Syllabus for S.Y.B.Sc.(Computer Science) to be implemented from 2014-15

Important to Note about Laboratory courses: It is absolutely necessary and essential that all the practical's for Paper III and Paper IV be conducted on Free and Open Source Operating System like Linux.

- All the practical's related to C and C++ needs to be conducted using GCC compiler.

- For laboratory work/assignments of Database Systems, PostGreSQL to be used.

1) **Title of the Course : B. Sc. Computer Science**

S.Y.B.Sc. Computer Science Syllabus
(To be implemented from Academic Year 2014-15)

2) **Preamble:**
B. Sc. Computer Science is a systematically designed three year course that prepares the student for a career in Software Industry. The syllabus of computer Science subject along with that of the three allied subjects (Mathematics, Electronics and Statistics) forms the required basics for pursuing higher studies in Computer Science. The Syllabus also develops requisite professional skills and problem solving abilities for pursuing a career in Software Industry.

3) **Introduction:**
At **first year of under-graduation** basic foundation of two important skills required for software development is laid. A course in programming and a course in database fundamentals forms the preliminary skill set for solving computational problems. Simultaneously two practical courses are designed to supplement the theoretical training. The second practical course also includes a preliminary preparation for website designing in the form of HTML programming.

Alongwith Computer Science two theory and one practical course each in Statistics, Mathematics and Electronics help in building a strong foundation.

At **second year under-graduation**: The programming skills are further strengthened by a course in Data structures and Object oriented programming. The advanced topics in Databases and preliminary software engineering form the second course. Two practical courses alongside help in hands-on training. Students also undertake a mini project using software engineering principles to solve a real world problem.

Simultaneously two theory and one practical course each in Mathematics and Electronics help in strengthening problem solving abilities.

At **third year under-graduation:** Six theory papers in each semester and practical courses cover the entire spectrum of topics necessary to build knowledge base and requisite skill set. Third practical course also includes project work which gives students hands on experience in solving a real world problem.

Objectives:
- To develop problem solving abilities using a computer
- To build the necessary skill set and analytical abilities for developing computer based solutions for real life problems.
- To imbibe quality software development practices. To create awareness about process and product standards
- To train students in professional skills related to Software Industry.

- To prepare necessary knowledge base for research and development in Computer Science
- To help students build-up a successful career in Computer Science

4) Eligibility:

Higher Secondary School Certificate (10+2) Science stream or its equivalent Examination as per the University of Pune eligibility norms.

Note: Admissions will be given as per the selection procedure / policies adopted by the respective college, in accordance with conditions laid down by the University of Pune.Reservation and relaxation will be as per the Government rules.

5 A) Examination Pattern:
First Year B. Sc. Computer Science
Subject: Computer Science

Pattern of Examination: Annual

Theory courses   (CS-101): Annual   (CS-102): Annual

Practical Course   (CS-103): Annual   (CS-104): Annual

Paper/ Course No.	Title	Total Number of lectures/practical's per Term	Standard of passing		
			Internal marks out of **20**	External marks out of **80**	Total marks out of **100**
Computer Science Paper I (CS-101)	Problem Solving Using Computers and 'C' Programming	Three lectures/Week (Total 80 lectures)	08	32	40 *
Computer Science Paper II CS-102)	File Organization and Fundamental of Databases	Three lectures/Week (Total 80 lectures)	08	32	40 *
Computer Science Practical Paper I (CS-103)	Computer Science Practical Paper I	25 Practical slots of 4 lectures each	08	32	40 *

Computer Science Practical Paper II (CS-104)	Computer Science Practical Paper II	25 Practical slots of 4 lectures each	08	32	40 *

* Subject to compulsory passing in external examination and getting minimum 40 marks out of 100

Notes:
1. Total marks: Theory (100 + 100) = 200 marks
2. Total marks per year 200 (Theory) + 100 marks (practical)+ Grade(practical) = 300 marks +Grade
3. Internal marks for theory papers given on the basis of internal assessment tests and for practicals on continuous assessment of lab work.
4. In case of Computer Science Practical Paper II, marks out of 100 will be converted to grades

Marks	Grade
75 and above	O
65 and above	A
55 and above	B
50 and above	C
45 and above	D
40 and above	E
Below 40 (indicates Failure)	F

Theory examination will be of three hours duration for each theory course. There shall be 5 questions each carrying equal marks. The pattern of question papers shall be:

Question 1	8 sub-questions, each of 2 marks; answerable in 2 -3 lines and based on entire syllabus
Question 2, 3 ,4 and 5	4 out of 5/6– short answer type questions; answerable in 8 – 10 lines ; mix of theory and problems

Internal examination: Internal assessment of the student by respective teacher will be based on written test, 10 marks each term. The written test shall comprise of objective type questions – Multiple Type Questions, True / False, Definitions, Answer in Two or three line question (Describe/Explain).There shall be 20 questions.
Practical: Continuous assessment of Lab work and mini project.

Practical Examination: Practical examination shall be conducted by the respective college at the end of the academic year. Practical examination will be of 3 hours duration for each practical course. Certified journal is compulsory to appear for practical examination. There shall be two expert and two examiners per batch for the practical examination.

Second Year B. Sc. Computer Science

No	Paper	Title: Semester I	Title: Semester II
1	Computer Science Paper I	CS-211:Data Structures using 'C'	CS-221:Object Oriented Concepts using C++
2	Computer Science Paper II	CS-212: Relational Database Management System	CS-222:Software Engineering
3	Computer Science Paper III	CS-223:Data structures Practicals and C++ Practicals	
4	Computer Science Paper IV	CS-224:Database Practicals & Mini Project using Software Engineering techniques	
5	Mathematics Paper I	MT-211:Mathematics Paper I-Sem I	MT-221:Mathematics Paper I-Sem II
6	Mathematics Paper II	MT-212:Mathematics Paper II-Sem I	MT-222:Mathematics Paper II-Sem II
7	Mathematics Paper III	MT-223:Practical Course in Mathematics	
8	Electronics Paper I	EL-211:Electronics Paper I-Sem I	EL-221:Electronics Paper I-Sem II
9	Electronics Paper II	EL-212:Electronics Paper II-Sem I	EL-222:Electronics Paper II-Sem II
10	Electronics Paper III	EL-223:Practical Course in Electronics	
11	English	EN-211:Technical English-Sem I	EN-221:Technical English – Sem II

Pattern of examination: Semester

Theory courses (Sem I: CS-211 and CS212): Semester
(Sem II: CS-221 and CS-222): Semester
Practical Course (CS-223 and CS-224): Annual

| Paper/ Course No. | Title | Total Number of lectures/practicals Per Semester | Standard of passing | | |
			Internal marks out of 10 (theory) Out of 20 (practicals)	External marks out of 40 (theory) Out of 80 (practicals)	Total passing marks out of 50 (theory) and out of 100(practicals)
Theory Paper I (CS- 211)	Data Structures using 'C'	Four lectures/Week (Total 48 per Semester)	04	16	20 *
Theory Paper II (CS 212)	Relational Database Management System	Four lectures/Week (Total 48 per Semester)	04	16	20 *
Theory Paper I (CS 221)	Object Oriented Concepts using C++	Four lectures/Week (Total 48 per Semester)	04	16	20 *
Theory Paper II (CS 222)	Software Engineering	Four lectures/Week (Total 48 per Semester)	04	16	20 *
Practical paper I (CS 223) (First & Second Sem)	Data structures Practicals and C++ Practicals	Practicals of 4 lectures each 25 practicals/Yr.)	08	32	40 **
Practical paper II (CS 223) (First & Second Semester)	Database Practicals & Mini Project using Software Engineering techniques	Practicals of 4 lectures each 25 practicals/ Yr.)	08	32	40 **

* Subject to compulsory passing in external examination and getting minimum 20 marks out of 50
** Subject to compulsory passing in external examination and getting minimum 40 marks out of 100

Notes:
1. Total marks: Theory for each semester (50 + 50) = 100 marks
2. Total marks per year 200 (Theory) + 100 marks (practicals)+Grade(practical) = 300 marks+Grade
3. Internal marks for theory papers given on the basis of Continuous internal assessment

Theory examination will be of two hours duration for each theory course. There shall be 4 questions carrying equal marks. The pattern of question papers shall be:

Question 1	10 questions, each of 1 marks	10 marks
Question 2 3	Sub-questions carrying 5 marks (2 out of 3)	10 marks each
Question 4	Sub-questions carrying marks depending on their complexity with options	10 marks

Internal examination: Internal assessment of the student by respective teacher will be based on written test, 10 marks each Semester. The written test shall comprise of objective type questions – Multiple Type Questions, True / False, Definitions, Answer in Two or three line question (Describe/Explain) There shall be 20 questions.

Practicals: Continuous assessment of practical performance

Practical Examination: Practical examination shall be conducted at the respective college at the end of the academic year. Practical examination will be of 3 hours duration. Certified journal is compulsory to appear for practical examination. There shall be one expert and two examiners per batch for the practical examination. One of the examiners will be external.

Third Year B. Sc. Computer Science

No	Paper	Title: Semester I	Title: Semester II
1	Computer Science Paper I	CS-331:System Programming	CS-341:Operating System
2	Computer Science Paper II	CS-332:Theoretical Computer Science	CS-342:CompilerConstruction
3	Computer Science Paper III	CS-333:Computer Networks-I	CS-343:Computer Networks-II
4	Computer Science Paper IV	CS-334: Internet Programming- I	CS-344:Internet Programming- II
5	Computer Science Paper V	CS-335:Programming in Java-I	CS-345:Programming in Java-II

6	Computer Science Paper VI	CS-336:Object Oriented Software Engineering	CS-346:Computer Graphics
7	Computer Science Paper VII	CS-347:Practicals Based on CS-331 and CS341 – Sem I &Sem II	
8	Computer Science Paper VIII	CS-348:Practicals Based on CS-335 and CS-344 – Sem I &Sem II and Computer Graphics using Java	
9	Computer Science Paper IX	CS-349:Practicals Based on CS-334 and CS-344 – Sem I &Sem II andProject	

Subject: Computer Science

Pattern of examination: Semester
Theory courses:
    (Sem III: CS-331-CS-336): Semester (Sem IV: CS-341-CS-346): Semester
Practical Course:
    (CS-347-CS-349): Annual

Theory Papers					
Paper/Course No.	Title	Total Number of lectures Per Semester	Standard of passing		
			Internal marks out of 10 (theory) Out of 20 (practicals)	External marks out of 40 (theory) Out of 80 (practicals)	Total passing marks out of 50 (theory) and out of 100 (practicals)
SEM III					
CS-331	System Programming	48	4	16	20*
CS-332	Theoretical Computer Science	48	4	16	20*
CS-333	Computer Networks-I	48	4	16	20*
CS-334	Internet Programming- I	48	4	16	20*
CS-335	Programming in Java-I	48	4	16	20*
CS-336	Object Oriented Software Engineering	48	4	16	20*
SEM IV					

CS-341	Operating System	48	4	16	20*
CS-342	Compiler Construction	48	4	16	20*
CS-343	Computer Networks-II	48	4	16	20*
CS-344	Internet Programming- I	48	4	16	20*
CS-345	Programming in Java-I	48	4	16	20*
CS-346	Computer Graphics	48	4	16	20*
Practical Papers					
CS 347 (Semester III & IV)	Practicals Based on CS-331 and CS-341 – Sem I &Sem II	25 practicals/ year	08	32	40 **
CS 348 (Semester III & IV)	CS-348:Practicals Based on CS-335 and Cs-344 – Sem I &Sem II and Computer Graphics using Java	25 practicals/ year	08	32	40 **
CS 349 (Semester III & IV)	CS-349:Practicals Based on CS-334 and CS-344 – Sem I &Sem II and Project	25 practicals/ year	08	32	40 **

* Subject to compulsory passing in external examination and getting minimum 20 marks out of 50
** Subject to compulsory passing in external examination and getting minimum 40 marks out of 100

Notes:
1. Total marks: Theory for each semester (50 $\times$ 6) = 300 marks
2. Total marks per year 600 (Theory) + 300 marks (practicals) = 900 marks
3. Internal marks for theory papers given on the basis of continuous internal assessment

Theory examination will be of two hours duration for each theory course. There shall be 4 questions carrying equal marks. The pattern of question papers shall be:

Question 1	10 questions, each of 1 marks	10 marks
Question 2 and 3	Sub-questions carrying 5 marks (2 out of 3)	10 marks each
Question 4	Sub-questions carrying marks depending on their complexity with options	10 marks

Internal examination: Internal assessment of the student by respective teacher will be based on written test, 10 marks each Semester. The written test shall comprise of objective type questions – Multiple Type Questions, True / False, Definitions, Answer in Two or three line question (Describe/Explain) There shall be 20 questions.
Practicals: one internal assessment test + practical journals + attendance + activity.

Practical Examination: Practical examination shall be conducted at the respective college at the end of the academic year. Practical examination will be of 3 hours duration. Certified journal is compulsory to appear for practical examination. There shall be one expert and two examiners per batch for the practical examination. One of the examiners will be external.

5 B) Standard of Passing:
i. In order to pass in the first year theory examination, the candidate has to obtain 40 marks out of 100 in each course. (Minimum 32 marks out of 80 must be obtained in the University Theory Examination.)
ii. In order to pass in the Second Year and Third Year theory examination, the candidate has to obtain 20 marks out of 50 in each course of each semester. (Minimum 16 marks out of 40 must be obtained in the University Theory Examination.)
iii. In order to pass in practical examination, the candidate has to obtain 40 marks out of 100 in each course. (Minimum 32 marks out of 80 must be obtained in the University Examination.)

5 C) ATKT Rules:

While going from F.Y.B.Sc. to S.Y.B.Sc. at least 8 courses (out of total 13) should be passed; however all F.Y.B.Sc. courses should be passed while going to T.Y.B.Sc.

While going from S.Y.B.Sc. toT.Y.B.Sc., at least 12 courses (out of 22) should be passed (Practical Course at S.Y.B.Sc. will be equivalent to 2 courses).

5 D)Award of Class:

The class will be awarded to the student on the aggregate marks obtained during the second and third year in the principal subject only. The award of the class shall be as follows:

1	Aggregate 70% and above	First Class with Distinction
2	Aggregate 60% and more but less than 70%	First Class
3	Aggregate 55% and more but less than 60%	Higher Second Class
4	Aggregate 50% and more but less than 55%	Second Class
5	Aggregate 40% and more but less than 50%	Pass Class
6	Below 40%	Fail

5 E) External Students: There shall be no external students.

5 F) Setting question papers:

F.Y.B.Sc.: For theory papers I and II annual question papers shall be set by the University of Pune and assessment done at the respective colleges. Questions should be designed to test the conceptual knowledge and understanding of the basic concepts of the subject. For Practical Papers, the Question paper slips will be provided by the University of Pune and assessment done at the respective colleges.

S.Y.B.Sc. and T.Y.B.Sc.: For theory papers I and II for each semester and also for the annual practical examination question papers set by the University of Pune. Centralized assessment for theory papers done as per the University instructions. Questions should be designed to test the conceptual knowledge and understanding of the basic concepts of the subject. For Practical Papers: Papers shall be set by the University of Pune and assessment done by the internal examiner and external examiner appointed by University of Pune.

5G) Verification and Revaluation Rules:
As per university Statues and rules for verification and revaluation of marks in stipulated time after declaration of the semester examination result.

6) Course Structure:
　　Duration: The duration of B.Sc. Computer Science Degree Program shall be three years.
　　a) **All are Compulsory Papers:**
　　　　F.Y.B.Sc. : 2 Theory + 2 Practical (Annual)
　　　　S .Y.B.Sc.: 2 Theory per semester + 2 Practical (Annual)
　　　　T.Y.B.Sc.: 6 Theory per semester + 3 Practical (Annual)
　　b) **Question Papers　　　　　　:**
　　　　F.Y.B.Sc.Theory paper:
　　　　University Examination　– 80 marks (at the end of 2nd term)
　　　　Internal Examination　– 20 marks
　　　　S.Y / T.Y. - B.Sc.Theory paper:
　　　　University Examination　– 40 marks (at the end of each term)
　　　　Internal Examination　– 10 marks
　　　　F.Y. / S.Y / T.Y. - B.Sc.Practical Paper:
　　　　University Examination　– 80 marks (at the end of 2nd term)
　　　　Internal Examination　– 20 marks
　　c) **Medium of Instruction:** The medium of instruction for the course shall be **English.**

7) Equivalence of Previous Syllabus:

Semester & Paper	Title of Paper (Old Pattern)(Implemented from theacademic year 2009-10)	Title of Paper (New Pattern)(to be implemented from the academic year 2014-15)
Semester-I, Paper-I	CS-211, Data Structures Using C	CS-211 Data Structures using 'C'
Semester-I, Paper-II	CS- 212, Relational Database Management System	CS-212 Relational Database Management System
Semester-II, Paper-I	CS-221, Object Oriented Concepts and Programming in C++	CS-221 Object Oriented Conceptsusing C++
Semester-II, Paper-II	CS-222, Software Engineering	CS-222Software Engineering
Practical paper II (CS 223) (First & Second	CS-224: Database Assignments and Mini Project using Software Engineering	CS-224: Database Practicals & Mini Project using Software

Semester)	Techniques	Engineering techniques

8) University Terms: Dates for commencement and conclusion for the first and second terms will be declared by the University authorities. Terms can be kept by only duly admitted students. The term shall be granted only on minimum 75 percent attendance at theory and practical course and satisfactory performance during the term.

9) Qualification of Teachers: M.Sc. Computer Science/M.C.A. or equivalent master degree in science with class/grades and NET/SET as per prevailing University/Government /UGC rules.

10) Detail Syllabus with Recommended Books:
S.Y.B.Sc. Computer Science Paper I

CS-211: Data Structures using 'C'
CS-221: Object Oriented Concepts using C++

S.Y.B.Sc. Computer Science Paper II

CS-212: Relational Database Management System
CS-222: Software Engineering

S.Y.B.Sc. Computer Science Paper III

CS-223: Data structures Practicals and C++ Practicals

S.Y.B.Sc. Computer Science Paper IV

CS-224: Database Practicals & Mini Project using Software Engineering
            techniques

S.Y.B.Sc. Computer Science Theory Paper I
Semester – 1
CS 211- DATA STRUCTURES USING 'C'
(Compulsory Course)

Total Lectures: 48
Objective:
1. To learn the systematic way of solving problem
2. To understand the different methods of organizing large amount of data
3. To efficiently implement the different data structures
4. To efficiently implement solutions for specific problems
Prerequisites: Knowledge of C Programming Language

1. Introduction to data structures [3]
  1.1 Concept
  1.2 Data type, Data object, ADT
    1.2.1 Data Type
1.2.2 Data Object
    1.2.3 ADT -Definition, Operation, examples on rational number
  1.3 Need of Data Structure
  1.4 Types of Data Structure

2. Algorithm analysis [2]
  2.1 Algorithm – definition, characteristics
  2.2 Space complexity, time complexity
  2.3 Asymptotic notation (Big O, Omega Ω)

3. Linear data structures [6]
  3.1 Introduction to Arrays - array representation
  3.2 Sorting algorithms with efficiency
    - Bubble sort, Insertion sort, Merge sort, Quick Sort
  3.3 Searching techniques –Linear Search, Binary search

4. Linked List [8]
  4.1 Introduction to Linked List
  4.2 Implementation of Linked List – Static & Dynamic representation,
  4.3 Types of Linked List
  4.4 Operations on Linked List
   - create, display, insert, delete, reverse, search, sort, concatenate &merge
  4.5 Applications of Linked List – polynomial manipulation
  4.6 Generalized linked list – Concept and Representation

5. Stacks [6]
   5.1 Introduction
   5.2 Representation- Static & Dynamic
   5.3 Operations
   5.4 Application - infix to postfix,infix to prefix, postfix evaluation,
   5.5 Simulating recursion using stack

6. Queues [4]
   6.1 Introduction
   6.2 Representation - Static & Dynamic
   6.3 Operations
   6.4 Circular queue, priority queue (with implementation)
   6.5 Concept of doubly ended queue

7. Trees [12]
   7.1 Concept & Terminologies
   7.2 Binary tree, binary search tree
   7.3 Representation – Static and Dynamic
   7.4 Operations on BST – create, Insert, delete, traversals (preorder, inorder,
       postorder), counting leaf, non-leaf & total nodes , non recursiveinorder
       traversal
   7.5 Application - Heap sort
   7.6 Height balanced tree- AVL trees- Rotations, AVL tree examples.

8. Graph [7]
   8.1 Concept & terminologies
   8.2 Graph Representation – Adjacency matrix, adjacency list, inverse
       Adjacency list, adjacency multilist, orthogonal list
   8.3 Traversals – BFS and DFS
   8.4 Applications – AOV network – topological sort, AOE network – critical
                     path

References:
 1. Fundamentals of Data Structures ---- By Horowitz Sahani (Galgotia)
 2. Data Structures using C and C++ --- By YedidyahLangsam, Aaron M.
    Tenenbaum, Moshe J. Augenstein
 3. Introduction to Data Structures using C---By Ashok Kamthane
 4. Data Structures using C --- Bandopadhyay&Dey (Pearson)
 5. Data Structures using C ---By Srivastava BPB Publication.

Semester – I

CS-212-Relational Database Management System
(Compulsory Course)

Total Lectures: 48
Objective:-

-To teach fundamental concepts of RDBMS (PL/PgSQL)

-To teach principles of databases

-To teach database management operations

-To teach data security and its importance

-To teach client server architecture

Prerequisites: Knowledge of DBMS

1. **Relational Database Design [14]**

1.1 Preliminaries

Functional Dependencies

Basic concepts : Closure of a set of functional dependencies, Closure of attribute set, Canonical cover, Decomposition.

1.2 PL/PgSqL: Datatypes, Language structure

1.3 Controlling the program flow, conditional statements, loops

1.4 Views

1.5 Stored Functions, Stored Procedures

1.6 Handling errors and exceptions

1.7 Cursors

1.8 Triggers

2 **Transaction Concepts and concurrency control [14]**

2.1 Describe a transaction, properties of transaction, state of the transaction.

2.2 Executing transactions concurrently associated problem in concurrent execution.

2.3 Schedules, types of schedules, concept of Serializability, precedencegraph for Serializability.

2.4 Ensuring Serializability by locks, different lock modes, 2PL and its variations.

2.5 Basic timestamp method for concurrency, Thomas Write Rule.

2.6 Locks with multiple granularity, dynamic database concurrency (Phantom Problem).

2.7 Timestamps versus locking.

2.8 Deadlock handling methods

      2.8.1 Detection and Recovery (Wait for graph).

      2.8.2 Prevention algorithms (Wound-wait, Wait-die)

3 Database Integrity and Security Concepts [8]

3.1 Domain constraints

3.2 Referential Integrity

3.3 Introduction to database security concepts

3.4 Methods for database security

      3.4.1 Discretionary access control method

      3.4.2 Mandatory access control and role base access control for multilevel security.

3.5 Use of views in security enforcement.

3.6 Overview of encryption technique for security.

3.7 Statistical database security.

4 Crash Recovery [8]

4.1 Failure classification

4.2 Recovery concepts

4.3 Log base recovery techniques (Deferred and Immediate update)

4.4 Checkpoints

4.5 Recovery with concurrent transactions (Rollback, checkpoints, commit)

4.6 Database backup and recovery from catastrophic failure.

5. Client-Server Technology [4]

5.1 Describe client-server computing.

5.2 Evolution of Client - Server information systems.

5.3 Client – Server Architecture benefits.

5.4 Client Server Architecture

- Components, Principles, Client Components

- Communication middleware components

- Database middleware components

- Client Server Databases

References:-

1. Fundamentals of Database Systems (4th Ed) By: Elmasri and Navathe

2. Database System Concepts (4th Ed) By: Korth, Sudarshan, Silberschatz

3. Practical PostgreSQL O'REILLY

4. Beginning Databases with PostgreSQL, From Novice to Professional, 2nd Edition By Richard Stones , Neil Matthew, Apress

---

CS-223 : Data structures Practicals and C++ Practicals

(semester 1)

Objective:-

1. Design and implement Data structures and related algorithms
2. Understand several ways of solving the same problem.

S.Y.B.Sc.(Computer Science) : Paper III : Data Structures using C Assignments		
No	Topic	Lectures
1	Sorting Algorithms – Bubble sort, Insertion	4
2	Recursive Sorting Algorithms – Quick sort , Merge Sort	4
3	Searching Method-Linear search, Binary search	4
4	Static/Dynamic stack implementation, infix to postfix, infix to prefix and evaluation of Postfix.	8
5	Static and Dynamic Queue Implementation – Linear Queue, Circular queue	8
6	Dynamic implementation of Singly Linked List, Doubly Linked List and Circular Linked List.	8
7	Polynomial addition (Using Linked list).	4
8	Binary Search Tree Traversal: Create, add, delete, and display nodes.	8
9	Adjacency matrix to adjacency list conversion, in degree, out degree	4
10	Graph: DFS, BFS.	4

<div style="border: 1px solid black; padding: 10px;">

CS-224:Database Practicals & Mini Project using Software Engineering techniques
(Semester 1)

</div>

Title: Database Assignments and Mini Project using Software Engineering techniques

Objective:-
- o Understanding the use of cursors, triggers, views and stored procedures
- o Understanding the steps of system analysis and design
- o Understanding Data requirements for a specific problem domain
- o Designing Data base as per the Data requirements
- o Designing queries as per the functional requirements

No	Topic	Lectures
1	Simple Queries	4
2	Nested Queries, using aggregate functions	4
3	Queries using Views	8
4	Queries using loops and conditional statements	8
5	Stored Function	12
6	Exception Handling	4
7	Cursors and Triggers	12

S.Y.B.Sc. Computer Science Theory Paper I
Semester II
CS 221 -Object Oriented Concepts using C++

Total Lectures: 48
Objective:-
    1. Acquire an understanding of basic object oriented concepts and the issues involved in effective class design
    2. Write C++ programs that use object oriented concepts such as information hiding, constructors, destructors, inheritance etc.

Prerequisites: Knowledge of C Programming Language

1. Object oriented concepts [2]
    1.1 Object oriented concepts
    1.2 Features, advantages and Applications of OOPS

2. Introduction to C++ [6]
    2.1 Data types, new operators and keywords, using namespace concept
    2.2 Simple C++ Program
    2.3 Introduction to Reference variables
    2.4 Usage of 'this' pointer
    2.5 Classes and Objects
    2.6 Access specifiers
    2.7 Defining Data members and Member functions
    2.8 Array of objects

3. Function in C++ [8]
    3.1 Call by reference, Return by reference
    3.2 Function overloading and default arguments
    3.3 Inline function
    3.4 Static class members
    3.5 Friend Concept – Function, Class

4. Constructors and destructor [4]
    4.1 Types of constructors
    4.2 Memory allocation (new and delete)
    4.3 Destructor

5. Operator overloading [4]
5.1 Overloading Unary and Binary operators
5.2 Overloading using friend function
5.3 Type casting and Type conversion

6. Inheritance [8]
6.1 Types of inheritance with examples
6.2 Constructors and destructor in derived classes
6.3 Virtual base classes, Virtual functions and Pure virtual function
6.4 Abstract base classes

7. Managing Input and Output using C++ [4]
7.1 Managing console I/O
7.2 C++ stream classes
7.3 Formatted and unformatted console I/O
7.4 Usage of manipulators

8. Working with files [6]
8.1 File operations – Text files, Binary files
8.2 File stream class and methods
8.3 File updation with random access
8.4 Overloading insertion and extraction operator

9. Templates [4]
9.1 Introduction to templates
9.2 Class templates, function templates and overloading of function templates
9.3 Templates with multiple parameters

10. Exception Handling in C++ [2]
10.1 try, catch and throw primitives

Reference Books: -

1. Object Oriented Programming with C++   by Robert Lafore
2. Object Oriented Programming with C++   by E. Balagurusamy
3. Object Oriented Modeling and Design by James Rumbough
4. The Complete Reference C++   by Herbert Schildt
5. Let us C++ by – YashwantKanitkar
6. Mastering C++   by Venugopal, T Ravishankar, RajkumarTHM Pub.
7.Trouble free C++ by HarimohanPande, ANE publication

S.Y.B.Sc.**Computer Science Theory paper-II**
Semester – II
CS - 222: Software Engineering

Total Lectures : 48

Objectives:-

- To teach basics of System Analysis and Design.
- To teach principles of Software Engineering
- To teach various process models used in practice
- To know about the system engineering and requirement engineering
- To build analysis model

Prerequisites: Basic knowledge of DBMS

1. **System Concepts   [5] (R1 : Chapter 1 & R3 : Chapter 1)**

    1.1   System Definition

    1.2   Characteristics of a System : Organization, Subsystem, Interaction, Interdependence, Integration, Central objective, Standards, Black-box

    1.3   Elements of a system : Outputs, Inputs, Processor(s), Control, Feedback, Environment, Boundaries, Interface.

    1.4   Types of Systems : Physical & Abstract Systems, Open & Closed Systems, Computer-based Systems (MIS : Management Information System & DSS : Decision Support System)

2. **Software and Software Engineering [5] (R2 : Chapter 1)**

    2.1   The Nature of Software

    2.1.1 Defining Software

    2.1.2 Software Application Domains

    2.1.3 Legacy Software

    2.2   Software Engineering

    2.3   The Software Process

2.4 Software Engineering Practice

    2.4.1 The Essence of Practice

    2.4.2 General Principles

2.5 Software Myths

3. System Development Life Cycle (SDLC) [8] (R3 : Chapter 1)

3.1 Introduction

3.2 Activities of SDLC

    3.2.1 Preliminary Investigation (Request Clarification, Feasibility Study, Request Approval)

    3.2.2 Determination of System Requirements

    3.2.3 Design of System

    3.2.4 Development of Software

    3.2.5 System Testing (Unit Testing, Integration testing, System Testing)

    3.2.6 System Implementation & Evaluation

    3.2.7 System Maintenance

4. Process Models   [6] (R2 : Chapter 2)

4.1 A Generic Process Model

4.2 Prescriptive Process Models

    4.2.1 The Waterfall Model

    4.2.2 Incremental Process Models

    4.2.3 Evolutionary Process Models

        4.2.3.1 Prototyping

        4.2.3.2 Spiral Model

    4.2.4 Concurrent Models

5. Requirements Engineering [8] (R2 : Chapter 5)

5.1 Introduction

5.2 Requirements Engineering Tasks

Reference Books :

R1 : System Analysis and Design (Second Edition) by Elias M. Awad, Galgotia Publications Pvt. Ltd.

R2 : Software Engineering : A Practitioner's Approach (Seventh Edition) by Roger S. Pressman, McGraw Hill International Edition.

R3 : Analysis and Design of Information Systems (Second Edition) by James A. Senn, McGraw Hill International Editions.

CS-223 : Data structures Practicals and C++ Practicals

(semester 2)

C++ Lab Assignments

1	Class , Object and methods implementation	4
2	Constructor: Copy Constructor, Default Constructor, Parameterized Constructor	4
3	Memory Allocation: new and delete operators , dynamic constructor	4
4	Inline function, friend function, default argument,	4
5	Function Overloading.	4
6	Operator overloading.	8
7	Inheritance: Single, multiple, multilevel, hierarchy, Constructor and destructor in derived class	12
8	File Handling: Updation of files using random access	4

CS-224: Database Practicals & Mini Project using Software Engineering techniques
(Semester 2)

No	Topic	Lectures
1	Problem definition , scope	8
2	Feasibility study	4
3	Gathering Data Requirements and Functional Requirement	12
4	ERD	4
5	Designing the normalized Database	8
6	Designing queries related to Functional requirements	12

University of Pune
S.Y.B.Sc.(Computer Science) Practical Examination
Lab Course I
(Data Structures Using C & Object Oriented Programming Concepts Using C++)

Duration: 3 hours   **Max. Marks: 80**

Q 1. Data Structures using C

1. Simple program based on searching / sorting / ADT of Stack, Queue, operations on linked list **[15]**

2. Program based on applications of stack/queue/linked list, trees / graph **[25]**

   OR

3. Program based on case study involving multiple data structures **[40]**

Q 2. Object Oriented Concepts and Programming in C++

1. Program based on different concepts in C++ **[30]**

   OR

2. Program based on different concepts in C++ **[30]**

3. Viva **[10]**