CS606-Compiler Construction

MID TERM MCQS

Prepared by: JUNAID MALIK

Question No 1

LL(1) parsing is called ___________________ parsing.

- **Predictive**
- Non-predictive
- Post-predictive
- Post determine

Question No 2

One kind of predictive parser is the ______________________.

- **Recursive descent parses**
- Bottom-up parser
- Natural parser
- Multilingual parser

Question No 3

Front-end of a two pass compiler consists of ____________________.

- Code optimizer
- **Scanner**
- Parser
- Assembler

Question No 4

The algorithm for _____________________ conversion is called Thompson's Construction.

- DFA to RE
- **RE to DFA**
- RE to NFA
- CFG to NFA

Question No 5

_________ convert the relocatable machine into absolute machine code by linking library and relocatable object files.

- Assembler
- **Loader/link editor**
- Compiler
- Preprocessor

Question No 6

The ____________________ parser uses a stack to hold the frontier.

- Top-down
- **Bottom-up**
- Predicative
- LL(1) parsers

Question No 7

________________ is the process of discovering a derivation for some sentence of a language.

- Compiling
- Scanning
- **Parsing**
- Interpretation

Question No 8

A ________________ accepts a string if we can follow transitions labeled with characters in the string from start state to some accepting state.

- **Finite automata…………………(For More Visit)**
- Regular expression
- Regular language
- Acceptor

Question No 9

The general idea behind NFA-to-DFA construction is that each _______ state corresponds to a set of _______ states.

- **DFA, NFA**
- NFA, DFA
- NFA, FA
- CFG, NFA

Question No 10

Optimal registers allocation is ____________ problem.

- NP-hard
- **NP-complete**
- Polynomial complete
- Multinomial complete

Question No 11

Majority of the algorithms employed in the back-end are _______________________.

- Polynomial time
- NP-Complete
- **Np**
- None of the given

Question No 12

In predictive parsing table, the rows are ________________________.

- **Non-terminals**
- Terminals
- Both non-terminals and terminals
- Neither non-terminals nor terminals

Question No 13

Predictive parser accepts _____________ grammars.

- LL(n+)
- LL(j-)
- **LL(k)**
- LL(K++)

Question No 14

________ and __________ actions are used by bottom-up parsing.

- Break and reduce
- Shift and break
- **Shift and reduce**
- Shift and expand

Question No 15

If the right-hand side of a production has k symbols, it has _________ placeholder positions.

- k
- **k+1**
- k+2
- k+3

Question No 16

Lexical analyzer function is not ___________________.

- Removing white space
- Removing constants, identifies and keywords
- **Removing comments**
- Register allocation

Question No 17

Reduce action __________ zero or more symbols from the stack.

- Pushes
- Pops
- **Both pushes and pops**
- Adds

Question No 18

Parser determines if the input is ____________ well formed.

- Programmatically
- Logically
- **Syntactically**
- Mechanically

Question No 19

Grammar with the ______________ property is called predictive grammar.

- LL(3)
- LL(2)
- **LL(1)**
- LL(n+)

Question No 20

The executable code typically runs as a _____________ in an Operating System Environment.

- **Process**
- Threads
- Routine
- Function

Question No 21

In a transition table cells of the table contain the _________ state.

- Reject State
- **Next State**
- Previous State
- Same State

Question No 22

In parser the two LL stand(s) for ____________________.

- Left-to-right scan of input
- Left-most derivation
- **Left-to-right scan of input and left-most derivation**
- Right to left scan of input

Question No 23

The _______________ builds intermediate Representation (IR) for source program.

- Scanner
- **Parser**
- Compiler
- Loader

Question No 24

Left factoring of a grammer is done to save the parser from _________________.

- Location tracking
- **Front tracking**
- Back tracking
- Random tracking

Question No 25

In Flex specification file different sections are separated by ______________.

- **%%**
- &&
- ##
- \\

Question No 26

Left factoring of a grammar is done to save the parser from __________.

- Location tracking
- Front tracking
- **Back tracking**
- Random tracking

Question No 27

Lexer and scanner are _________ phase of compiler.

- **Same**
- Different
- Backend
- Middle end

Question No 28

Back End of two-pass compiler uses ____________ algorithm.

- O(n)
- (n log n)
- **NP Complete**
- O(n2)

Question No 29

The ____________ contains a lot of unneeded information.

- **Parse tree**

- Abstract syntax tree
- Complete tree
- Abstract syntax tree

Question No 30

A _______________ start at the leaves and grows toward root of the parse tree.

- Top-down parser
- **Bottom-up parser**
- Vertical parser
- Horizontal parser

Question No 31

Typical compilation means programs written in high-level languages to low-level ________.

- **Object code PG # 06**
- Byted code
- Unicode
- Object Code and byte code

Question No 32

In compilation process, Hierarchical analysis is also called _____________.

- Parsing
- **Syntax**
- Parsing and Syntax analysis o
- None of given

Question No 33

IR (Intermediate Representation) stores the value of its operand in ______________.

- **Registers PG # 10**
- Memory
- Hard disk
- None Of given

Question No 34

A lexeme is a sequence of characters in the source program that is matched by the pattern for a ___________.

- Linker
- **Token**
- Control flow
- None of given

Question No 35

Parsers take ____________ as Input from lexical analyzer.

- Linker
- **Token**
- Instructions
- None of the given

Question No 36

What kind of abstract machine can recognize strings in a regular set?

- **DFA**
- NFA
- PDA
- None of the given

Question No 37

In multi pass compiler, during the first pass it gathers information about ____________.

- **Declaration**
- Bindings
- Static information
- None of the given

Question No 38

In DFA minimization, we construct one ____________ for each group of states from the initial DFA.

- **State PG # 30**
- NFA
- PDA
- None of given

Question No 39

______(Lexical Analyzer generator), is written in Java..

- Flex
- **Jlex PG # 31**
- Complex
- None of given

Question No 40

Recursive ________ parsing is done for LL(I) grammar.

- **Decent**
- Ascent
- Forward
- None of the given

Question No 41

Alternative of the backtrack in parser is Look ahead symbol in _________.

- **Input**
- Output
- Input and Output
- None of the given

Question No 42

Parser takes tokens from scanner and tries to generate ___________.

- Binary Search tree
- **Parse tree**
- Binary Search tree and Parse tree
- None of the given

Question No 43

In predictive parsing table, the rows represents __________.

- Terminals
- Both non-terminal and terminals
- **Non-terminals PG # 62**
- None of the given

Question No 44

In LL(I) parsing algorithm,________ contains a sequence of grammar symbols.

- **Stack PG # 62**
- Link List
- Array
- None of the given

Question No 45

Consider the grammar

A → B C D

B → h B | £

C → C g | g | C h | i

D → AB | £

First of A is __________.

- **h, g, i**
- g
- h
- None of the given

Question No 46

Consider the grammar

A → B C D

B → h B | £

C → C g | g | C h | i

D → AB | £

First of C is ___________.

- **g, i**
- g
- h, i
- None of the given

Question No 47

Bottom-up parsing uses only __________ kinds of actions.

- **Two PG # 71**
- Three
- Four

- Five

Question No 48

The shift action _________ a terminal on the stack.

- **Pushes PG # 73**
- Pops
- Both push and pops
- None of the given

Question No 49

Reduce ________ action zero or more symbols from the stack.

- Pushes
- **Pops PG # 73**
- Both push and pops
- None of the given

Question No 50

In compilers, linear analysis is also called _______.

- Lexical analysis
- Scanning
- **Lexical analysis and scanning**
- None of the given

Question No 51

Back End of two-pass compiler uses _______ algorithm.

- O(n)
- O(n log n)
- **NP Complete**
- None of the Given

Question No 52

The Back End of a compiler consist of _________.

- **Instruction selection**
- Register allocation
- Instruction scheduling
- All of the given

Question No 53

In Back End module of compiler, optimal register allocation uses ___________.

- O(log n)
- O(n log n)
- **NP-Complete**
- None of the given

Question No 54

.___________is a regular expression for the set of all strings over the alphabet {a} that has an even number of a' s.

- **(aa)***
- (aa)*a
- aa*a
- a*(aa)

Question No 55

_________algorithm is used in DFA minimization.

- James's
- Robert’s
- **Hopcroft’s PG # 25**
- None of given

Question No 56

____________ is an important component of semantic analysis.

- Code checking
- **Type checking PG # 39**
- Flush checking
- None of the given

Question No 57

Consider the grammar

A → B C D

B → h B | £

C → C g | g | C h | i

First of B is_____________.

- h, i
- **h, £**
- g
- None of the given

Question No 58

Consider the grammar,

A → B C D

B → h B | £

C → C g | g | C h | i

D → AB | £

First of D is_____________.

- h, g
- h
- **h, g , i, £**

- None of the given

Question No 59

__________ Parsers never shifts into an error state.

- LS
- LT
- **LR**
- LP

Question No 60

In_________ , certain checks are performed to ensure that components of a program fit together meaningfully..

- Linear analysis
- Hierarchical analysis
- **Semantic analysis**
- None of given

Question No 61

___________ read the input character and produce sequence of tokens as output.

- **Lexical analyzer**
- Parser
- Symbol table
- None of the given

Question No 62

. The regular expression _________denotes, the set of all strings of a's and b’s of length two.

- a*
- (a* | b*)*

- (a*b*)*
- **(a|b)(a|b)**

Question No 63

__________ of a two-pass compiler is consists of Instruction selection, Register allocation and Instruction scheduling.

- **Backend PG # 9**
- Frontend
- Start
- None of the Given

Question No 64

___________ is evaluated to yield a value.

- Command
- **Expression**
- Declaration
- None of the given

Question No 65

Bottom-up parsers handle a ____________ class of grammar.

- **Large PG # 49**
- Small
- Medium
- None of the given

Question No 66

The input specification file to flax consists of-------- section.

- 1
- 2

- **3**
- 4

Question No 67

Lexer and scanner are two different phases of compiler.

- True
- **False**

Question No 68

In compiler linear analysis is also called.

- Lexical analysis
- Scanning
- **Both lexical analysis and scanning**
- None of the given

Question No 69

Which of the following statements is NOT true?

- The language accepted by finite automate are the languages denoted by regular expression.
- For every DFA there is a regular expression denoting its languages.
- **For a regular expression r, there does not exit NFA with L(r) any transit that accept.**
- None of the given.

Question No 70

Top – down parsing expands a ----------- from the start symbol to the leaves.

- Parse tree
- Abstract Syntax tree
- Parse tree
- **All of the given**

Question No 71

A top-down purser starts with the ------------ of the parse tree.

- **Root**
- Leaf
- Middle
- None of the given

Question No 72

Can a DFA simulate NFA?

- NO
- **YES**
- Sometimes
- Depends on NFA

Question No 73

Bottom- up parsing is also called------------------.

- **LR parsing**
- LT parsing
- LS parsing
- SS parsing

Question No 74

The ----------- is optimized for hardware it is to run on.

- C++ code
- C code
- **Assembly code**
- None of the given

Question No 75

Abstract syntax tree summarizes ------------ without the details of derivation.

- Theory
- **Grammatical structure**
- Data
- None of the give

Question No 76

LR parsing---------- a string to the start symbol by inverting production.

- **Reduces**
- Shifts
- Adds
- None of the given

Question No 77

Alternative to backtrack in top- down parser is --------------.

- Context free grammar
- Trees
- **Look ahead**
- None of the given

Question No 78

Typical compilation means programs written in high- level languages to low-languages--------------.

- **Object code**
- Byte Code
- Unicode
- Both object code and Byte code

Question No 79

The ------------------- Checks the stream of work (token) and their parts of speech for grammatical correctness.

- Parser
- **Scanner**
- **Compiler**
- None of the given

Question No 80

A grammar must be ---------------- before use for predicative parsing.

- Right factored
- **Left factored**
- Factored
- None of the given

Question No 80

Parser does not distinguish between valid and invalid sequences of token.

- True
- **False**

Question No 81

In Compilation process Hierarchical analysis is also called:

- Parsing
- **Syntax analysis**
- Both parsing and Syntax analysis
- None of the given

Question No 82

The pair <role, word> is given the name-------------.

- Word
- **Token**
- Syntax
- None of the given

Question No 83

The back end maps intermediate Representation (IR) into target--------------.

- Object code
- Machine code
- **Source code**
- Linker

Question No 84

------------- is evaluated to yield a value.

- Command
- **Expression**
- Declaration
- Declaration and Command

Question No 85

Front- end of a two pass compiler is consists of scanner.

- **TRUE**
- FALSE

Question No 86

The ------------ returns a sequence of mating token at output (or an error) and it always return the longest matching token.

- Scanner
- Lexer
- **Lexical analyzer**

- All of the given

Question No 87

--------------- avoid hardware stalls and interlocks.

- Register allocation
- **Instruction scheduling**
- Instruction selection
- None of given

Question No 88

LR parsers can handle _________________ grammars.

- Left-recursive (Page 63)
- file-recursive
- End-recursive
- Start-recursive

Question No 89

_____ convert the relocatable machine code into absolute machine code by linking library and relocatable object files.

- Assembler
- Loader/link-editor
- Compiler
- Preprocessor

Question No 90

Follow of C is ______________ .

- g, h, i, $
- g, h, $
- h, i, $

- h, g, $

Question No 91

Consider the grammar A --> B C D

B --> h B | epsilon

C --> C g | g | C h | i

D --> A B | epsilon

Follow of B is _____________.

- **h**
- g, h, i, $
- g, i
- g

Question No 92

An important component of semantic analysis is __________________.

- Code checking
- **type checking (Page 33)**
- flush checking
- None of the given

Question No 93

In PASCAL ___________ represent the inequality test.

- :=
- =
- <>
- None of the given

Question No 94

Lexical Analyzer generator ________________ is written in Java.

- Flex
- **Jlex (Page 26)**
- Complex
- None of given

Question No 95

____________avoid hardware stalls and interlocks.

- Register allocation
- **Instruction scheduling (Page 121)**
- Instruction selection
- None of given

Question No 96

Consider the following grammar,

A --> B C D

B --> h B | epsilon

C --> C g | g | C h | i

D --> A B | epsilon

First of A is _______________.

- h, g, i
- g
- **h**
- None of the given

Question No 97

Recursive _____________ parsing is done for LL(1) grammar.

- **Descent (Page 47)**
- Ascent
- Forward
- Backward

Question No 98

One of the core tasks of compiler is to generate fast and compact executable code.

- **True**
- False

Question No 99

Left factoring of a grammar is done to save the parser from back tracking.

- **True**
- False

Question No 100

Responsibility of _______________ is to produce fast and compact code.

- **Instruction selection (Page 9)**
- Register allocation
- Instruction scheduling
- None of given

Question No 101

_________________ algorithm is used in DFA minimization.

- James's
- Robert's
- **Hopcroft's (Page 25)**
- None of given

Question No 102

Compilers are sometimes classified as.

- Single pass
- Multi pass
- Load and go
- **All of the given**

Question No 103

In multi pass compiler during the first pass it gathers information about ____________________.

- **Declaration**
- Bindings
- Static information
- None of the given

Question No 104

Flex is an automated tool that is used to get the minimized DFA (scanner).

- True
- **False (Page 26)**

Question No 105

In compilation process Hierarchical analysis is also called

- Parsing

- Syntax analysis
- **Both Parsing and Syntax analysis**
- None of given

Question No 106

For each language to make LL(1) grammar, we take two steps, 1st is removing left recurrence and 2nd is applying left factoring in sequence.

- **True**
- False

Question No 107

______________is evaluated to yield a value.

- Command
- **Expression**
- Declaration
- Declaration and Command

Question No 108

Optimal registers allocation is an NP-hard problem.

- True
- **False**

Question No 109

Parser takes tokens from scanner and tries to generate ________________ .

- Binary Search tree
- **Parse tree**
- Syntax trace
- None of the given

Question No 110

Front end of two pass compiler takes_________________ as input.

- **Source code (Page 5)**
- Intermediate Representation (IR)
- Machine Code
- None of the Given

Question No 111

In in we construct one _______________ for each group of states from the initial DFA.

- **State (Page 25)**
- NFA
- PDA
- None of given

Question No 112

In Three-pass compiler _____________ is used for code improvement or optimization.

- Front End
- **Middle End (Page 10)**
- Back End
- Both Front end and Back end

Question No 113

________________ of a two-pass compiler is consists of Instruction selection, Register allocation and Instruction scheduling.

- **Back end (Page 10)**
- Front end
- Start
- None of given

Question No 114

NFA is easy to implement as compared to DFA.

- True
- **False (Page 19)**

Question No 115

We can get an LL(1) grammar by ________________ .

- Removing left recurrence
- Applying left factoring
- **Removing left recurrence and Applying left factoring**
- None of the given

Question No 116

Parser always gives a tree like structure as output

- True
- False

Question No 117

In Back End module of compiler, optimal register allocation uses________________ .

- O(log n)
- O(n log n)
- **N P-Complete (Page 10)**
- None of the given

Question No 118

Lexer and scanner are two different phases of compiler
Select correct option:

- True
- **False**

Question No 119

In Flex specification file different sections are separated by ____________________.

- **%% (Page 26)**
- &&
- ##
- \\

Question No 120

_____________________ phase which supports macro substitution and conditional compilation.

- Semantic
- Syntax
- **Preprocessing** Click here for detail
- None of give

Question No 121

_____________tree in which each node represents an operator and children of the node represent the operands.

- **Abstract syntax (Page 100)**
- Concrete syntax
- Parse
- None of the given

Question No 122

In _____certain checks are performed to ensure that components of a program fit together meaningfully.

- Linear analysis
- Hierarchical analysis
- **Semantic analysis (Page 33)**
- None of given

Question No 123

In compiler linear analysis is also called.

Select correct option:

- Lexical analysis
- Scanning
- **Both lexical analysis and scanning click here for details**
- None of the given

Question No 124

Which of the following statement is true about Two pass compiler.

- Front End depends upon Back End
- **Back End depends upon Frond End (Page 5)**
- Both are independent of each other
- None of the given

Question No 125

Ambiguity can easily be handled by Top-down Parser

- **True click here for detail**
- False

Question No 126

Alternative of the backtrack in parser is Look ahead symbol in _________ .

- **Input (Page 46)**
- Output
- Input and Output
- None of the given

Question No 127

Typical compilation means programs written in high-level languages to low-level ___________.

- Object code
- Byte code
- Unicode
- **Both Object Code and byte code (Page 2)**

Question No 128

LR parsing _______________ a string to the start symbol by inverting.

- **Reduces (Page 63)**
- Shifts
- Adds
- None of the given

Question No 129

In parser the two LL stand(s) for _____________.

- Left-to-right scan of input
- left-most derivation
- **All of the given click here for detail**
- **None of given**

Question No 130

Bottom-up parsers handle a ________________ class of grammars.

- **Large (Page 63)**
- Small
- Medium
- None of the given

Question No 131

Compilers are some times classified as.

- Single pass
- Multi pass
- Load and go

- **All of the given**

Question No 132

The Back End of a compiler consist of ______________ .

- **Instruction selection** (Page 10)
- Register allocation
- Instruction scheduling
- All of the given

Question No 133

Can a DFA simulate NFA?

- **Yes** click here for detail
- No
- Sometimes
- Depend upon nfa

Question No 134

Which of the statement is true about Regular Languages?

- Regular Languages are the most popular for specifying tokens.
- Regular Languages are based on simple and useful theory.
- Regular Languages are easy to understand.
- **All of the given (Page 15)**

Question no 135

In a transition table cells of the table contain the ________ state.

- Reject state
- **Next state (Page 18)**
- Previous state
- None of the given

Question No 136

The transition graph for an NFA that recognizes the language (a | b)*abb will have following set of states.

- {0}
- {0,1}
- {0,1,2}
- **{0,1,2,3} (page 23) Not sure**

Question No 137

Functions of Lexical analyzer are?

- Removing white space
- Removing constants, identifiers and keywords
- Removing comments
- **All of the given** click here for detail

Question No 138

________ method is known as Subset Construction Method.

- **NFA TO DFA (Page 19)**
- DFA
- DFA maximization
- None of given

Question No 139

Which one is NOT the Role of Run-Time System

- garbage collection
- memory management
- run time error checking
- **none of given (Page 11)**

Question No 140

Compiler_________ information from one representation to another

- modified
- **translate (Page 2)**

- execute
- extract

Question No 141

Front-end of a two pass compiler is consists of Scanner.

- True
- **False (Page 12)**

Question No 142

Bottom-up parsing uses only ________________ kinds of actions.

- **Two (Page 64)**
- Three
- Four
- Five

Question No 143

Bottom-up parsing is also called ________________ .

- **LR parsing (Page 63)**
- LT parsing
- LS parsing
- SS parsing

Question No 144

Backtracking is a costly operation which is caused due to Left Recursion.

- True
- **False**

Question No 145

A non-recursive predictive parser is also called _________________.

- **table-driven parser (Page 54)**

- Abstract parser
- Conceptual parser
- None of the given

Question No 146

AST summarizes the grammatical structure with the details of derivations.

- True
- **False (Page 8)**

Question No 147

Backtrack is term associated with Top-down parsing.

- **True (Page 42)**
- False

Which of the following statement is true about Two-pass compiler.

- Front End depends upon Back End
- Back End depends upon Front End
- Both are independent of each other
- None

2. In a three-pass compiler, ___ is used for code improvement or optimization.

- Front End
- Middle End
- Back End
- Both Front End and Back End

3. In multi-pass compiler during the first pass, it gathers information about ___

- Declaration
- Bindings
- Static information

- None

4. Which of the statement is true about Regular Languages?

- Regular Languages are the most popular for specifying tokens.
- Regular Languages are based on simple and useful theory.
- Regular Languages are easy to understand
- All of the given

5. The DFA uses its state to keep track of ___ the NFA can be in after reading each input symbol.

- Accept state
- Reject state
- Next state
- All possible states

6. In ___ certain check are performed to ensure that components of a program fit together meaningfully.

- Linear analysis
- Hierarchical analysis
- Semantic analysis
- None

7. Compilers are some times classifies as:

- Single-pass
- Multipass
- Load and go
- All of the given

8. In compiler, linear analysis is also called

- Lexical analysis
- Scanning
- Both lexical analysis and scanning
- None

9. In a transition table cells of the table contain the ___ state.

- Reject state
- Next state
- Previous state
- None

10. In Back End module of the compiler, optimal register allocation uses ___.

- O(log n)
- O(n log n)
- NP-Complete
- None

11. A non-recursive predictive parser is also called ___

- table-driven parser
- Abstract parser
- Conceptual parser
- None

12. The shift action ___ a terminal on the stack

- Pushes
- Pops
- Both pushes and pops
- None

13. Bottom-up parsing is also called ___

- LR parsing
- LT parsing
- LS parsing
- SS parsing

14. ___ of a two-pass compiler is consists of instruction selection, Register allocation and instrcution scheduling

- Backend
- Frontend
- Start

- None

15. Consider the grammar
A -> B C C
B -> h B | epsilon
C -> C g | g | C h | i
D -> A B | epsilon
First of A is ___

- h, g, i
- g
- h
- None

16. A -> B C C
B -> h B | epsilon
C -> C g | g | C h | i

D -> A B | epsilon
Follow of A ___

- h, $
- $
- i
- g

17. A -> B C C
B -> h B | epsilon
C -> C g | g | C h | i
D -> A B | epsilon
Follow of C is ___

- g, h, i, $
- g, h, $
- h, i, $
- h, g, $

18. Left factoring of grammar is done to save the parser from backtracking.

- True
- False

19. Typical compilation means programs written in high-level languages to low-level ___

- Object code
- Byte code
- Unicode
- Object code and byte code

20. In compilation process, Hierarchical analysis is also called ___

- Parsing
- Syntax
- Parsing and Syntax analysis
- None

21. IR (Intermediate Representation) stores the value of its operand in ___

- Registers
- Memory
- Hard disk
- None

22. A lexeme is a sequence of characters in the source program that is matched by the pattern for a ___

- Linker
- Token
- Control flow
- None

23. Parsers take ___ as Input from lexical analyzer.

- Linker
- Token
- Instructions
- None

24. What kind of abstract machine can recognize strings in a regular set?

- DFA
- NFA
- PDA
- None

25. In DFA minimization, we construct one ___ for each group of states from the initial DFA.

- State
- NFA
- PDA
- None

26. ___ (lexical Analyzer generator), is written in Java

- Flex
- Jlex
- Complex
- None

27. In flex specification file, different sections are separated by ___

- %%
- &&
- ##
- None

28. Recursive ___ parsing is done for LL(I) grammar.

- Descent y
- Ascent
- Forward
- None

29. Alternative of the backtrack in parser is Look ahead symbol in ___

- Input
- Output
- Input and Output
- None

30. Parser takes tokens from scanner and tries to generate ___

- Binary Search Tree
- Parse tree
- Binary Search and Parse tree
- None

31. In predictive parsing table, the rows represents ___

- Terminals
- Both non-terminals and terminals

- Non-terminals
- None

32. In LL(I) parsing algorithm, ___ contains a sequence of grammar symbols.

- Stack
- Link list
- Array
- None

33. Consider the grammar

A -> B C D
B -> h B | epsilon
C -> C g | g | C h | i
D -> A B | epsilon
First of C is ___

- g, i
- g
- h, i
- None

34. Bottom-up parsing uses only ___ kinds of actions.

- Two
- Three
- Four
- Five

35. Reduce ___ action zero or more symbols from the stack.

- Pushes
- Pops
- Both push and pops
- None

36. Back End of two-pass compiler uses ___ algorithm

- O(n)
- O(n log n)
- NP complete
- None

37. The Back End of a compiler consists of ___

- Instruction selection
- Register allocation

- Instruction scheduling
- All of given

38. ___ is a regular expression for the set of all strings over the alphabet {a} that has an even number of a’s

- aa*
- (aa)*
- aa*a
- a(aa)*

39. ___ algorithm is used in DFA minimization.

- Jame’s
- Robert’s
- Hopcroft’s y
- None

40. ___ is an important component of semantic analysis.

- Code checking
- Type checking
- Flush checking
- None

41. Consider the grammar

A – > B C D
B -> h B | epsilon
C -> C g | g | C h | i
First of B is ___

- h, i
- h, epsilon
- g
- None

42. Consider the grammar

A -> B C D
B -> h B | epsilon
C -> C g | g | C h | i
D -> A B | epsilon
First of D is ___

- h, g
- h

- h, g, i, epsilon
- None

43. ___ parsers never shifts into an error state.

- LS
- LT
- LR
- LP

44. ___ read the input character and produce sequence of tokens as output.

- Lexical analyzer
- Parser
- Symbol table
- None

45. The regular expression ___ denotes, the set of all strings of a's and b's of length two.

- a*
- (a*| b*)*
- (a*b*)*
- (a|b)(a|b)

46. ___ is evaluated to yield a value.

- Command
- Expression
- Declaration
- None

47. Bottom-up parsers handle a ___ class of grammar

- Large
- Small
- Medium
- None

48. LR parsers can handle ___ grammars.

- Left-recursive
- File-recursive
- End-recursive
- Start-recursive

49. ___ convert the relocatable machine code into absolute machine code by linking library and relocatable object files.

- Assembler
- Loader/link-editor
- Compiler
- Preprocessor

50. Consider the grammar
A -> B C D
B -> h B | epsilon
C -> C g | g | C h | i
D -> A B | epsilon
Follow of B is ___

- h
- g, h, i, $
- g, i
- g

51. In PASCAL ___represent the inequality test.

- :=
- =
- <>
- None

52. ___ avoid hardware stalls and interlocks.

- Register allocation
- Instruction scheduling
- Instruction selection
- None

53. One of the core tasks of compiler is to generate fast and compact executable code.

- True
- False

54. Responsibility of ___ is to produce fast and compact code.

- Instruction selection
- Register allocation
- Instruction scheduling
- None

55. Compilers are sometimes classified as

- Single pass

- Multi pass
- Load and go
- All of given

56. Flex is an automated tool that is used to get the minimized DFA (scanner).

- True
- False

57. For each language to make LL(1) grammar, we take two steps, 1st is removing left recurrence and 2nd is applying left factoring in sequence.

- True
- False

58. Optimal registers allocation is an NP-hard problem.

- True
- False

59. Front end of two pass compiler takes ___ as input.

- Source code
- Intermediate Representation (IR)
- Machine code
- None

60. NFA is easy to implement as compared to DFA

- True
- False

61. We can get an LL(1) grammar by ___

- Removing left recurrence
- Applying left factoring
- Removing left recurrence and Applying left factoring
- None

63. Lexer and scanner are two different phases of compiler.

- True
- False

64. ___ phase which supports macro substitution and conditional compilation.

- Semantic
- Syntax
- Preprocessing

- None

65. ___ tree in which each node represents an operator and children of the node represent the operands

- Abstract syntax
- Concrete syntax
- Parse
- None

66. Ambiguity can easily be handled by top-down parser

- True
- False

67. LR parsing ___ a string to the start symbol by inverting productions.

- Reduces
- Shifts
- Adds
- None

68. In parser the two LL stand(s) for ___

- Left-to-right scan of input
- left-most derivation
- All of the given
- None

69. Can a DFA simulate NFA?

- Yes
- No
- Sometimes
- Depend upon NFA

70. The transition graph for an NFA that recognized the language (a|b)*abb will have following set of states.

- {0}
- {0, 1}
- {0, 1, 2}
- {0, 1, 2, 3}

71. Functions of Lexical analyzer are?

- Removing white space
- Removing constants, identifiers and keywords

- Removing comments
- All of given

72. ___ method is known as subset construction method.

- NFA to DFA
- DFA
- DFA maximization
- None

73. Which one is NOT the Role of Run-Time System?

- garbage collection
- memory management
- run time error checking
- None

74. Compiler ___ information from one representation to another.

- modified
- translate
- execute
- extract

75. Front-end of a two pass compiler is consists of scanner.

- True
- False

76. Backtracking is a costly operation which is caused due to Left Recursion.

- True
- False

77. AST summarizes the grammatical structure with the details of derivations.

- True
- False

78. Backtrack is term associated with Top-down parsing.

- True
- False

AL-JUNAID TECH INSTITUTE