**COMPILER DESIGN**

**QUESTION BANK**

**UNIT – I**

**ONE MARK QUESTIONS**

1. A translator that takes as input a high level language program and translates into machine language in one step is known as
2. Compiler b) Interpreter c) Preprocessor d) Assembler
3. Which of the following is used for grouping of characters into tokens?

a)Parser b) Code optimization c) Code generator d) Lexical analyzer

3.What is the function of a loader?

1. The process of assigning load addresses to the various parts of the program and adjusting the code and data in the program to reflect the assigned addresses is called \_\_\_\_\_\_\_.

a) Assembly b) Parsing c) Relocation d) Symbol resolution

5. Define pattern.

1. The \_\_\_\_\_\_\_\_\_ phase converts an intermediate code into an optimized code that takes lesser space and lesser time to execute.

a) code optimization b) syntax directed translation

c) code generation d) intermediate code generation

7.\_\_\_\_\_\_\_\_\_\_\_\_\_\_is invoked whenever any fault occurs in the compilation process of source program

a) syntax analyzer b) code generator c)Error handler d) Lexical analyzer

8. What is Symbol table?

9. What is the role of Scanner?

10. List out cousins of the compiler.

11. Distinguish between compiler and interpreter.

12. Mention the issues in a lexical analyzer.

13. What is the role of Linker and Loaders?

14. What are the representation methods of Intermediate code?

15. What are the compiler construction tools?

16. S-> SS+/SS\*/a is ambiguous grammar or not for the string aa+a\*?

17. What is the Structure of LEX?

18.Define Pass?

19. what is the difference between pass and phase of a compiler?

20. On what basis we are dividing the phases of compiler?

**UNIT-II**

1. Define left factoring?

2.Define ambiguity?

3.Write different types of Bottom up parser?

4.What is backtracking?

5.Define left recursion. Give an example?

6.Write different types of top down parser?

7.What are the rules to remove ambiguity in grammar?

8. If we remove the left recursion or left factoring from a grammar, while it removes ambiguity of a grammar or not?

9.Define first() and follow() functions?

10. Remove the left recursion from the grammar A->A! (if not write down the reason?)

**TENMARK QUESTIONS**

1. What are various phases of compiler? Explain each phase in detail. Write down the output of each phase for expression a=b+c\*50.
2. a).Explain about Compiler construction tools?

b). What is the role of LEX? Explain the structure of LEX program with an example.

1. Explain the input buffer scheme for scanning the source program.

UNIT-II

1. Consider the grammar,

E -> TE’

E’->+TE’ | €

T -> FT’

T’->\*FT’ | €

F ->(E) | id.

Construct a predictive parsing table for the grammar given above. Verify whether the input string id + id \* id is accepted by the grammar or not.

2.Consider a Grammar G as follows :

S→W

W→ZXY/XY

Y→c/ϵ

Z→a/d

X→Xb/ϵ

Check it is LL(1) grammar or not? If yes draw the LL(1) Parsing Table.