

# Solapur University, Solapur

## Semester Pattern Revised Syllabus of B. Sc. II – Computer Science w. e. f. June – 2015

### Credit System Structure for B. Sc. – II, Semester – III

Class	Semester	Subject	Title of Papers/Practicals	Hrs/Week			Paper Marks		
				L	T	P	Total	UA	CA
B.Sc. II	III	Computer Science	Paper III – Object Oriented Programming using C++	3	-	-	100	70	30
			Paper IV – RDBMS	3	-	-	100	70	30
			<b>Total</b>	<b>6</b>			<b>200</b>		
		Practical- Computer Science	CSSC CSPR – Lab – II	4					
			CSSC CSPR – Lab – III	4					
<b>Total</b>				<b>8</b>					

Abbreviations: L: lectures, T: Tutorials, P: Practicals; UA: University Assessment by End Semester Examination; CA: College assessment by Internal Continuous Examination

UA (University Assessment): University Theory paper shall be of 70 marks for 3.00 hrs duration

CA (College Assessment): The internal examination for Theory and Practical course.

### Credit System Structure for B. Sc. – II, Semester – IV

Class	Semester	Subject	No. of Papers/ Practicals	Hrs/Week			Paper Marks			Practicals		
				L	T	P	Total	UA	CA	Total	UA	CA
B.Sc. II	IV	Computer Science	V – Data Structure	3	-	-	100	70	30			
			VI – System Analysis and Design	3	-	-	100	70	30			
<b>Total</b>				<b>6</b>			<b>200</b>					
		Practical-Computer Science	CSSC CSPR – Lab – II	-	-	4				100	70	30
			CSSC CSPR – Lab – III	-	-	4				100	70	30
<b>Total</b>						<b>8</b>				<b>200</b>		

Abbreviations: L: lectures, T: Tutorials, P: Practicals; UA: University Assessment by End semester Examination; CA: College assessment by Internal Continuous Examination

UA (University Assessment): University Theory paper shall be of 70 marks for 3.00 hrs duration

CA (College Assessment): The internal examination for theory and Practical course.

## **B. Sc. – II (Computer Science) Semester – III**

### **Paper III: Object Oriented Programming using C++**

**45** contact hours, **3** credits

#### **Unit 1**

[10]

Object oriented concepts Difference between object oriented and procedural oriented programming, the object oriented approach, Object oriented design, Concept of OOP –Data, abstraction, Encapsulation, Inheritance, Polymorphism. Introduction to C++: Introduction, Terminology –Tokens, keywords, Identifiers, Basic Data types, Operators, Input –Output streams, Structure of C++.

#### **Unit 2**

[15]

Concept of Class and Object, Simple class, Member function, private, public & protected member, Array of objects, Nested class, Passing objects as parameter, Inline function, reference arguments Constructor and Destructor. Introduction of constructor and destructor, Default constructor, Copy constructor, parameterized constructor, Multiple Constructor in class, Friend function.

#### **Unit 3**

[20]

Polymorphism: Concepts, Types of polymorphism, Overloading of function, Virtual function. Operator overloading and type conversions, Concept of operator overloading, Rules for overloading operators, Overloading of Unary, Binary and Special operators, Type conversion, Dynamic memory allocation (New and Delete), this pointer, Dynamic Initialization of variable, reference variable Inheritance: Concept of inheritance, defining base and derived classes, Behavior of constructor and destructor in inheritance, Types of Inheritance, Virtual Class, Delegation.

#### **Reference Books**

1. Object oriented programming by E. Balgurusamy
2. Mastering C++ by Venugopal
3. Mastering C++ by Ravichandran

## **Paper IV: RDBMS**

**45** contact hours, **3** credits

### **Unit 1** [10]

Introduction to RDBMS: Definition of data and data information, database, Concepts of DBMS & RDBMS, DBA & responsibilities of DBA, RDBMS terminology, Relation, Attribute, Domain, Tuple.

### **Unit 2** [20]

SQL (Structured Query Language): Features of SQL, Data types, Integrity Constraints, Classification of SQL commands, DDL and DML commands, SQL operators and clauses, Logical, Relational, in, between, like operator, Order by, group by, having clause, SQL functions: Arithmetic functions, Conversion functions, Aggregate functions, Date Functions, String Functions, Views, indexes, sequence, synonyms, Sub queries and join, Sub queries and nesting sub queries, Join: Cartesian Join, Equi-Join, Self Join.

### **Unit 3** [15]

PL-SQL: Comparison between SQL and PL/SQL, Structure of PL/SQL, If-else construct, Loop statement for loop, while loop, Cursor: Definition of cursor, Types of cursor-implicit, Explicit, Open, Fetch, Cursor, Attributes, Close cursor, Procedure & function: Definition of procedure & functions., IN, OUT AND INOUT Parameters, Triggers, Packages, Exception handling.

### **Reference Books**

1. Database System Concepts-Korth, Silberschetz, Sudarshan.
2. Oracle, the Complete Reference- Oracle Press.
3. SQL & PL/SQL-Ivan Bayross

## **B. Sc. – II (Computer Science) Semester – IV**

### **Paper V: Data Structures**

**45** contact hours, **3** credits

#### **Unit 1**

[20]

Introduction to Data structure and Array: Concept of Abstract Data Types, Definitions – Data types, Data objects, Data Structure., Array.

Stack: Definition of stack, Operations on stack, Representation of Stack, Applications of stack –Recursion, infix, prefix and postfix, expression.

Queue: Definition of queue, operations on queue, Types of queue –Linear and Circular queue, Implementation of Linear and Circular Queue., Priority queue, Applications of queue Comparison between Stack and Queue.

#### **Unit 2**

[15]

Linked List: Drawbacks of sequential storage, Concept of Linked List, Implementation of Linked List, Operations on List, Operation On circular linked list, Doubly Linked List, Implementation of Stack and queue using linked list.

Trees: Tree terminology, Representation of Binary Trees, Operation on binary trees, Tree Traversal – (Preorder, Inorder, Postorder), AVL tree, threaded binary tree.

#### **Unit 3**

[10]

Sorting and Searching: Introduction, Efficiency consideration, Sorting Methods., Exchange Sort, Bubble Sort, Insertion Sort., Selection sort and tree sort, Merge and radix sort, Linear Search, Binary Search- B-Tree, B+-Tree., Indexed sequential search, Tree search- multiway search tree, Hashing.

#### **Reference Books**

1. Data structure using C++ by Tannenbaum
2. Aho, Hopcroft, Ulman: Data structures and Algorithms.
3. Niklaus Wirth: Algorithms, data structures, Programs.

## **Paper VI: System Analysis and Design**

**45** contact hours, **3** credits

### **Unit 1**

[18]

System Analysis and design overview: Meaning and definition, Characteristics, Elements of System, and Types of system, subsystem, System development life cycle, and feasibility study operational, Technical, and Economical.

System Analysis: Definition, Role & skills of system Analyst, System planning and initial Investigation, Fact finding Technique – Interviews, Questionnaires, Record reviews, observation.

### **Unit 2**

[17]

Charting Technique: Decision Tables, Decision Trees, Program flowchart, System flowchart, Table contents.

Process and Data design: Context diagram, Data flow diagram- Level of DFD, Drawing DFD for payroll system etc, Concept of Entity. Attributes, Types of relation, Entity.

Relationship Diagram, Normalization –forms of normalization (up to 3 NF).

System Design: Input Design, Output Design, File Design.

### **Unit 3**

[10]

System Implementation: Hardware and software selection, System Testing-Black Box, White Box Testing etc., System Implementation.

Case studies: Library information system, college admission system, payroll system.

### **Reference Books**

1. System Analysis and Design-Awad E. H.
2. System Analysis and Design- Parthsarathy/Khalkar
3. Software Engineering –Roger Pressman

## **Laboratory course in Computer Science**

### **Lab II: Laboratory course – II**

1. Write a program to calculate factorial of given number by using recursion.
2. Write a program for addition, subtraction, multiplication and division of two complex numbers by using return by object method.
3. Create 2 distance classes class A stores distance in meter and cm and B stores distance in feet and inches and add two distances by friend function and display the result.
4. Generate the result for 5 students with following data -  
Name, exam no., Theory marks in 5 subject, grade.  
Use any form of constructor.
5. Write a program to calculate root of quadratic equation by using default argument constructor.
6. Write a program to demonstrate friend function, friend class, member function of a class is friend to another class.
7. Write a program to display no. of objects created by using static data member & member function.
8. Write a program to overload unary operators (++ , -- , -).
9. Write a program to overload binary operator.(+ , - , \* , / , %).
10. Write a program to overload binary operator.(+ , - , \* , / , %) by using friend function.
11. Write a program to concatenate string using operator overloading.
12. Write a program to implement--
  - a. Simple inheritance.
  - b. Multiple inheritance.
  - c. Multilevel inheritance.
  - d. Multipath inheritance.
  - e. Hybrid inheritance.
  - f. Hierarchical inheritance.
13. Write a program to for hybrid inheritance by using virtual base class
14. Write a program to demonstrate constructor invocation (use 4 classes)
15. Write a program to demonstrate use to virtual function
16. Write a program to implement all manipulators
17. Write a program to implement istream class
18. Write a program to implement ostream class
19. Write a program to copy one file into another file
20. Write a program to append one file into another
21. Write a program to which shows sequential file access
22. Write a program to random file access
23. Write a program to implement command line argument
24. Write a recursive function
  - a. to find the prime number.
  - b. to find out face value of given number.  
(e.g. 5678 -> 26 -> 8)
25. Write a menu driven program to implement stack (using array)
26. Write a program to check whether the expression is valid or not.
27. Write a program to convert infix expression to postfix.
28. Write a program to implement queue (array).

29. Write a program to implement queue dynamically.
30. Write a program to implement stack dynamically.
31. Write a menu driven program to implement singly linked list with operation
  - a) insert at beginning of linked list
  - b) delete the first node
  - c) insert at the end of linked list
  - d) delete the last node
  - e) display the list
32. Write a menu driven program to implement singly linked list with operation
  - a) sort list
  - b) maximum value
  - c) minimum value
  - d) find & replace a value
  - e) count the number of nodes.
33. Write a program to implement singly circular linked list.
34. Write a menu driven program to implement various operations on doubly linked list with operation.
35. Write a program to create binary search tree and display its contents by using inorder traversal method.
36. Write a program to create binary search tree and display its contents by using preorder traversal & postorder traversal method.
37. Write programs to implement
  - a) Bubble Sort Technique.
  - b) Straight Selection Sort Technique.
  - c) Simple Insertion Sort Technique.
  - d) Shell Sort Technique.
  - e) Quick Sort Technique.
  - f) binary search tree Technique.
  - g) heap sort technique.
  - h) address calculation Sort Technique.
  - i) Radix Sort Technique.
  - j) Merge Sort Technique.
38. Write programs to implement
  - a) sequential searching Technique.
  - b) indexed searching Technique.
39. Write a program to implement simple hash function.
40. Write a program to traverse a graph through BFS method.
41. Write a program to traverse a graph through DFS method.
42. Write a program to add two polynomials by using array (single variable).
43. Write a program to add two polynomials by using Linked list (single variable).
44. Write a program to subtract two polynomials by using Linked list (single variable).
45. Write a program to represent matrix as sparse matrix.
46. Write a program to transpose sparse matrix.
47. Write a program to add two sparse matrix.



## Laboratory course in Computer Science

### Lab III: Laboratory course – III

1. Consider the following table and solve the following queries:

Table Name: Employee

Column\_name Datatype Constraint Description

Eno Varchar2(6) Primary key Starts from 'E' character

Ename Varchar2(20) Not Null

Eaddr Varchar2(20) Not Null

Edob Date Not null

Edname Varchar2(20) Not null Dept name must be from sales, purchase, production, research, marketing

Emgr Varchar2(20) Not null

Ejob Varchar2(12) Not null

Edoj Date Not null Date of joining must greater than edob

Esal Number(9,2) Not null Default 5000.00

- i. Insert at least 10 records.
- ii. Display all the employees working in 'sales' dept.
- iii. Sort the employee list according to joining date.
- iv. Increase 5% salary whose experience is more than 2 years.
- v. Display the names of employees in ascending order of employee name.
- vi. Find out employee who are either working as 'Analyst' or salary greater than 5000.
- vii. List the department name, no. of employees in each department
- viii. Find out the employee who is getting maximum salary.
- ix. Remove the employees who work as 'clerk' or 'account'.
- x. Raise the salary of all 'salesman' by 20%.
- xi. Display the names of employees whose age is greater than 50.
- xii. Display the all details of employees who are not manager.
- xiii. Display the names of employees having experience more than 5 years in the company.
- xiv. Display the names of employees whose salary is greater than employee 'ramesh' but less than 'sunil'.
- xv. Display the depart names who are having more than 3 employees.
- xvi. Display the job names whose total salary is greater than 40000 for each job.
- xvii. Display the names of employees who are getting highest salary.
- xviii. Display the names of employees who are getting 5 digit salary.
- xix. Write a query to list the employees who have joined in the last seven days
- xx. List all the employees whose names are having 'R' as last character.
- xxi. Find all the employees who joined the company before their manager.
- xxii. Display the department where there are no employees.
- xxiii. Display those emp whose salary is odd value.

2. Write a PL/SQL block to display the details of given emp\_no.
3. Write a PL/SQL block of code to calculate the area of a circle for a value of radius & store calculated area in a table.
4. The HRD manager has decided to raise the salary for all the employees working as salesman by 0.05%. Whenever such raise is given to the employees, record for the same is maintained in the emp\_raise table. It includes emp\_no, date when the raise was given & actual raise. Write a PL/SQL block to update salary of each employee & insert a record in the emp\_raise table.
5. Define cursor that will accept acc\_no & update the balance amount (bal\_amt) by 3% as interest if bal\_amt > 1,00,000 from the Acc\_Master(acc\_no, name, city, bal\_amt).
6. Consider the table salespeople(snum, sname, city, commission). Write a PL/SQL block to increase commission of a particular salesperson by the given increment value.
7. Consider the following entities & their relationships.
  - Employee(empno, empname, joiningdate, sex, salary, commission, deptno)
  - Department(deptno, deptname, location)
    - i. Display names of all employees working in the 'Accountants' department.
    - ii. Display names of all employees along with their salary and department name.
    - iii. Display names of all departments along with no. of employees working in that department.
8. Consider the following entities & their relationships.
  - Company(company\_name, address, city, phone, share\_value)
  - Person(per\_name, per\_city)
  - Comp\_per(company\_name, per\_name, no\_of\_shares)
 Company & person are related with many-to-many relationship.
  - i. Write a PL/SQL block to transfer the shares owned by 'Mr.Kale' to 'Mr. Joshi'.
  - ii. Write a PL/SQL block to print name of persons along with their total invested values in various companies.

**B. Sc. (Computer Science) – II (Semester Pattern)**  
**Nature of Practical Examination**

1. No. of sections: 02
2. Each section contains 02 questions
3. Student must attempt 3 questions
4. Each question carries 20 marks.
5. Total 60 marks
6. Viva: 10 marks

**Total Marks: 70 marks**