B.Sc. Computer Science (Optional)
(w. e. f 2014-15 onwards)

Subject Code	Subject Name	Workload Hrs/week		Max Marks	IA	Total Marks
		Th	Pr			
Semester – I						
14BSCCSCT11	Computer Concepts & C-Programming	04	-	80	20	100
14BSCCSCP12	C- Programming Lab	-	04	40	10	50
Semester – II						
14BSCCSCT21	Data Structure Using C	04	-	80	20	100
14BSCCSCP22	Data Structure Lab	-	04	40	10	50
Semester – III						
14BSCCSCT31	OOPs Using C++	04	-	80	20	100
14BSCCSCP32	C++ Programming Lab	-	04	40	10	50
Semester – IV						
14BSCCSCT41	Introduction to UNIX	04	-	80	20	100
14BSCCSCP42	UNIX Programming Lab	-	04	40	10	50
Semester – V						
14BSCCSCT51	Operating System	04	-	80	20	100
14BSCCSCP52	Operating System Lab	-	04	40	10	50
14BSCCSCT53	Database Management System	04	-	80	20	100
14BSCCSCP54	DBMS Lab	-	04	40	10	50
Semester – VI						
14BSCCSCT61	Computer Networks	04	-	80	20	100
14BSCCSCP62	Computer Networks Lab	-	04	40	10	50
14BSCCSCT63	Core Java	04	-	80	20	100
14BSCCSCP64	Java Programming Lab	-	04	40	10	50

Syllabus for B.Sc. Semester – V

COMPUTER SCIENCE (Optional)

14BSCCSCT51 : Operating Systems (Paper – I) Total : 50 Hrs

Unit 1:

Introduction: Batch Systems, Concepts of Multiprogramming and Time Sharing, Parallel, Distributed and real time Systems, Operating System Structures, Components and Services, System programs, Virtual machines. **Process Management :** Process concept, Process scheduling, Co-operating process, Threads, Inter process communication, CPU scheduling criteria, Scheduling algorithm. **12Hrs**

Unit 2:

Process synchronization and deadlocks: The critical section problem, Synchronization hardware, Semaphores, Classical problems of synchronization, Critical regions, monitors, Dead locks –System model , characterization, Dead lock prevention, avoidance and detection, Recovery from dead lock. **10 Hrs**

Unit 3:

Memory Management: Logical and Physical address space, Swapping Contiguous allocation, Paging, Segmentation, Virtual memory – Demand paging and it's performance, page replacement algorithms, Allocation of frames, thrashing**. 10 Hrs**

Unit 4:

File management (System, Secondary storage structure): File concepts, Access methods, Directory structure, Protection and consistency, semantics, File system structure, Allocation methods, Free space management. **8 Hrs**

Unit 5:

Disk Management (Structure, Disk Scheduling Methods): Disk structure and Scheduling methods, Disk management, Swap – Space management. **Protection and Security:** Goals of protection, Domain protection, Access matrix security problem, Authentication, One time password. **10 Hrs**

Text books:

1. Abraham siberschatz and peter Bear Galvin, Operating System Concepts, Fifth Edition, Addision – Wesley
2. Nutt: Operating system, 3/e person education 2004.

References:

1. Milan Milonkovic, Operating System Concepts and design, II Edition, McGraw Hill 1992.
2. Richard Peterson, Linum – The complete reference.
3. Tanenbaum, Operation System Concepts, Person Education.
4. Nutt, Operating Systems, Person Education.
5. Stallings, Operating Systems, Pearson Education.

14BSCCSCP52 : Operating Systems Lab – B.Sc. Semester - V

(Implement the following on LINUX or other Unix like platform. Use C for high level language implementation)

1. Write programs using the following system calls of UNIX operating system: fork, exec, getpid, exit, wait, close, stat, opendir, readdir
2. Write programs using the I/O system calls of UNIX operating system (open, read, write, etc)
3. Write C programs to simulate UNIX commands like ls, grep, etc.
4. Given the list of processes, their CPU burst times and arrival times, display/print the Gantt chart for FCFS and SJF. For each of the scheduling policies, compute and print the average waiting time and average turnaround time. (2 sessions)
5. Given the list of processes, their CPU burst times and arrival times, display/print the Gantt chart for Priority and Round robin. For each of the scheduling policies, compute and print the average waiting time and average turnaround time. (2 sessions)
6. Developing Application using Inter Process communication (using shared memory, pipes or message queues)
7. Implement the Producer – Consumer problem using semaphores (using UNIX system calls).
8. Implement some memory management schemes – I
9. Implement some memory management schemes – II
10. Implement any file allocation technique (Linked, Indexed or Contiguous)

Example for exercises 8 & 9 :

Free space is maintained as a linked list of nodes with each node having the starting byte address and the ending byte address of a free block. Each memory request consists of the process-id and the amount of storage space required in bytes. Allocated memory space is again maintained as a linked list of nodes with each node having the process-id, starting byte address and the ending byte address of the allocated space. When a process finishes (taken as input) the appropriate node from the allocated list should be deleted and this free disk space should be added to the free space list. [Care should be taken to merge contiguous free blocks into one single block. This results in deleting more than one node from the free space list and changing the start and end address in the appropriate node]. For allocation use first fit, worst fit and best fit

Additional Programs :

1) To write a C-Program to simulate the fork () & exit() system call.
2) To write a C-Program for displaying a command using exec() and wait() system call.
3) To write a C-Program to perform the system call to get the process id.
4) To write a C program to perform open and read a directory using UNIX Operating systems.
5) To write a C-program to perform open and read a directory using UNIX Operating systems.
6) To write a program using the I/O system calls (open, read, write) of UNIX operating system.
7) To write a C-program for simulating ls command in UNIX
8) To write a C-program in UNIX environment to implement the First Come First Serve scheduling with arrival time.
9) To write a C-program in UNIX environment to implement the Shortest Job First Scheduling.
10) To write a C-program in UNIX environment to implement the Priority Scheduling.
11) To write a C-program in UNIX environment to implement the Round Robin Scheduling.
12) To write a C-program to develop an application using Inter process Communication (IPC) using pipes.
13) To write a C-program to develop an application using Inter process Communication (IPC) using Shared Memory.
14) To write a C-program to implement producer consumer relationship using semaphore.
15) To write a C-program in UNIX to implement Dynamic Storage Allocation Strategy for First Fit.
16) To write a C-program in UNIX to implement Dynamic Storage Allocation Strategy for Best Fit.
17) To write a C-program in UNIX to implement Dynamic Storage Allocation Strategy for Worst Fit.
18) To write a C-program in UNIX to implement Contiguous Allocation.
19) To implement FIFO and LRU Pages replacement algorithms
20) To write a C-program for linked allocation of a file
21) To implement Bankers algorithm for Deadlock Detection
22) To implement the program for the Deadlock Avoidance.
23) To simulate Bankers algorithm for Deadlock Prevention.

Practical Examination (Scheme of Valuation)

Evaluation criteria for practical examinations shall be as follows:

1. Writing of Programs -15 Marks

a. One program from the journal list – 08 Marks

b. Another program given by examiner based on the concepts studied -07Marks

2. Execution of programs – 15 Marks

a. Journal Program - 08 Marks

b. Program of Examiner's Choice -07 Marks

3. Viva-Voce -05 Marks

4. Journal / Laboratory Report – 5 Marks

Total Marks -40 Marks

Syllabus for B.Sc. Semester – V

COMPUTER SCIENCE (Optional)

14BSCCSCT53 : Database Management Systems (Paper – II) Total : 50 Hrs

Unit 1:

Introduction: Database and Database Users, Characteristics of the Database Approach, Actors on the scene, Workers behind the Scene, Advantages of using DBMS, Brief History. **Database System Concepts and Architecture:** Data Models, Schemas, and Instances, Three Schema Architecture and Data Independence, Database language and interfaces, the database system Environment, Centralized and Client/Server Architectures for DBMS, Classification of Database Management Systems. **10 Hrs**

Unit 2:

Data modeling using the Entity–Relationship (ER) model: High level conceptual data models for database design with an example, Entity types, Entity sets, Attributes and Keys, Relationship types, Relationship sets, Roles and Structural Constraints, Weak Entity Types, ER Diagrams, Naming Conventions, and Design Issues, Relationship types of degree higher than two, EER Model. **10Hrs**

Unit 3:

Relational Data Model and Relational Algebra: Relation Data Model and Relational Database Constraints, Relation Algebra, Relational Database Design by ER and EER to Relational Mapping. **10Hrs**

Unit 4:

Functional dependencies and Normalization for Relational Databases: Informal Design Guidelines for Relational Schemas, Functional Dependencies, Normal Forms based on Primary Keys, General Definition of 2NF and 3NF, Boyce-Codd Normal Form(BCNF). **10Hrs**

Unit 5:

Relational Database Language: Data definition in SQL, Queries in SQL, Insert, Delete and Update Statements in SQL, Views in SQL, Specifying General Constraints as Assertions, Specifying indexes, Embedded SQL. **Transaction Processing Concepts**: Introduction, Transaction and System Concepts, Desirable properties of transaction, Schedules and Recoverability, Serializability of Schedules, Transaction Support in SQL, Locking Techniques for Concurrency Control. **10Hrs**

Text Book:

A. Ramez Elmasri & Shamkant B. Navathe, Fundamentals of Database Systems(Sixth Edition), Pearson Education, 2011).

B. Sundarraman , Oracle 9i programming A Primer, 1/e Pearson Education.

References:

1. Kahate, Introduction to Database Management System, Pearson Education 2004.
2. Abrahamsi, Silberschatag, Henry. F. Korth, S. Sudarshan, Database System Concepts, Mc. Raw hill.
3. Jefry . D. Ullman, Principles of database system.
4. Oracle Press: ORACLE – Computer reference.
5. C.J. Date, Introduction to database systems, Sixth Edition, Addisonwesley 1995.
6. Raghu Ram Krishnan, Database Management Systems, Second Edition, Mc Graw Hill, 2000.

14BSCCSCP54 : Database Management Systems Lab – B.Sc. Semester - V

Journal programs

I. Consider the Insurance database given below. The primary keys are underlined and the data types are specified.

PERSON (driver – id #: String, name: string, address: strong)
CAR (Regno : string, model: string, year: int)
ACCIDENT (report-number: int, accd-date: date, location: string)
OWNS (driver-id #:string, Regno:string)
PARTICIPATED (driver-id: string, Regno:string, report-number:int, damageamount:int)

a) Create the above tables by properly specifying the primary keys and the foreign keys.
b) Enter at least five tuples for each relation.
c) Demonstrate how you
    a. Update the damage amount for the car with a specific Regno in the accident with report number 12 to 25000.
    b. Add a new accident to the database.
d) Find the total number of people who owned cars that were involved in accidents in 2008.
e) Find the number of accidents in which cars belonging to a specific model were involved.
f) Generate suitable reports.

II. Consider the following relations for an order processing database application in a company.

CUSTOMER (cust #: int , cname: string, city: string)
ORDER (order #: int, odate: date, cust #: int, ord-Amt: int)
ORDER – ITEM (order #: int, item #: int, qty: int)
ITEM (item # : int, unit price: int)
SHIPMENT (order #: int, warehouse#: int, ship-date: date)
WAREHOUSE (warehouse #: int, city: string)

a) Create the above tables by properly specifying the primary keys and the foreign keys.
b) Enter at least five tuples for each relation.
c) Produce a listing: CUSTNAME, of orders, AVG_ORDER_AMT, where the middle column is the total numbers of orders by the customer and the last column is the average order amount for that customer.
d) List the order# for orders that were shipped from *all* the warehouses that the company has in a specific city.
e) Demonstrate the deletion of an item from the ITEM table and demonstrate a method of handling the rows in the ORDER_ITEM table that contain this particular item.
f) Generate suitable reports.

III. Consider the following database of student enrolment in courses & books adopted for each course.

STUDENT (regno: string, name: string, major: string, bdate:date)
COURSE (course #:int, cname:string, dept:string)
ENROLL (regno:string, course#:int, sem:int, marks:int)
BOOK _ ADOPTION (course# :int, sem:int, book-ISBN:int)
TEXT (book-ISBN:int, book-title:string, publisher:string, author:string)

a) Create the above tables by properly specifying the primary keys and the foreign keys.
b) Enter at least five tuples for each relation.
c) Demonstrate how you add a new text book to the database and make this book be adopted by some department.
d) Produce a list of text books (include Course #, Book-ISBN, Book-title) in the alphabetical order for courses offered by the 'CS' department that use more than two books.
e) List any department that has *all* its adopted books published by a specific publisher.
f) Generate suitable reports.

IV. The following tables are maintained by a book dealer.

AUTHOR (author-id: int, name: string, city: string, country: string)
PUBLISHER (publisher-id: int, name: string, city: string, country: string)
CATALOG (book-id: int, title: string, author-id: int, publisher-id: int, category-id: int, year: int, price: int)
CATEGORY (category-id: int, description: string)
ORDER-DETAILS (order-no: int, book-id: int, quantity: int)

a) Create the above tables by properly specifying the primary keys and the foreign keys.
b) Enter at least five tuples for each relation.
c) Give the details of the authors who have 2 or more books in the catalog and the price of the books is greater than the average price of the books in the catalog and the year of publication is after 2000.
d) Find the author of the book which has maximum sales.
e) Demonstrate how you increase the price of books published by a specific publisher by 10%.
f) Generate suitable reports.

V. Consider the following database for a banking enterprise

BRANCH (branch-name: string, branch-city: string, assets: real)
ACCOUNT (accno: int, branch-name: string, balance: real)
DEPOSITOR (customer-name: string, accno: int)
CUSTOMER (customer-name: string, customer-street: string, customer-city: string)
LOAN (loan-number: int, branch-name: string, amount: real)
BORROWER (customer-name: string, loan-number: int)

a) Create the above tables by properly specifying the primary keys and the foreign keys
b) Enter at least five tuples for each relation
c) Find all the customers who have at least two accounts at the *Main* branch.

d) Find all the customers who have an account at *all* the branches located in a specific city.
e) Demonstrate how you delete all account tuples at every branch located in a specific city.
f) Generate suitable reports.

Practical Examination

Evaluation criteria for practical examinations shall be as follows:

1. Writing of Programs -15 Marks

a. One program from the journal list – 08 Marks

b. Another program given by examiner based on the concepts studied -07Marks

2. Execution of programs – 15 Marks

a. Journal Program - 08 Marks

b. Program of Examiner's Choice -07 Marks

3. Viva-Voce -05 Marks

4. Journal / Laboratory Report – 5 Marks

Total Marks -40 Marks

Syllabus for B.Sc. Semester – VI

COMPUTER SCIENCE (Optional)

14BSCCSCT61 : Computer Networks (Paper – I) Total : 50 Hrs

Unit 1:

Introduction: Computer Networks and its applications, Network structure, network architecture, Topologies, LAN, WAN, MAN, The OSI reference model, The TCP/IP reference model, services - SMDS, Frame relay, network standards, example networks, **The Physical Layer:** Transmission Media – Twisted pair, coaxial cable, optical fiber, radio transmission, microwaves and infrared transmission, Switching – message switching Circuit switching, packet switching

10 Hrs

Unit 2:

The Data Link Layer: Data Link Layer design issues, Error detection – Single parity checking, polynomial codes – CRC, Error correction- Hamming code, Elementary data link protocols, sliding window protocols, Example data link protocols. **10 Hrs**

Unit 3:

The Medium Access Control: The channel allocation problem, multiple access protocols – ALOHA, Slotted ALOHA, CSMA protocols, Collision free protocols, Ethernet, Wireless LAN, Bluetooth. **10 Hrs**

Unit 4:

The network Layer: Network layer design issues, Routing algorithms – Flooding, Distance vector routing , Hierarchical routing, Link state routing, Congestion control algorithms – Leaky bucket, token bucket algorithm, admission control, hop by hop choke packets , Quality of Service. **10 Hrs**

Unit 5:

The Transport Layer and Application Layer: Transport service, Elements of Transport protocols, Internet transport protocols (TCP & UDP), DNS, Electronic Mailing, and World Wide Web. **10 Hrs**

TEXT BOOKS:

1. Andrew S. Tanenbaum, David J. Wetherall, Computer Networks, Fifth Edition, Pearson Pub. 2012.

References:

1. Ulyses Black, Computer Networks: Protocols, standard and interfaces, PHI.

2. James Martin, Local Area Networks: Architecture and implementation, PHI.

3. Behrous Foruzan, Data Communication and Networking. TMH.

4. W. Stallings, Data and Computer Communications, Pearson Education.

5. Prakash Gupta, Data Communications, PHI.

6. James F. Kurose & Keith W. Ross, Computer Networking A TOP DOWN Approach Featuring the Internet, 2nd Edition, Pearson Education.

14BSCCSCP62 : Computer Networks Lab – B.Sc. Semester - VI

1) Programs using TCP Sockets (like date and time server & client, echo server & Client, etc...)
2) Programs using UDP Sockets (like simple DNS)
3) Programs using raw sockets (like packet capturing and filtering)
4) Programs using RPC
5) Simulation of sliding window protocols
6) Experiments using simulators (like OPNET)
7) Performance comparison of MAC protocols
8) Performance comparison of Routing protocols
9) Study of TCP/UDP performance

Syllabus for B.Sc. Semester – VI

COMPUTER SCIENCE (Optional)

14BSCCSCT63 : Core Java (Paper – II) Total : 50 Hrs

Unit 1:

Introduction: Internet origin and development – internet architecture frame work world wide web. **Introduction to JAVA:** JAVA Evolution, Java History, Java features, How java differs from C and C++, Java and Internet, Java and World Wide Web. Web Browsers, Hardware and Software requirements, Java support system, Java Environment. Overview of JAVA Language: Introduction, Simple Java Program, More of Java, An Application with Two Classes Java Program structure, Java Tokens, Java Statements, Implementing a Java Program, Java Virtual Machine, Command Line Arguments, Programming Style. Constants, Variables and Data Types: Introduction, Constants, Variables, Data Types, Declaration of Variables, Giving Values to Variables, Scope of variables, Symbolic Constants, Type Casting, Getting Values of Variables, Standard Default Values, Operators and Expressions; Introduction, Arithmetic Operators, Relational Operators, Logical Operators, Bitwise Operators, Special Operators, Arithmetic Expressions, Evaluation of Expressions, Precedence of Arithmetic Operators, Type conversion and Associatively, Mathematical Functions. Decision Making and Branching: Introduction, Decision making with if Statement, Simple if Statement, The if…… else Statement, Nesting of if…….. else Statement, The else if Ladder, The Switch Statement, The ?: Operator. Decision Making and Looping: Introduction. The while Statement, The do Statement, The for Statement, Jumps in Loops Labeled Loops.

12 Hrs

Unit 2:

Classes, Arrays, Strings and Vectors: Classes, Objects and Methods: Introduction, Defining a Class, Adding Variables, Adding methods, Creating Objects, Accessing Class members, Constructors, Methods Overloading, Static members, nesting of Methods, Inheritance: Extending a Class Overriding Methods, Final Variables and methods, Finalizer methods, Abstract methods and Classes, Visibility Control. Arrays, Strings and Vectors: Arrays, One – Dimensional Arrays, Creating an Array, Two – dimensional Arrays, Strings, Vectors, Wrapper Classes.

10 Hrs

Unit 3:

Interfaces, Packages and Multithreaded Programming: Interfaces: Multiple Inheritance: Introduction, Defining Interfaces, Extending Interfaces, Implementing Interfaces, Accessing Interface Variable. **Packages:** Putting Classes together: Introduction, Java API Package, Using System Packages, Naming Conventions, Creating Packages, Accessing a Packages, Using a Packages, Adding a Class to a Package, Hiding Classes.

Multithreaded Programming: Introduction, Creating Threads, Extending the Thread Methods, Thread Exceptions, Thread Priority, Synchronization, Implementing the 'Runnable' Interface.

10 Hrs

Unit 4:

Managing Exceptions, Applet Programming: Managing Errors and Exception: Introduction, Types of Errors, Exceptions, Syntax of Exception handling Code, Multiple Catch Statements, Using Finally Statement, Throwing Our Own Exceptions, Using Exceptions for Debugging. Applet Programming: Introduction, how Applets Differ from Applications, Preparing to Write Applets, Building Applet Code, Applet Life Cycle, Creating an Executable applet, Designing a Web Page, Applet Tag, Adding Applet to HTML File, running the Applet, More about Applet Tag, Passing Parameters to Applets, Aligning the Display, More About HTML Tags, Displaying Numerical Values, Getting Input from the User.

12 Hrs

Unit 5:

Graphics Programming, Input / Output: Graphics Programming: Introduction, The Graphics Class, Lines and rectangles, circles and Ellipses, Drawing Arcs, Drawing Polygons, Line Graphs, Using Control Loops in Applets, Drawing Bar Charts. **Managing Input / Output in JAVA:** Introduction, Concept of Streams, Stream Classes, Byte Stream Classes, Character Stream Classes, Using Streams. Other Useful I/O Classes, Using the File Class, Input / Output Exceptions, Creation of Files, Reading/Writing Characters, Reading/Writing Bytes, handling Primitive Data Types, Concatenating and Buffering Files, Interactive Input and Output, Other Stream Classes.

8 Hrs

TEXT BOOKS:

1. E. Balaguruswamy, Programming with JAVA,A Primer, 4th Edition., TMH (1999), (Chapter 2 – 16)
2. Shishir Gundavaram, CGI Programming on the "World Wide Web, O'Reilly and Associates, (1996). (Chapter 1)

References:

1. Thomas Boutel, CGI programming in C and Perl, Addison – Wesley(1996).
2. Jefry Dwight et al, Using CGI,(second Edition), Prentice Hall, India, (1997).
3. Darrel Ince and Adam Freeman, Programming the Internet with Java, Addison Wesley, (1997).
4. Ken Arnold and James Gosling, the Java Programming Language, Addison – Welsey (1998).
5. Patrick Naughton and Herbert Schildt, JAVA 2: The Complete Reference, 3rd Edition,
TMH,(1999).

14BSCCSCP64 : Core Java Lab – B.Sc. Semester - VI

Java Programming LAB

Sample programs

1. Program to calculate the distance travelled by light in 100 years
2. Program to demonstrate dynamic initialization
3. Program to find prime series up to n by accepting the limit from user from command line arguments.
4. Program to demonstrate the use of all iterative and jump statements.
5. Program to implement thread, applet and graphics by implementing animation of ball moving.

Journal programs

1. Program to demonstrate typecasting and type promotions in java.
2. Program to implement all bitwise operations by reading the input by user and display input and output errors.
3. Program to demonstrate method overloading.
4. Program to implement at least 10 string operations on Strings.
5. Program to demonstrate multilevel inheritance. Show the usage of super ().
6. Program to demonstrate method overriding and dynamic method dispatch.
7. Program to demonstrate constructor overloading by passing different number of parameters of different types.
8. Program to demonstrate a) Packages b) Interfaces.
9. Program to illustrate the usage of try, catch, throws and finally to show exception handling in java.
10. Program to show thread synchronization by creating threads using runnable interface.
11. Program to demonstrate thread priorities. Create the thread by extending thread class.
12. Program to create student report using applet, read the input using text boxes and generate the grades.
13. Program to demonstrate a) Abstract class b) Inner class
14. Program to demonstrate drawing bar chart in applets using graphics programming.
15. Program to copy bytes from one file to another.
16. Program to implement mouse events.

Practical Examination

Evaluation criteria for practical examinations shall be as follows:

1. Writing of Programs -15 Marks

a. One program from the journal list – 08 Marks

b. Another program given by examiner based on the concepts studied -07Marks

2. Execution of programs – 15 Marks

a. Journal Program - 08 Marks

b. Program of Examiner's Choice -07 Marks

3. Viva-Voce -05 Marks

4. Journal / Laboratory Report – 5 Marks

Total Marks -40 Marks

COMPUTER SCIENCE (Optional)

QUESTION PAPER PATTERN FOR ALL SEMESTER

THEORY PAPERS :

Question paper has to be set for total marks of 80.

Section–A: Ten questions to be answered out of twelve each carry 2 marks

2 x 10 = 20

Note: Two questions to be set from each unit, and last two questions from any unit.

Section–B: Five questions to be answered out of seven each carry 4 marks

4 x 5 = 20

Section–C: Four questions to be answered out of six each carry 10 marks

10 x 4 = 40

Total Marks = 80

Practical Examination

Evaluation criteria for practical examinations shall be as follows:

1. Writing of Programs -15 Marks

a. One program from the journal list – 08 Marks

b. Another program given by examiner based on the concepts studied -07Marks

2. Execution of programs – 15 Marks

a. Journal Program - 08 Marks

b. Program of Examiner's Choice -07 Marks

3. Viva-Voce -05 Marks

4. Journal / Laboratory Report – 5 Marks

Total Marks -40 Marks

COMPUTER SCIENCE (Optional)

QUESTION PAPER PATTERN FOR ALL SEMESTER

(w. e. f 2014-15 onwards)

Time : 3 Hours] **[Max. Marks: 80**

Section A

Answer any ten questions, each carries two marks. 2 x 10 = 20

1)

2)

3)

4)

5)

6)

7)

8)

9)

10)

11)

12)

Section B

Answer any five questions, each carries 4 marks. 4 x 5 = 20

13)

14)

15)

16)

17)

18)

19)

Section C

Answer any four questions, each carries 10 marks. 10 x 4 = 40

20)

21)

22)

23)

24)

25)

* * * * *