D 70927 (Pages : 2) Name.....

Reg. No.....

THIRD SEMESTER M.Sc. DEGREE (REGULAR) EXAMINATION NOVEMBER 2019

Computer Science

CSS 3C 02—PRINCIPLES OF COMPILERS

Time : Three Hours Maximum : 36 Weightage

Part A

Answer **all** *questions.*
Each question carries 1 *weightage.*

1. Compare assembler and compiler.
2. Write down the regular expression for the following :
   (a) Identifiers of C language.
   (b) To check the correct syntax for the email address.
3. Is it possible to design a compiler without a distinct lexical analysis phase ? Why ?
4. What is an intermediate code ?
5. Define basic block.
6. What are function calls ?
7. Contrast loop fission and loop fusion.
8. What are the problems in compiler optimization ?
9. Define cache management.
10. Explain assignment statements.
11. Compare static and dynamic allocation.
12. Comment on the features of symbol tables.

(12 × 1 = 12 weightage)

Part B

Answer any **six** *questions.*
Each question carries 2 *weightage.*

13. Discuss the principal sources of optimization.
14. Explain activation trees and records. Give examples.

Turn over

15. Discuss the factors affecting code generation.

16. Explain register interference graph construction.

17. Show the annotated parse tree and code generation process for the following arithmetic expression ?

    (a) $a + (b - c) * d.$

    (b) $- a (a + b) * (c + d) + (a * b + c).$

18. What is a type expression ? Compare and contrast weakly typed language and strongly typed language.

19. With suitable example, explain how a regular expression is converted to an NFA.

20. Write a note on compiler construction tools.

21. Compare NFA and DFA with examples.

(6 × 2 = 12 weightage)

Part C

Answer any **three** *questions.*
Each question carries 4 *weightage.*

22. What could be the components of a compiler ? Describe the role of each.

23. Explain control flow analysis and data flow analysis with examples.

24. Explain the algorithm to minimize the number of states of a DFA. Illustrate.

25. Explain the features of recursive descent parser and predictive parser.

26. (i) Show that the following grammar is LR(1) but not LALR(1) :

    S → Aa/bAc/Bc/bBa.
    A → d.
    B → d.

    (ii) Write a note on heap management.

27. (i) Give an overview of Region based analysis in optimization.

    (ii) With example, explain basic blocks and flow graphs. Discuss representation of flow graphs.

(3 × 4 = 12 weightage)