



Dr.G.R.Damodaran College of Science

(Autonomous, affiliated to the Bharathiar University, recognized by the UGC)Re-
accredited at the 'A' Grade Level by the **NAAC** and ISO 9001:2008 Certified
CRISL rated 'A' (TN) for MBA and MIB Programmes

III B.Sc CS [2015-2018]

Semester V

Core: System Software and Compiler Design - 507A

Multiple Choice Questions.

1. _____ is used to represent the differences between the semantics of two domains.
- A. Semantic gap.
 - B. Execution gap.
 - C. Both semantic & execution gap.
 - D. Lexical gap

ANSWER: A

2. The gap between the application domain and programming language domain is _____.
- A. execution gap.
 - B. design gap.
 - C. specification gap.
 - D. semantic gap.

ANSWER: C

3. _____ is a language processor which bridges the execution gap.
- A. Compiler
 - B. Interpreter
 - C. Spectrum
 - D. Migratory.

ANSWER: B

4. The two types of language processing activities are _____.
- A. program generation & Program execution.
 - B. program generation & program location.
 - C. program execution & program relocation.
 - D. application & specification.

ANSWER: A

5. The two popular models for program execution are _____.
- A. translation & assembler.
 - B. interpretation & language translator.
 - C. language translator & translation.
 - D. translation & Interpretation.

ANSWER: B

6. The CPU uses _____.
- A. program counter.

- B. operands
- C. interpretation.
- D. program relocation.

ANSWER: C

7. Language processing is the combination of_____.

- A. source program
- B. target program
- C. analysis of source program
- D. synthesis of target program

ANSWER: B

8. Which is not a Language processing specification rules?

- A. Lexical rule.
- B. Offset rule.
- C. Semantic rule.
- D. Syntax rule.

ANSWER: A

9. Toy compiler consists of two different ends namely_____.

- A. front end
- B. back end
- C. front & back ends
- D. IR

ANSWER: B

10. IR produced by the front end consists of two components namely_____.

- A. tables of information & intermediate code.
- B. tables of information & temporary area.
- C. IC & information.
- D. IC & SP.

ANSWER: A

11. The symbol table is built during _____.

- A. lexical analysis.
- B. semantic analysis.
- C. syntax analysis.
- D. SP & TP analysis.

ANSWER: A

12. The IC for a statement is a _____

- A. string of tokens.
- B. tokens
- C. class code.
- D. strings

ANSWER: A

13. Semantic analysis determines the meaning of_____

- A. sub tree in the IC.
- B. sub tree in the PL.
- C. sub tree in the IC & PL.

D. sub tree in SP.

ANSWER: A

14. The back end in the Toy compiler performs _____

- A. memory allocation.
- B. code generation
- C. memory allocation & code generation.
- D. linking

ANSWER: B

15. Code generation uses the knowledge of _____

- A. the target architecture.
- B. the target computer.
- C. instruction
- D. addressing nodes

ANSWER: A

16. _____ is used for type conversion operations.

- A. Conversion.
- B. Compilation
- C. Code generator.
- D. Code Executer.

ANSWER: C

17. Grammars are classified on the basis of _____.

- A. nature of production.
- B. nature of reduction.
- C. nature of comparison.
- D. primary language.

ANSWER: A

18. Use of _____ codes for machine instructions eliminates the need to memorize numeric operations

- A. symbolic code.
- B. numeric code.
- C. mnemonic operation.
- D. assembly

ANSWER: C

19. _____ can be associated with data or instruction.

- A. Data declaration.
- B. Machine codes.
- C. Mnemonic opcodes
- D. Symbolic operands.

ANSWER: D

20. _____ instruction moves a value between a memory word and a register.

- A. MOVEM
- B. MOVE
- C. MOVER
- D. MOV

ANSWER: B

21. Converse is true for the _____ instruction.

- A. MOVEM
- B. COMP
- C. MOVE
- D. MOVER

ANSWER: A

22. All arithmetic operations performed in a register sets a _____ .

- A. machine opcode.
- B. mnemonic opcode.
- C. condition code.
- D. symbolic code.

ANSWER: C

23. The _____ statements reserves areas of memory and associates names with them.

- A. declare storage.
- B. imperatives
- C. declare constants.
- D. declare memory.

ANSWER: A

24. The _____ statements construct memory words containing constants.

- A. imperatives
- B. declare constants.
- C. memory allocation.
- D. declare storage.

ANSWER: B

25. The _____ instructs the assembler to perform certain actions during program assembly.

- A. assembler
- B. instructor
- C. assembler directives.
- D. analyzer

ANSWER: C

26. The first item of information depends on the _____ .

- A. name field.
- B. address field.
- C. machine opcode.
- D. source program.

ANSWER: D

27. A data structure used to implement memory allocation is _____ .

- A. location counters.
- B. data counter.
- C. memory counters.
- D. stack counter.

ANSWER: A

28. Two pass translation of an assembly language processor can handle _____ easily.

- A. mnemonic opcodes.
- B. forward references.
- C. instruction address.
- D. lc processing.

ANSWER: B

29. The Intel 8088 provides addressing capability up to _____ of primary memory.

- A. 16kb.
- B. 64kb
- C. 1mb
- D. 12kb

ANSWER: C

30. The Intel 8088 architecture provides _____ addressing modes.

- A. 08
- B. 16
- C. 24
- D. 32

ANSWER: C

31. FRT contains a set of _____ .

- A. stack.
- B. symbols
- C. tables
- D. linked lists.

ANSWER: D

32. The _____ operator performs the special function of creating a new memory operand.

- A. SEG
- B. BUFFER
- C. NEW_NAME
- D. THIS

ANSWER: D

33. The _____ statement indicates that a segment register contains a base address of a segment.

- A. BUFFER
- B. ASSUME
- C. BASE
- D. SEGMENT

ANSWER: B

34. Aligning the LC to its requisite boundary is called as _____ .

- A. LC boundary.
- B. LC alignment.
- C. LC format.
- D. LC routine.

ANSWER: B

35. Macros are used to provide a program generation facility through _____ .

- A. macro expansion.
- B. macro definition.

- C. macro specification
- D. macro generation.

ANSWER: A

36. _____ language is used to provide built-in facilities for writing macros.

- A. Java.
- B. C
- C. C++
- D. Dos

ANSWER: C

37. A macro is a unit of specification for program generation through _____ .

- A. expansion
- B. generation.
- C. specification
- D. specialization.

ANSWER: B

38. _____ kind of expansion can be readily identified.

- A. One.
- B. Two.
- C. Three
- D. Four

ANSWER: B

39. Lexical expansion implies replacement of a character string by another character string during _____ .

- A. program execution.
- B. program definition.
- C. program generation.
- D. program specialization.

ANSWER: C

40. Semantic expansion implies generation of instructions tailored to the requirement of a _____ .

- A. specific usage.
- B. execution usage.
- C. generation usage.
- D. logic usage.

ANSWER: A

41. A _____ is enclosed between macro header and macro end statement.

- A. macro prototype.
- B. macro expansion.
- C. macro definition.
- D. macros

ANSWER: B

42. A macro definition consist of _____ .

- A. a macro prototype statement.
- B. one or more model statement.
- C. macro pre-processor statement.

D. all the above.

ANSWER: D

43. A model statement is a statement from which an assembly language statement may be generated during _____.

- A. macro prototype.
- B. macro expansion.
- C. macro definition.
- D. macros

ANSWER: B

44. _____ statement is used to perform auxiliary function during macro expansion.

- A. Prototype.
- B. Pre-processor.
- C. Model
- D. None of the above.

ANSWER: B

45. _____ leads to macro expansion

- A. Macro end.
- B. Macro start.
- C. Macro call.
- D. Macro definition.

ANSWER: C

46. To differentiation between the original statement of a program and the statement relating from macro expansion, each expanded statement is marked with a _____ presiding its label.

- A. #
- B. -
- C. *
- D. +

ANSWER: D

47. The default flow control during macro expansion is _____.

- A. normal
- B. sequential
- C. direct flow.
- D. indirect flow.

ANSWER: B

48. The flow of control during macro expansion is implemented using a _____.

- A. macro sequential control.
- B. macro direct floe control.
- C. macro expansion control.
- D. macro indirect flow control.

ANSWER: C

49. A model statement consists of three type of _____.

- A. pointer.
- B. numbers
- C. string

D. data
ANSWER: C

50. The rule for determining the value of a formal parameter depending on the kind of _____.
- A. format.
 - B. parameter
 - C. value
 - D. numbers

ANSWER: B

51. _____ is a standard assumption of the absents of an explicit specification by the programmer.
- A. Default.
 - B. Data
 - C. Parameter
 - D. Integer

ANSWER: A

52. _____ may be defined to use both positional and keyword parameter.
- A. Assembler.
 - B. Compiler
 - C. Macro
 - D. Interpreters

ANSWER: B

53. A macro statement in a macro may constitute a call on another macro. Such calls are known as _____ .
- A. default macro call.
 - B. normal macro call.
 - C. nested macro call.
 - D. sub macro call.

ANSWER: C

54. Expansion of nested macro calls follows the _____ rule.
- A. FIFO (first in first out).
 - B. LIFO (last in first out).
 - C. FILO (first in last out).
 - D. FOLI (first out last in).

ANSWER: B

55. Expansion time variable are the variable which can only be used during the expansion of _____ .
- A. macro definition.
 - B. macro declaration.
 - C. macro execution.
 - D. macro call.

ANSWER: D

56. Local and global Expansion time variable are created through _____.
- A. definition.
 - B. declaration
 - C. execution
 - D. call

ANSWER: D

57. Values of Expansion time variable can be manipulated through the pre-processor statement _____.

- A. set.
- B. get
- C. put
- D. lock

ANSWER: A

58. The type, length and size attributes have the name _____.

- A. _T,_L,_S.
- B. T, L, S.
- C. &T, &, L,&S
- D. #T, #L, #S.

ANSWER: B

59. While writing a general purpose macro it is important to ensure execution efficiency of its _____.

- A. declared code.
- B. generated code.
- C. executed code.
- D. output code.

ANSWER: B

60. It is often necessary to generate many similar statements during the _____ of the macro.

- A. expansion.
- B. declaration
- C. call
- D. output

ANSWER: A

61. Many assemblers provide other facilities for _____ expansion.

- A. conditional.
- B. assignment
- C. logical
- D. relational

ANSWER: A

62. A table called the _____ is designed to hold the name of all macro definition programs.

- A. actual parameter table.
- B. macro name table.
- C. parameter default table.
- D. macro default table.

ANSWER: B

63. A table called _____ is designed to hold the value of formal parameter during the expansion of macro call.

- A. actual parameter table.
- B. macro name table.
- C. parameter default table.

D. macro default table.

ANSWER: A

64. If a macro call statement does not specify a value for some parameter, its default value would copy from _____.

- A. APT to PDT.
- B. PDT to APD.
- C. ATP to PTP.
- D. PTD to ADP.

ANSWER: B

65. APTAD means_____.

- A. which contains formal parameter name.
- B. which contains actual parameter name.
- C. which contains formal parameter value.
- D. which contains actual parameter value.

ANSWER: C

66. The stack consist of expansion records, each expansion record accommodating one set of _____.

- A. expansion time data structure.
- B. normal time data structure.
- C. expansion time data flow.
- D. normal time data flow.

ANSWER: A

67. When a nested macro call is recognized, a new expansion record is pushed on the stack to hold the_____ for the call.

- A. data flow.
- B. data structure.
- C. data control.
- D. data analysis

ANSWER: B

68. To design the pass structure of a macro-assembler we identify the function of a _____.

- A. macro expansion.
- B. macro declaration.
- C. macro pre-processor.
- D. macro definition.

ANSWER: C

69. _____ are expanded in Last In First Out.

- A. Macro call.
- B. Nested macro calls.
- C. Macro expansion.
- D. Macros

ANSWER: B

70. A compiler is a _____ that reads a program written in one language, and translates it into an equivalent program in machine language.

- A. software.

- B. instruction
 - C. phase
 - D. program
- ANSWER: D

71. The two parts of compilers are _____.

- A. lexical and synthesis.
- B. source program and target program.
- C. analysis and synthesis.
- D. static checkers and interpreters.

ANSWER: C

72. The _____ part breaks up the source program into constituent pieces and creates an intermediate representation of the source program.

- A. synthesis part.
- B. analysis.
- C. syntax.
- D. argumentation.

ANSWER: C

73. In analysis the operation implied by the source program are determined and recorded in a structure called _____.

- A. flat structure.
- B. bilateral
- C. line
- D. tree

ANSWER: D

74. Which is the tool that reads a program, analysis it and attempts to discover potential bugs without running the program?

- A. Structure editors.
- B. Static checkers.
- C. Interpreters
- D. Pretty pointers.

ANSWER: B

75. Which one of the following is not an element in the tokens given?

- A. The plus sign.
- B. The identifier position.
- C. The identifier initial.
- D. The condition as such.

ANSWER: D

76. The first and second basic rules of the hierarchical structure of a program are _____.

- A. any identifier or number is an expression.
- B. defining expressions in terms of operators and non recursive.
- C. if identifier1:=exp2.
- D. while (exp1) do statement2 if (exp1) then statement2.

ANSWER: A

77. A _____ is a data structure containing a record for each identifier, with fields for the

attribute for the identifier.

- A. table.
- B. open file table.
- C. symbol table.
- D. lexical analyzer.

ANSWER: C

78. The _____ and _____ phases usually handle a large fraction of the errors detectable by the compilers.

- A. syntax ,semantics.
- B. semantics ,lexical.
- C. error handler ,code generator.
- D. code generator ,semantics.

ANSWER: A

79. A leaf is a _____ with two or more fields

- A. column.
- B. record
- C. token
- D. code

ANSWER: B

80. The intermediate representation has a form called _____ which is like the assembly language for a machine in which every memory location can act like a register.

- A. two address code.
- B. single address code.
- C. three address code.
- D. three tier instruction.

ANSWER: C

81. The final phase of the compiler is the generation of target code, consisting of replaceable _____.

- A. intermediate code.
- B. semantic code.
- C. machine code
- D. buffer code

ANSWER: C

82. _____ is a mnemonic version of machine code.

- A. Rational preprocessors.
- B. Parse code.
- C. Intermediate code.
- D. Assembly code

ANSWER: D

83. The program that performs the two functions of loading and link-editing is called as _____.

- A. loaders.
- B. linkers.
- C. link-editor.
- D. optimal code.

ANSWER: A

84. Which one of the following is not included in front end?

- A. Semantic
- B. Syntactic
- C. Lexical.
- D. Target machine.

ANSWER: D

85. _____ is a process of determining if a string of tokens can be generated by a grammar.

- A. Systemization.
- B. Parsing
- C. Translation
- D. Phasing

ANSWER: B

86. A grammar for a language often treats identifier as a _____.

- A. block of strings.
- B. reserved words
- C. keywords
- D. token

ANSWER: D

87. The lexical analyzer is the _____ part of the compiler.

- A. first
- B. second
- C. before last.
- D. last

ANSWER: A

88. The term _____ denotes any finite set of symbols.

- A. language
- B. characters
- C. strings
- D. alphabets

ANSWER: B

89. Each regular expression denotes a language_____.

- A. L.
- B. LA.
- C. S.
- D. P.

ANSWER: A

90. The technique used for speeding up the lexical analyzer is _____.

- A. schemes
- B. buffer
- C. semantics
- D. sentinels

ANSWER: D

91. A _____ is a sequence of characters in the source program that is matched by the pattern for a token.

- A. fields
- B. table
- C. lexeme
- D. pattern

ANSWER: C

92. The phases of the compilers are collected into a _____ and _____ ends.

- A. primary , front.
- B. front, back.
- C. first, last.
- D. primary, posterior.

ANSWER: B

93. Which are the analyses that are not grouped into one pass?

- A. Lexical
- B. Syntax
- C. Semantic
- D. Code optimizer

ANSWER: D

94. The output of the second pass is usually a _____ machine code.

- A. locatable.
- B. not relocatable.
- C. relocatable
- D. distinguishable

ANSWER: C

95. Pass is _____ an input file once.

- A. writing
- B. rewriting
- C. controlling
- D. reading

ANSWER: D

96. The symbol * represents _____.

- A. reading bit.
- B. bit to be written.
- C. locating bit.
- D. relocation bit.

ANSWER: D

97. A parse tree is called as _____ syntax tree.

- A. abstract
- B. concrete
- C. comparative
- D. replicating

ANSWER: B

98. When the last statement is a recursive call of the same procedure then it is _____.

- A. tail recursive.
- B. dual recursive.
- C. nested recursive
- D. error

ANSWER: A

99. Consider the statement `count = count + increment;` which is the token in the statement?

- A. Count, increment.
- B. Only count.
- C. ;
- D. +

ANSWER: A

100. The lexical analyzer and a parser form a _____ pair.

- A. consumer-producer.
- B. producer-consumer.
- C. middleman-consumer.
- D. consumer-middleman.

ANSWER: B

101. The symbol that denotes an empty string is _____.

- A. {Є}.
- B. E.
- C. {Ф}.
- D. {}.

ANSWER: A

102. If a concatenation is done for x and y strings the appending is done in the _____ form.

- A. y to x.
- B. x to y.
- C. xx.
- D. yy.

ANSWER: A

103. The unary operator _____ has the highest precedence in regular expression.

- A. # .
- B. &.
- C. *.
- D. !.

ANSWER: C

104. If two regular expressions r & s denote the same language then r and s are_____.

- A. equal
- B. equivalent
- C. statement with error
- D. regular

ANSWER: B

105. The procedure _____ has access to the buffer.

- A. `install_id()`.
- B. `gettoken()`.

- C. start ().
- D. other digit().

ANSWER: A

106. In the compiler model, the parser obtains a string of tokens from the _____ .

- A. rest of front end.
- B. symbol table.
- C. lexical analyzer.
- D. source analyzer.

ANSWER: C

107. The output from the parser is usually _____ .

- A. symbol table
- B. string.
- C. source programs.
- D. parse tree.

ANSWER: D

108. Misspelling an identifier is an example for _____ error.

- A. lexical
- B. syntactic
- C. semantic
- D. logical

ANSWER: A

109. If an operator is applied to an incompatible operand then it is an example for _____ error.

- A. lexical
- B. syntactic
- C. semantic
- D. logical

ANSWER: C

110. Usually the error detection and recovery is a compiler is done in _____ phase.

- A. lexical
- B. syntax
- C. semantic
- D. logical

ANSWER: B

111. Performing local corrections on the remaining input such as replacing a comma by semicolon is _____ .

- A. panic mode.
- B. phrase level.
- C. error production.
- D. global correction.

ANSWER: B

112. A set of string is an example for _____ .

- A. terminals.
- B. non terminals.
- C. productions.

D. start symbol.

ANSWER: B

113. The manner in which the terminals and non terminals can be combined to form strings deals with _____ of a grammar.

- A. start symbol.
- B. forward reference.
- C. backward reference.
- D. production

ANSWER: D

114. A language that can be generated by a grammar is said to be _____ language.

- A. text-free
- B. context-free
- C. forward
- D. backward

ANSWER: B

115. Rightmost derivations are also called as _____ derivations.

- A. conical
- B. canonical
- C. triangular.
- D. depth first

ANSWER: B

116. Which of the following is not a type of derivation?

- A. Leftmost
- B. Rightmost
- C. Canonical
- D. Triangular

ANSWER: D

117. An efficient non-backtracking form of top-down parser is called _____ .

- A. unpredeictive parser.
- B. predictive parser.
- C. ambiguous parser.
- D. semantic parser.

ANSWER: B

118. _____ parsing attempts to construct a parse tree for an input string from the leaves to the root.

- A. Top down.
- B. Target program.
- C. Bottom up.
- D. Source program.

ANSWER: C

119. The string appearing to the right of a handle contains only _____ symbols.

- A. non-terminal
- B. production
- C. start
- D. terminal

ANSWER: D

120. In _____ action of a shift reducer parser, the parser announces successful completion of parsing.
- A. shift
 - B. reduce
 - C. accept
 - D. error

ANSWER: C

121. The set of prefixes of right sentential forms that can appear on the stack of a shift reduce parse are called _____ prefixes.
- A. variable
 - B. viable
 - C. static
 - D. dynamic

ANSWER: B

122. In the synthesis analysis model of a compiler, the front end translates a source program into an _____ code for the generation of target program.
- A. intermediate
 - B. machine
 - C. object
 - D. lexical

ANSWER: A

123. Syntax tree, postfix notations and three-address codes are example for _____ language.
- A. source language.
 - B. machine
 - C. compiler
 - D. intermediate

ANSWER: D

124. A linear representation of a syntax tree where it lists the nodes of a tree in which a node appears immediately after its children is _____ notation.
- A. prefix
 - B. postfix
 - C. infix
 - D. outfix

ANSWER: B

125. Quadruples, triples and indirect triples are representation of _____ .
- A. three-address statements.
 - B. four-address statements.
 - C. two-address statements.
 - D. triple-address statements.

ANSWER: A

126. The op field in quadruples contains _____ for the operator.
- A. external code.
 - B. arguments
 - C. internal code.

D. function
ANSWER: C

127. Translating a Boolean expression into three address code without generating code for any of the Boolean operators and without hairing the code is _____ code.

- A. short circuit
- B. breaking
- C. choosing
- D. omitting

ANSWER: A

128. op,result,arg1 and arg2 are fields of _____.

- A. triples.
- B. indirect triples.
- C. quadruples.
- D. three address code.

ANSWER: C

129. $\text{expr} ::= (\text{expr}) \text{Expr} ::= -\text{expr} \text{Expr} ::= \text{id}$ In the above grammar the terminal symbols are

- A. -and ()
- B. -and==
- C. () and==
- D. id and expr.

ANSWER: A

130. X,Y and Z represent _____ symbols which are either terminals or non terminals.

- A. tokens
- B. identifiers
- C. string of terminals.
- D. grammar

ANSWER: D

131. Usually the _____ side of the first production is the start symbol.

- A. right
- B. left
- C. right and left.
- D. center

ANSWER: B

132. _____ specifies the manner in which the terminals and non terminals can be combined to form strings.

- A. Start symbol
- B. Token
- C. Identifiers
- D. Production

ANSWER: D

133. The language defined by the grammar is denoted by a set of strings called _____ .

- A. start symbol.
- B. token
- C. identifiers

D. production

ANSWER: A

134. An infinitively recursive call is an example for _____ error.

- A. lexical
- B. syntactic
- C. semantic
- D. logical

ANSWER: D

135. +-> denotes _____

- A. implies in one or more steps.
- B. implies in zero or more steps.
- C. derives in one or more steps.
- D. derives in zero or more steps.

ANSWER: C

136. Algorithms deal with _____ type of error recovery strategy.

- A. panic mode.
- B. phrase level.
- C. error productions.
- D. global correction.

ANSWER: D

137. Production is treated as a rewriting rule in which the non terminal on the left side is replaced by the string on the right side of the production in _____ .

- A. synchronization
- B. passing
- C. definitions
- D. derivations

ANSWER: D

138. Elimination or replacement of unnecessary instruction in object code is called _____ .

- A. code replacement
- B. code elimination
- C. code improvement
- D. code optimization

ANSWER: B

139. Optimization technique based on what happens across a basic block is called _____ optimization.

- A. local code
- B. basic code
- C. global code
- D. lock code

ANSWER: C

140. Global optimizations are based on analysis.

- A. code flow
- B. data flow
- C. instruction flow
- D. program flow.

ANSWER: B

141. A computer optimization must preserve the of the program.

- A. semantics
- B. syntax
- C. codes
- D. locks

ANSWER: A

142. Data redundancy is sometimes available at the level.

- A. design
- B. target
- C. source
- D. encoding

ANSWER: C

143. By using a to eliminate the redundancies, the program becomes efficient.

- A. compiler
- B. optimizer
- C. eliminator
- D. analyzer

ANSWER: A

144. The transformation of replacing an expensive operation with a cheaper one is called _____.

- A. operation reduction.
- B. operation replacement
- C. strength reduction
- D. size reduction.

ANSWER: C

145. Code optimization & generation phases are often referred as _____.

- A. design phase.
- B. back end
- C. optimization phase
- D. front end.

ANSWER: C

146. Many code transformations depend upon the _____ of loops in a flow graph.

- A. conditions
- B. jump
- C. execution
- D. identification

ANSWER: C

147. Local optimization enables improvement in _____ of code.

- A. efficiency
- B. running time
- C. execution time
- D. compile time

ANSWER: C

148. _____ is the process of determining if a string of tokens can be generated by a grammar

- A. Parsing
- B. Compiling
- C. Analyzing
- D. Translating

ANSWER: A

149. A _____ is a binding performed after the execution of a program has begun.

- A. dynamic
- B. static
- C. idle
- D. exception

ANSWER: A

150. Data structures in language processing can be classified based on _____.

- A. nature
- B. purpose
- C. lifetime
- D. all the above

ANSWER: D

Staff Name
Suresh Kumar M.