



K18U 0190

Reg. No. :

Name :

VI Semester B.C.A. Degree (CBCSS – Reg./Supple./Imp.) Examination,
May 2018
Core Course
6B21BCA : SYSTEM SOFTWARE
(2014 Admn. Onwards)

Time : 3 Hours

Max. Marks : 40

SECTION – A

1. **One word answer :** (8×0.5=4)
- a) _____ converts assembly language program to object program.
 - b) Hardware device capable to execute programming instruction is _____
 - c) _____ register holds the address of next executable instruction during program execution.
 - d) Basic blocks can be analysed by _____
 - e) Resolution of externally defined symbol is handled by _____
 - f) Bootstrap loader resides in _____
 - g) The method which merges the bodies of two loops is _____
 - h) The errors pointed out by a compiler usually known as _____

SECTION – B

Write short notes on **any seven** of the following questions : (7×2=14)

- 2. What do you mean by forward references ?
- 3. List the classifications of grammar based on nature of production.
- 4. What is DFA ?
- 5. What is operator precedence grammar ?

P.T.O.

K18U 0190



6. Explain the role of assembler directives.
7. What is cross referencing ?
8. What is dead code elimination ?
9. Give the use of entry and extern.
10. Give an example for 3 address code.
11. Briefly explain the purpose of segment index field in LE data record.

SECTION – C

Answer **any four** of the following questions :

(4×3=12)

12. Discuss the problems associated with deletion in Hash Table Organisation with rehash techniques.
13. Briefly explain the phases in language processing.
14. Explain process of top down passing.
15. List and explain briefly the data structures and files used in a two pass assembler.
16. Explain conditional macro expansion.
17. Differentiate between static and dynamic memory allocation.

SECTION – D

Write an essay on **any two** of the following :

(2×5=10)

18. Write a procedure to search and locate a symbol in a binary search organisation.
 19. Explain ambiguity in grammar with suitable example.
 20. Explain the functions of a macro processor.
 21. Using a diagram explain the phases of compiling.
-