Unit–IV

1. A run-time environment is responsible for:

    A) Lexical analysis

    B) Managing memory and execution support during program execution

    C) Parsing source code

    D) Generating tokens

    Answer: B

2. The memory area primarily used for function calls and local variables is:

    A) Heap

    B) Stack

    C) Code segment

    D) Data segment

    Answer: B

3. Stack allocation is most suitable for:

    A) Dynamically sized objects with arbitrary lifetimes

    B) Recursive procedure calls

    C) Global variables only

    D) Permanent storage

    Answer: B

4. Each procedure call creates a new:

    A) Symbol table

    B) Activation record

    C) Parse tree

    D) Basic block

    Answer: B

5. An activation record typically contains:

A) Tokens and lexemes

B) Local variables, parameters, and return address

C) Only object code

D) Parsing tables

Answer: B

6. Access links in activation records are used to:

A) Connect basic blocks

B) Access nonlocal data

C) Store machine instructions

D) Generate intermediate code

Answer: B

7. Nonlocal variables are variables that are:

A) Declared within the current procedure

B) Declared in an enclosing scope

C) Global constants

D) Temporary variables

Answer: B

8. The memory area used for dynamic memory allocation is:

A) Stack

B) Heap

C) Register file

D) Cache

Answer: B

9. Heap management deals with:

A) Parsing expressions

B) Allocation and deallocation of dynamic memory

C) Register allocation

D) Lexical analysis

Answer: B

10. Garbage collection is the process of:

A) Removing syntax errors

B) Reclaiming memory occupied by unreachable objects

C) Allocating stack space

D) Optimizing code

Answer: B

11. Which of the following is an automatic memory management technique?

A) Stack allocation

B) Heap allocation

C) Garbage collection

D) Register allocation

Answer: C

12. Trace-based garbage collection identifies:

A) Unused variables in source code

B) Reachable and unreachable objects

C) Parsing conflicts

D) Register usage

Answer: B

13. In trace-based collection, the collector starts tracing from:

A) Basic blocks

B) Root references

C) Parse trees

D) Symbol tables

Answer: B

14. Which memory allocation technique has minimal allocation overhead?

    A) Heap allocation

    B) Stack allocation

    C) Dynamic allocation

    D) Garbage collection

    Answer: B

15. The main objective of code generation is to:

    A) Produce efficient target code

    B) Perform lexical analysis

    C) Generate parse trees

    D) Build symbol tables

    Answer: A

16. The input to the code generator is usually:

    A) Source program

    B) Intermediate representation

    C) Object code

    D) Machine instructions

    Answer: B

17. The target language of a compiler is generally:

    A) Regular expressions

    B) Assembly language or machine code

    C) Context-free grammar

    D) Source code

    Answer: B

18. One important issue in code generator design is:

A) Token recognition

B) Instruction selection

C) Lexeme extraction

D) Input buffering

Answer: B

19. Target code addresses may refer to:

A) Registers, memory locations, and constants

B) Parse trees only

C) Tokens only

D) Grammar productions

Answer: A

20. A basic block is:

A) A sequence of instructions with a single entry and single exit

B) A collection of tokens

C) A set of grammar rules

D) A group of activation records

Answer: A

21. A flow graph represents:

A) Data types in a program

B) Control flow among basic blocks

C) Symbol table entries

D) Heap allocation

Answer: B

22. Nodes in a flow graph correspond to:

A) Variables

B) Basic blocks

C) Tokens

D) Registers

Answer: B

23. Optimization within a basic block is called:

A) Global optimization

B) Local optimization

C) Syntax optimization

D) Dynamic optimization

Answer: B

24. Which of the following is a common basic block optimization?

A) Common subexpression elimination

B) Token generation

C) Parsing

D) Buffer management

Answer: A

25. Peephole optimization examines:

A) Entire programs at once

B) Small sequences of target instructions

C) Parse trees only

D) Symbol tables

Answer: B

26. Removing redundant instructions is an example of:

A) Syntax analysis

B) Peephole optimization

C) Lexical analysis

D) Heap management

Answer: B

27. Register allocation determines:

    A) The number of parsing tables

    B) Which variables should reside in registers

    C) The number of tokens

    D) The size of the heap

    Answer: B

28. Register assignment determines:

    A) Which specific registers hold variables

    B) Which variables are global

    C) The number of activation records

    D) The order of parsing

    Answer: A

29. Dynamic programming in code generation is used to:

    A) Optimize instruction selection

    B) Perform lexical analysis

    C) Build symbol tables

    D) Allocate heap memory

    Answer: A

30. Which optimization technique replaces a sequence of instructions with a more efficient sequence?

    A) Syntax-directed translation

    B) Peephole optimization

    C) Recursive descent parsing

    D) Garbage collection

    Answer: B