser (4 * 2 = 8 marks)

   B) For the given grammar compute1) FIRST and FOLLOW 2) Construct Predictive Parsing Table.
   S-> KL | oNk
   K-> kl | m
   L->nM
   M->oM | ϵ
   N->pN | ϵ

   FIRST & FOLLOW(3 marks)
   Construction of parsing table (4 marks)

9. a) Explain the applications of Compiler? (5 marks)
   b) Mention where you have seen or used these applications(5 marks)
   c) What are the steps involved in conversion of .C file to .EXE file(5 marks)

10. a) Explain with suitable example the impact of compilation process if the semantic analysis phase is not working(8 marks)
    b) Explain with suitable example , the role of code optimization in compilation process (7 marks)

11. Define Ambiguous Grammar with suitable Example.(3 marks). Check whether following Grammar is ambiguous or not (4marks*3=12 marks)
    a. S → aSb/bSa/SS/ε
    b. S -> SS | AB , A -> Aa | a , B -> Bb | b
    c. S -> AA , A -> aA , A -> b

12. Briefly define the concept of Derivation(3 marks) Which derivation is more suitable for top down parsing and Why? (2 marks). Derive the leftmost and rightmost derivation for the following Grammar (5marks*2= 10marks)
    d. S -> bB | aA , A → b / bS / aAA , B → a / aS / bBB for input string w= bbaababa
    e. S → A1B , A → 0A / ∈ B → 0B / 1B / ∈ for input string w = 00101

13. Explain how Ambiguity is resolved using operator precedence(5 marks). Disambiguate the following grammar using operator precedence. (5marks *2=20 marks)
    a. E -> E * E | a , F -> F+F | b , G -> G-G | id
    b. S-> S* A | A , A -> A^A | cd , B -> B+B | id

14. Discuss various problems of Top Down Parsing. (5 marks). Solve the Following
    a. Eliminate left Factoring from the given Grammar (5 marks).
       S-> aSb | abS |ab

    b. Eliminate left Recursion from the given Grammar (5 marks).
       S-> SaA | b , A-> AaB | d, B-> Ba | b

15. State the Algorithm for calculating First and Follow of given Grammar. (5 marks). Calculate the First and Follow set for given Grammar.
    a. S-> Bb | Cd , B-> aB | ε , C-> cC |ε (5 marks).

b. S-> AaAb | BbBa , A-> ε , B-> ε (5 marks).

16. Construct the parsing table for given Grammar and implement the parsing actions for given string w (7 marks)

S-> (L) | a , L-> SL' , L'-> *SL' |ε

a. Input string w = (a*a*a)
b. Input string w = ((a*a)*(a*a))

17. Convert the given Grammar into LL(1) Grammar

S-> SaA/b , E-> EbB/ id , F-> aA/ abbC/acd/aB , C-> bcS/bd/baB, A-> ε B-> ε

(15 marks- Includes 5 marks for accurate Parse table)