CD Lab Viva Questions

[redacted].

1. Differentiate between a Scanner and a Scanner Generator?
2. What is LEX? Explain.
3. What is the difference between a token, pattern and lexeme?
4. What is the purpose of lexical analysis? In how many ways can a Lexical analyzer be constructed? What are they?
5. What are the parts of a LEX program? Explain.
   What parts are optional?
6. How is a LEX program compiled? Explain the files generated.
7. Give LEX definitions for integer, real no, identifier.
8. Explain yylex, yylval, yyin, yytext, yywrap.
9. How do you tokenize #include <stdio.h>?
   What are non tokens in C? Give examples.
10. What are metacharacters? Explain any 5.
11. What is a Recursive Descent parser? Why is it called s
12. What are topdown and bottomup parsers? List few.
13. Give the rules for finding FIRST and FOLLOW sets.
    Explain with examples.
14. What are the pre requisites for a Recursive Descent parse
15. What is Left Recursion? Explain with an example.
    Why should it be eliminated.
16. What is Left Factoring? Explain with an example.
    Why should a grammar be Left factored?
17. Why are FIRST and FOLLOW sets required?

18. What is a shift Reduce parser ? Explain.

19. What are the conflicts in shift reduce parsing? Exp.

20. What is a handle ? Give an example.

21. What is a viable prefix ? Give an example.

22. Differentiate between a parser and a parser generator

23. What is YACC ? Explain. Give the full form of YACC

24. What is the purpose of syntax analysis?

25. What are the parts of a YACC program ? Explain.
What parts are optional?

26. What are the different Bottom up parsing Techniques ?

27. Which parsing technique makes use of YACC?

28. What are the different Top down parsing Techniques?

29. What is YYSTYPE, YYPARSE, YYERROR, yylex, yylval, YYWRAP.

30. What is the syntax of a YACC rule? Explain with an example

31. How is a YACC program compiled? Explain the files generated.

32. What is the default type of identifier in YACC?

33. How is precendence and associativity of operators mentioned in a YACC program? Give an example.

34. Can a YACC program be standalone?

35. How are tokens sent to yacc? (how many ways?)

What are -ll and -ly options in compiling?

What are $, $$, $i ? Give examples.

38. What is intermediate code? Give example.

39. What is three address code? Give the syntax and example

40. Why is three address code required? How is it represented? What are the different representation? Explain with examples.

41. What is the default action for E : T in yacc?

42. How is E → E+T|T represented in YACC?

43. What is single and doube address code? Which is efficient?

44. Differentiate between target code and intermediate code?

45. Give target code for the following expressions?

46. Give intermediate code (3 address code) for the following expressions!

47. What is a compiler, translator, preprocessor, assembler, loader and linker.

48. Differentiate between FRONT END and BACK END. Give examples.

49. What are the different errors in lexical analysis, syntax analysis.

50. What are the different compiler construction tools

51. What other tasks are performed by lexical analysis besides tokenization?

52. What is the output of Lex compiler and yacc compiler?

53. What are the optional parts of a lex and yacc program

54. What are the minimal (Smallest) lex and yacc programs?

55. What are the advantages of LEX and YACC over handwritten scanners and parsers?

56. LEX uses ______ where as YACC uses ______.
(Regular Expressions, CFG's)

57. What is the difference between Booting and Bootstrapping?

58. Give 3 address code for 'if', 'for', 'while' & 'switch'

59. What is peephole optimization?

60. What is loop optimization?

61. What are the different code optimization techniques?
Why should code be optimized?

62. What is constant folding?

63. What are machine dependent optimizations?
Name them.

64. What are machine independent optimizations?
Name them.

65. Differentiate static vs dynamic storage organization

66. Differentiate static vs dynamic scope?

67. What is a symbol table? What are its contents?

What is an activation record? What are its contents? When is it created?

69. What is a Basic Block and a Flow graph?

70. Differentiate between RISC and CISC?

71. What are the different parameter passing mechanisms

72. Explain Call-by-Value, Call-by-Reference, Call-by-Name.

73. What is Aliasing?

74. What are language processors? give examples.

75. What type of grammars YACC cannot parse?

76. What is Backpatching, Type Checking, Type Equivalence.

77. What is an ambiguous grammar? give example

78. What is LALR parsing? Compare with LR(1), SLR(1

79. What is an LR(1) item?

80. What is BruteForce parsing?

81. Define L.L.(1) parsing? Which grammars are allowed to be LL(1) parsed? Is every LL(1) unambiguous?

82. What happens when YYPARSE ( ) is called?

83. How does LEX work in conjunction with YACC. In how many ways and how can you include the output of Lex into yacc.

84. What is the advantage of generating intermediate code

What

85. Differentiate between Intermediate Code and Target Code

86. What is a DAG? What is its advantage?

87. Construct DAG for the following expression / 3 address code

88. What is the difference between a tree and a DAG?

89. What is an SDT and an SDD? What is their use?

90. What is a Regular Expression? Give an example.

91 What is a CFG? Give an example.

92. How do you declare C variables in LEX?

93. What is ctype.h? What does 'isalpha()' return?

94. What is YYSTYPE? What is its use?

95. What is a pseudo variable? Give an example.

96. Explain Recursive Descent Parser with an example.

97. Generate Target Code for the given postfix expression.

98. What are LR(0) items? What does an item indicate?

99. What are the error recovery strategies used by Lexical and Syntax Analyzers?

100. What is the difference between Top Down and Bottom Up parsers? (Bottom up parsers are more powerful and there are no restrictions on grammars