Maa Pateswari University, Balrampur

Syllabus
of
Computer Science
As Major for
B.Sc. Programme
in
Choice Based Credit System
based on
(CBCS)
National Education Policy-2020

(Common Minimum Syllabus for all U.P. State Universities and Colleges)

MAA PATESWARI UNIVERSITY, BALRAMPUR, UTTAR PRADESH

Structure of Syllabus for the Program: B. Sc. Subject: Computer Science

w. e. f. Session 2025-26

Structure of Syllabus Developed by			
Name of BOS Convener/ BOS Member	**Designation**	**Department**	**College/ University**
Prof. Jitendra Singh	Convener	Faculty of Physics	Shri Lal Bahadur Shastri Degree College, Gonda
Dr. Alok Shukla	Member	Physics	M. L. K. P. G. College, Balrampur
Sri Santosh Kumar Srivastava	Member	Physics	Shri Lal Bahadur Shastri Degree College, Gonda
Dr. Suraj Kumar Rai	Member	Physics	A. N. D. Kisan P. G. College, Babhnan, Gonda
Prof. Rakesh Tiwari	External Expert	Physics	Pt. Deen Dayal Upadhyay University, Gorakhpur
Prof. Om Prakash Yadav	External Expert	Physics	K.S. Saket P.G. College, Ayodhya
Dr Ram Kishor Singh	External Expert	Physics	Shivpati Degree College, Shohratgarh, Siddharthnagar

Year wise Structure of B.Sc. (Computer Science)

This course shall be offered in B.Sc. programme as a major subject along with two other major subjects and combinations available for the students of B.Sc. programme. Computer Science shall be one major subject along with other two major subjects which may be opted by the students as per the combinations offered by the University/College under CBCS.

Year	Sem.	Course Code	Paper Title	Theory/Practical	Credits
I	I	B070101T	Problem Solving using Computer	Theory	4
		B070102P	Software Lab using Python	Practical	2
	II	B070201T	Database Management Systems	Theory	4
		B070202P	Database Management Systems Lab	Practical	2
II	III	B070301T	Operating Systems	Theory	4
		B070302P	Operating Systems Lab	Practical	2
	IV	B070401T	Computer System Architecture	Theory	4
		B070402P	Computer System Architecture Lab	Practical	2
III	V	B070501T	Analysis of Algorithms and Data Structures	Theory	4
		B070503P	Lab on Algorithms and Data Structures with C++ based on Course code B070501T)	Practical	2
		B070502T	Soft Computing	Theory	4
		B070504P	Viva-voce Examination conducted by external examiner at the end of the Session based on Course code B070502T	Practical	2
	VI	B070601T	Data Communication and Computer Networks	Theory	4
		B070603P	Lab on Computer Networks based on Course code B070601T)	Practical	2
		B070602T	Cyber Security & Cyber Laws	Theory	4
		B070604P	Viva-voce Examination conducted by external examiner at the end of the Session based on Course code B070602T	Practical	2

Year wise Structure of B.Sc. (Computer Science)

Subject prerequisites To study the Computer Science, a student must have had the subject(s) computer science OR Mathematics in class/12th.	
Programme outcomes (POs): Students taking admission to B.Sc. program are expected to get prepared with following outcomes:	
PO 1	Explaining the basic scientific principles and methods.
PO 2	Inculcating scientific thinking and awareness among the student.
Programme specific outcomes (PSOs)	
PEO 1	To prepare students for career in computer science and its applications in professional career
PEO 2	To develop the student to cope up with the advancements in respective science field
PEO 3	The student will determine the appropriate level of technology for use in: a) experimental design and implementation, b) analysis of experimental data, and c) numerical and mathematical methods in problem solutions.
PEO 4	Investigate and apply mathematical problems and solutions in a variety of contexts related to science, technology, business and industry, and illustrate these solutions using symbolic, numeric, or graphical methods

Year wise Structure of B.Sc. for subject Computer Science

Type of Award	Subject: **Computer Science**										Total Credits of the subject
	Year	Sem.	Paper 1 Theory	credit	Paper 2 Theory	credit	Paper3 Practical	credit	Paper 4 Practical	credit	
Certificate in	1	I	Problem Solving using Computer	4	--		Software Lab using Python	2	Nil	Nil	6
		II	Database Management Systems	4	--		Database ManagementSystems Lab	2	Nil	Nil	6
Diploma in Computer	2	III	Operating Systems	4	--		Operating Systems Lab	2	Nil	Nil	6
		IV	Computer System Architecture	4	--		Computer System Architecture Lab	2	Nil	Nil	6
Bachelor of Science	3	V	Analysis of Algorithms andData Structures	4	Soft Computing	4	Lab on Algorithms and Data Structures with C++ based on Course code B070501T)	2	Viva-voce Examination conducted by external examiner at the end of the Session based on Course code B070502T	2	12
		VI	Data Communication and Computer Networks	4	Cyber Security & Cyber Laws	4	Lab on Computer Networks based on Course code B070601T)	2	Viva-voce Examination conducted by external examiner at the end of the Session based on Course code B070602T	2	12
Total Credits:											**48**

Syllabus for B.Sc.: Subject: Computer Science

Programme/Class: **Certificate**	Year: **First**	Semester: **First**
Subject: **Computer Science**		
Course Code: B070101T	Course Title: **Problem Solving using Computer**	

Course outcomes:

CO1: Understand hardware components of computer system such as memory system organization, input/output devices, aware of software components of computer system, and windows operating system concepts.

CO 2: Develops basic understanding of computers, the concept of algorithm and algorithmic thinking.

CO3: Develops the ability to analyze a problem, develop an algorithm to solve it.

CO4: Develops the use of the Python programming language to implement various algorithms, and develops the basic concepts and terminology of programming in general.

CO5: Introduces the more advanced features of the Python language

Credits: **4**	**Core Compulsory**
Max. Marks: **25+75**	Min. Passing Marks: As per UGC/University CBCS norm

Total No. of Lectures-Tutorials-Practical (in hours per week): **4-0-0**

Unit	Topic	No. of Lectures
I	**Computer Fundamentals:** Introduction to Computers: Characteristics of Computers, Uses of computers, Types and generations of Computers.	7
II	**Basic Computer Organization** - Units of a computer, CPU, ALU, memory hierarchy, registers, I/O devices. **Planning the Computer Program:** Concept of problem solving, Problem definition, Program design, Debugging, Types of errors in programming, Documentation.	8
III	**Techniques of Problem Solving:** Flowcharting, decision table, algorithms, Structured programming concepts, Programming methodologies viz. top-down and bottom-up programming.	7
IV	**Overview of Programming:** Structure of a Python Program, Elements of Python	8

V	**Introduction to Python:** Python Interpreter, Using Python as calculator, Python shell, Indentation. Atoms, Identifiers and keywords, Literals, Strings, Operators (Arithmetic operator, Relational operator, Logical or Boolean operator, Assignment, Operator, Ternary operator, Bit wise operator, Increment or Decrement operator).	8
VI	**Creating Python Programs:** Input and Output Statements, Control statements (Looping- while Loop, for Loop, Loop Control, Conditional Statement- if...else, Difference between break, continue and pass).	7
VII	**Structures:** Numbers, Strings, Lists, Tuples, Dictionary, Date & Time, Modules, Defining Functions, Exit function, default arguments.	7
VIII	**Introduction to Advanced Python:** Objects and Classes, Inheritance, Regular Expressions, Event Driven Programming, GUI Programming.	8

Suggested Readings:

1. P. K. Sinha & Priti Sinha , "Computer Fundamentals", BPB Publications, 2007.
2. Dr. Anita Goel, Computer Fundamentals, Pearson Education, 2010.
3. T. Budd, Exploring Python, TMH, 1st Ed, 2011
4. Python Tutorial/Documentation www.python.or 2010
5. Allen Downey, Jeffrey Elkner, Chris Meyers , How to think like a computerscientist : learning with Python , Freely available online.2012

Suggestive digital platforms web links-

https://www.pearsoned.co.in/prc/book/anita-goel-computer-fundamentals-1e-1/9788131733097

http://docs.python.org/3/tutorial/index.html
http://interactivepython.org/courselib/static/pythonds
http://www.ibiblio.org/g2swap/byteofpython/read/

Programme/Class: **Certificate**	Year: **First**	Semester: **First**
Subject: Computer Science		
Course Code: B070102P	**Course Title: Software Lab using Python**	

Course outcomes:

1. To learn and understand Python programming basics.
2. To learn and understand python looping, control statements and string manipulations.
3. Students should be made familiar with the concepts of GUI controls and designing GUI applications.
4. To learn and know the concepts of file handling, exception handling and database connectivity.

Credits**: 2**	Max. Marks: **50**	Min. Passing Marks: As per UGC/University CBCS norm
Total No. of Lectures-Tutorials-Practical (in hours per week): **0-0-4**		

Suggested Readings:

1. Allen B. Downey, "Think Python: How to Think Like a Computer Scientist", 2nd edition, Updated for Python 3, Shroff/O'Reilly Publishers, 2016 (http://greenteapress.com/wp/thinkpython/)
2. Guido van Rossum and Fred L. Drake Jr, "An Introduction to Python – Revised and updated for Python 3.2, Network Theory Ltd., 2011.
3. Charles Dierbach, "Introduction to Computer Science using Python: A Computational Problem-Solving Focus, Wiley India Edition, 2013.
4. John V Guttag, "Introduction to Computation and Programming Using Python'', Revised and expanded Edition, MIT Press , 2013
5. Kenneth A. Lambert, "Fundamentals of Python: First Programs", CENGAGE Learning, 2012.

Section: A (Simple programs)

1. Write a menu driven program to convert the given temperature from Fahrenheit to Celsius and vice versa depending upon user's choice.
2. WAP to calculate total marks, percentage and grade of a student. Marks obtained in each of the three subjects are to be input by the user. Assign grades according to the following criteria :

   Grade A: Percentage >=80
   Grade B: Percentage>=70 and <80 Grade C: Percentage>=60 and <70 Grade D: Percentage>=40 and <60 Grade E: Percentage<40
3. Write a menu-driven program, using user-defined functions to find the area of rectangle, square, circle and triangle by accepting suitable input parameters from user.

4. WAP to display the first n terms of Fibonacci series.
5. WAP to find factorial of the given number.
6. WAP to find sum of the following series for n terms: 1 – 2/2! + 3/3!--------- n/n!
7. WAP to calculate the sum and product of two compatible matrices.

Section: B (Visual Python)

All the programs should be written using user defined functions, wherever possible.

1. Write a menu-driven program to create mathematical 3D objects
   I. curve
   II. sphere
   III. cone
   IV. arrow
   V. ring
   VI. Cylinder.

2. WAP to read n integers and display them as a histogram.
3. WAP to display sine, cosine, polynomial and exponential curves.
4. WAP to plot a graph of people with pulse rate p vs. height h. The values of p and h are to be entered by the user.
5. WAP to calculate the mass m in a chemical reaction. The mass m (in gms) disintegrates according to the formula m=60/(t+2), where t is the time in hours. Sketch a graph for t vs. m, where t>=0.
6. A population of 1000 bacteria is introduced into a nutrient medium. The population p grows as follows:
   $P(t) = (15000(1+t))/(15+ e)$
   where the time t is measured in hours. WAP to determine the size of the population at given time t and plot a graph for P vs t for the specified time interval.

7. Input initial velocity and acceleration, and plot the following graphs depicting equations of motion:
   I. velocity wrt time (v=u+at)
   II. distance wrt time (s=u*t+0.5*a*t*t)
   III. distance wrt velocity (s=(v*v-u*u)/2*a)

Programme/Class: **Certificate**	Year: **First**	Semester: **Second**
Subject: **Computer Science**		
Course Code: B070201T	Course Title: **Database Management System**	

Course outcomes:

After the completion of the course the students will be able to:

1. Understands the basic concepts of data base management systems.
2. Design E-R diagrams for real world applications.
3. Formulate relational algebraic expressions using relational data models and languages.
4. Apply normalization transaction properties and concurrency control to design database.
5. Analyze the security algorithms for database protection.

Credits: **4**	**Core Compulsory**
Max. Marks: **25+75**	Min. Passing Marks: As per UGC/University CBCS norm

Total No. of Lectures-Tutorials-Practical (in hours per week): **4-0-0**

Unit	Topic	No. of Lectures
I	**Introduction:** Database System Concepts, File system vs. database system, Database system architecture, Data models and their types, Data base scheme and instances, Data independence, Database Languages and Interfaces.	7
II	**Data Modeling Concepts** ER model concepts: Notations for ER diagram, Extended E-R diagram, Extended E-R model, E-R model design issues, constraints, and keys:Weak entity set strong entity set, Relationships of higher degree.	8
III	**Relational model concepts:** code rules, constraints, RelationalAlgebra operations, Extended relational algebra operations, Relational Calculus, Tuple and Domain relational calculus.	7
IV	**Database Design** Functional dependencies, Normal forms, First, second, and third normal forms, BCNF, Multi-valued dependencies and Fourth Normalform, Join Dependencies and Fifth Normal form.	8
V	**Transaction, Query Processing** Transaction and system concepts: transaction states, ACID properties of transactions, concurrent execution schedules and Recoverability,Serializability of schedules. Query Processing and Optimization: Measures of Query cost, Cost, Evaluation of expression. Optimization: Transformation of relational expression, Choice of evaluation plan.	7
VI	**Concurrency Control**: Concurrency Control Techniques: Two phase Locking Techniques for Concurrency Control; Time stamping in Concurrency control.	8

VII	**Introduction to SQL** Basic Structure of SQL Query, Set operators, SELECT, UNION, INTERSECT, and EXCEPT, Nested queries, Aggregate function, Null values, Derived Relations, Modification of the Database, Joined relations and up-dates in SQL.	8

VIII	**Database Security** Importance of data, Threats and risks, Users and database privileges,Access Control, Security for Internet Applications, Role of Database Administrator.	7

Suggested Readings:

1. Henry F. Korth and Abraham Silberschatz, "Database System Concepts," Second Edition,McGraw Hill, 1991.
2. AtulKahate, "Introduction to Database Management Systems," Pearson India, 2004.
3. Raghu Ramakrishnan and Johannes Gehrike, "Database Management Systems," ThirdMcGraw Hill, Edition, 2003.
4. R. Elmasri, S.B. Navathe Database Systems Models, Languages, Design and applicationProgramming, 6 Edition, Pearson Education,2013.
5. A. Silberschatz, H.F. Korth, S. Sudarshan, Database System Concepts 6th Edition, McGraw Hill, 2010.

Programme/Class: **Certificate**	Year: **First**	Semester: **Second**
Subject: Computer Science		
Course Code: B070202P	Course Title: Database Management Systems Lab	

Course outcomes:

Ability to:

1. Understand, analyze and apply common SQL statements including DDL, DML and DCL statements to perform different operations.
2. Design and implement a database schema for a given problem.
3. Do connectivity of PHP and MySQL to develop applications.

Credits**: 2**	Max. Marks: **50**	Min. Passing Marks: As per UGC/University CBCS norm

Total No. of Lectures-Tutorials-Practical (in hours per week): **0-0-4**

Suggested Readings:

1. Paul DuBois, "MySQL Cookbook: Solutions for Database Developers and Administrators," Third Edition, O'Reilly Media, 2014.
2. Frank M. Kromann, "Beginning PHP and MySQL: From Novice to Professional," Fifth Edition, Apress, 2018.
3. Joel Murach and Ray Harris, "Murach's PHP and MySQL," First Edition, Mike Murach & Associates, 2010.
4. Luke Welling, Laura Thomson, "PHP and MySQL Web Development," Fourth Edition, Addison-Wesley, 2008.

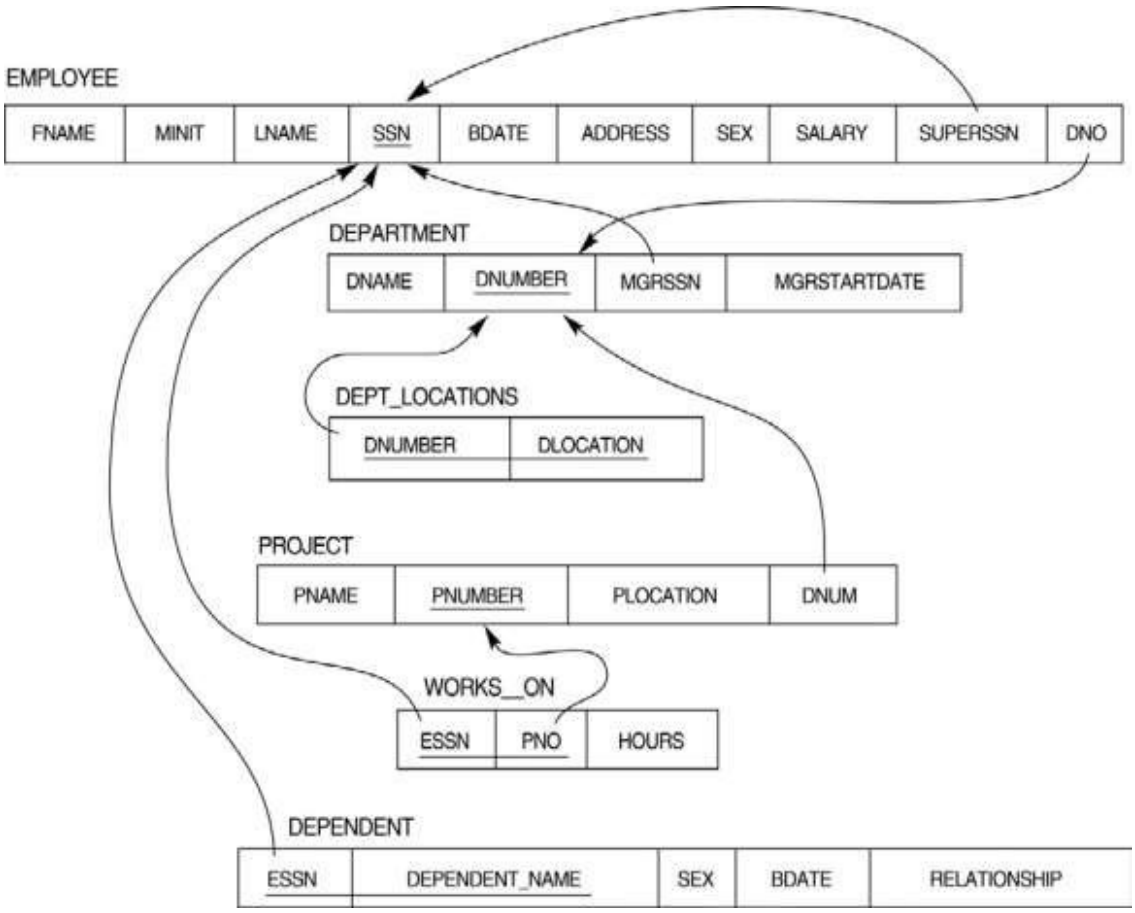
Software Lab based on Database Management Systems

Note: PHP/MySQL may be used

List of Experiments

1. Creation of databases and execution of SQL queries.
2. Creation of Tables using MySQL: Data types, Creating Tables (along with Primary and Foreign keys), Altering Tables and Dropping Tables.
3. Practicing DML commands- Insert, Select, Update, Delete.
4. Practicing Queries using ANY, ALL, IN, EXISTS, NOT, EXISTS, UNION, INTERSECT, and CONSTRAINTS, etc.
5. Practice Queries using COUNT, SUM, AVG, MAX, MIN, GROUP BY, HAVING, VIEWS Creation and Dropping.
6. Use of COMMIT, ROLLBACK and SAVEPOINT.
7. Practicing on Triggers - creation of trigger, Insertion using trigger, Deletion using trigger, Updating using trigger.
8. To remove the redundancies and anomalies in the above relational tables, Normalize up to Third Normal Form.

Relational Database Schema - COMPANY



Questions to be performed on above schema

1. Create tables with relevant foreign key constraints
2. Populate the tables with data
3. Perform the following queries on the database :
    1. Display all the details of all employees working in the company.
    2. Display ssn, lname, fname, address of employees who work in department no 7.
    3. Retrieve the birthdate and address of the employee whose name is 'Franklin T. Wong'
    4. Retrieve the name and salary of every employee
    5. Retrieve all distinct salary values

6. Retrieve all employee names whose address is in ‘Bellaire’
7. Retrieve all employees who were born during the 1950s
8. Retrieve all employees in department 5 whose salary is between 50,000 and 60,000(inclusive)
9. Retrieve the names of all employees who do not have supervisors
10. Retrieve SSN and department name for all employees
11. Retrieve the name and address of all employees who work for the 'Research' department
12. For every project located in 'Stafford', list the project number, the controlling department number, and the department manager's last name, address, and birthdate.
13. For each employee, retrieve the employee's name, and the name of his or her immediate supervisor.
14. Retrieve all combinations of Employee Name and Department Name
15. Make a list of all project numbers for projects that involve an employee whose last name is 'Narayan’ either as a worker or as a manager of the department that controls the project.
16. Increase the salary of all employees working on the 'ProductX' project by 15%. Retrieve employee name and increased salary of these employees.
17. Retrieve a list of employees and the project name each works in, ordered by the employee's department, and within each department ordered alphabetically by employee first name.
18. Select the names of employees whose salary does not match with salary of any employee in department 10.
19. Retrieve the name of each employee who has a dependent with the same first name and same sex as the employee.
20. Retrieve the employee numbers of all employees who work on project located in Bellaire, Houston, or Stafford.
21. Find the sum of the salaries of all employees, the maximum salary, the minimum salary, and the average salary. Display with proper headings.
22. Find the sum of the salaries and number of employees of all employees of the ‘Marketing’ department, as well as the maximum salary, the minimum salary, and the average salary in this department.
23. Select the names of employees whose salary is greater than the average salary of all employees in department 10.
24. For each department, retrieve the department number, the number of employees

in the department, and their average salary.

25. For each project, retrieve the project number, the project name, and the number of employees who work on that project.

26. Change the location and controlling department number for all projects having more than 5 employees to 'Bellaire' and 6 respectively.
27. For each department having more than 10 employees, retrieve the department no, no of employees drawing more than 40,000 as salary.
28. Insert a record in Project table which violates referential integrity constraint with respect to Department number. Now remove the violation by making necessary insertion in the Department table.

29. Delete all dependents of employee whose ssn is '123456789'.

30. Delete an employee from Employee table with ssn = '12345'(make sure that this employee has some dependents, is working on some project, is a manager of some department and is supervising some employees). Check and display the cascading effect on Dependent and Works on table. In Department table MGRSSN should be set to default value and in Employee table SUPERSSN should be set to NULL

31. Perform a query using alter command to drop/add field and a constraint in Employee table.

Note: The instructors should design detailed experiments based on above suggested experiments.

Programme/Class: **Diploma**	Year: **Second**	Semester: **Third**
Subject: **Computer Science**		
Course Code: B070301T	Course Title: **Operating System**	

Course outcomes:

After the completion of the course the students will be able:

1. Understand role, responsibilities, features, and design of operating system.
2. Analyze memory management schemes and process scheduling algorithms.
3. Apply process synchronization techniques to formulate solution for critical section problems.
4. Illustrate concept of disk scheduling.
5. Evaluate process deadlock handling techniques.

Credits: **4**	**Core Compulsory**
Max. Marks: **25+75**	Min. Passing Marks: As per UGC/University CBCS norm

Total No. of Lectures-Tutorials-Practical (in hours per week): **4-0-0**

Unit	Topic	No. of Lectures
I	**Introduction** Operating system and functions, Classification of Operating systems: Batch, Interactive, Time sharing, Real Time System, Multiprocessor Systems, Multiuser Systems, Multithreaded Systems, Operating System Structure, System Components, Operating System Services, Kernels, Monolithic and Microkernel Systems.	7
II	**Process Management** Process Concept, Process States, Process Synchronization, Critical Section, Mutual Exclusion, Classical Synchronization Problems, Process Scheduling, Process States, Process Transitions, Scheduling Algorithms Interprocess Communication, Threads and their management, Security Issues.	8
III	**CPU Scheduling** Scheduling Concepts, Techniques of Scheduling, Preemptive and Non-Preemptive Scheduling: First-Come-First-Serve, Shortest Request Next, Highest Response Ration Next, Round Robin, Least Complete Next, Shortest Time to Go, Long, Medium, Short Scheduling, Priority Scheduling. Deadlock: System model, Deadlock characterization, Prevention, Avoidance and detection, Recovery from deadlock.	8
IV	**Memory Management** Memory allocation, Relocation, Protection, Sharing, Paging, Segmentation, Virtual Memory, Demand Paging,Page Replacement Algorithms, Thrashing.	7
V	**I/O Management and Disk Scheduling** I/O devices, and I/O subsystems, I/O buffering, Disk storage and disk scheduling, RAID.	8

VI	**File System:** File concept, File organization and access mechanism, Filedirectories, and File sharing, File system implementation issues, File system protection and security.	7
VII	**Shell introduction and Shell Scripting:** What is shell and various type of shell, Various editors present in linux, Different modes of operation in vi editor,	7
VIII	What is shell script, Writing and executing the shell script, Shell variable (user defined and system variables) System calls, Using system calls, Pipes and Filters, Decision making in Shell Scripts (If else, switch), Loops in shell, Functions, Utility programs (cut, paste, join, tr , uniq utilities), Pattern matching utility (grep)	8

Suggested Readings:

1. Andrew S. Tanenbaum and Herbert Bos,"Modern Operating Systems," Fourth Edition, Pearson, 2014.
2. Abraham Silberschatz, Greg Gagne, and Peter B. Galvin, "Operating System Concepts," Tenth Edition, Wiley, 2018.
3. William Stallings, "Operating Systems: Internals and Design Principles," Seventh Edition, Prentice Hall, 2011.
4. Dhanjay Dhamdhere, "Operating Systems," First Edition, McGraw-Hill, 2008
5.

Programme/Class: **Diploma**	Year: **Second**	Semester: **Third**
Subject: Computer Science		
Course Code: B070302P	Course Title: Operating Systems Lab	

Course outcomes:

Ability to:

1. Use of Linux operating system and able to write shell programs.
2. Simulate and demonstrate the concepts of operating systems.

Credits**: 2**	Max. Marks: **50**	Min. Passing Marks: As per UGC/University CBCS norm
Total No. of Lectures-Tutorials-Practical (in hours per week): **0-0-4**		

Suggested Readings:

1. Sumitabh Das, "Your Unix/Linux: The Ultimate Guide," McGraw Hill, 2012.
2. Richard Blum and Christine Bresnahan, "Linux Command Line and Shell Scripting Bible," Wiley, 2015.
3. Stroustrup, Bjarne, Programming: Principles and Practice Using C++, Addison Wesley, USA, 2014, 2nd ed.
4. E Balagurusamy, Object Oriented Programming with C++, McGraw Hill Education (India) Pvt. Ltd., India, 2013, 6th ed.

Lab on Operating Systems

Note: Following exercises can be performed using Linux or Unix

1. Usage of following commands:
   ls, pwd, tty, cat, who, who am I, rm, mkdir, rmdir, touch, cd.
2. Usage of following commands:
   cal, cat(append), cat(concatenate), mv, cp, man, date.
3. Usage of following
   commands: chmod, grep,
   tput (clear, highlight), bc.
4. Write a shell script to check if the number entered at the command line is prime or not.
5. Write a shell script to modify "cal" command to display calendars of the specified months.
6. Write a shell script to modify "cal" command to display calendars of the specified range of months.
7. Write a shell script to accept a login name. If not a valid login name display message – "Entered login name is invalid".
8. Write a shell script to display date in the mm/dd/yy format.
9. Write a shell script to display on the screen sorted output of "who" command along with the total number of users .
10. Write a shell script to display the multiplication table any number,

11. Write a shell script to compare two files and if found equal asks the user to delete the duplicate file.
12. Write a shell script to check whether the file have all the permissions or not.
13. Simulate FCFS CPU scheduling algorithm in C++.
14. Simulate SJF CPU scheduling algorithm in C++.
15. Simulate Priority CPU scheduling algorithm in C++.
16. Simulate Round Robin CPU scheduling algorithm in C++.
17. Simulate FIFO page replacement algorithm in C++.
18. Simulate LRU page replacement algorithm in C++.

Programme/Class: **Diploma**	Year: **Second**	Semester: **Fourth**
Subject: **Computer Science**		
Course Code: B070401T	Course Title: **Computer System Architecture**	

Course outcomes:

The student will be able to understand the basic arithmetic of a Computer System; how the data is represented, how the various operation are performed on the data, the basic circuits to perform these operations, how instructions are formatted and how these instructions are executed to accomplish a particular operation. Student can also learn the organization of the peripheral devices, the interface between these devices to the system. Student can also understand the architecture of a basic computer, its registers, bus system and the interaction flow among them.

Credits: **4**	**Core Compulsory**
Max. Marks: **25+75**	Min. Passing Marks: As per UGC/University CBCS norm

Total No. of Lectures-Tutorials-Practical (in hours per week): **4-0-0**

Unit	Topic	No. of Lectures
I	**Data Representation and basic Computer Arithmetic:** Number systems, complements, fixed and floating point representation, character representation, addition, subtraction, magnitude comparison.	7
II	**Logic gates and circuits:** logic gates, boolean algebra, combinational circuits, circuit simplification, introduction to flip-flops and sequential circuits, decoders, multiplexers, registers, counters.	8
III	**Basic Computer Organization and Design:** Computer registers, bus system, instruction set, timing and control, instruction cycle, memory reference, input-output and interrupt.	7
IV	**Central Processing Unit:** Register organization, arithmetic and logical micro-operations, stack organization, Hardwired vs. micro programmed control. Pipeline control: Instruction pipelines, pipeline performance, super scalar processing, Pipelining, RISC & CISC	8
V	**Programming the Basic Computer:** Instruction formats, addressing modes, instruction codes, assembly language	7
VI	**Memory Organization:** Memory device characteristics, random access memories, serial access memories, Multilevel memories, address translation, memory allocation, Main features, address mapping, structure versus performance.	8
VII	**Input-output Organization:** Peripheral devices, I/O interface, Modes of data transfer: Programmed, Interrupt Driven and Direct Memory Access.	8

VIII	**Parallel processing:** Processor-level parallelism, multiprocessor architecture	7

Suggested Readings:

1. M. Mano, "Computer System Architecture", Pearson Education, New Jersey, 2017,Third Edition.
2. W. Stallings, "Computer Organization and Architecture Designing for Performance",Prentice Hall of India, 2015, Tenth Edition.
3. M. Mano, "Digital Design", Pearson Education, New Jersey, 2018, Sixth Edition.

Suggested equivalent online courses:

1. https://onlinecourses.nptel.ac.in/noc20_cs64;

Programme/Class: **Diploma**	Year: **Second**	Semester: **Fourth**
Subject: **Computer Science**		
Course Code: B070402P	Course Title: Computer System Architecture Lab	
Course outcomes: An ability to understand: **CO1** The functions of various hardware components and their building blocks **CO2** Boolean algebraic expressions to digital design **CO3** And implementation of different sequential and Combinational circuits **CO4** computer buses and input/output peripherals **CO5** memory hierarchy and design of primary memory		
Credits: **2**	Max. Marks: **25**	Min. Passing Marks: As per UGC/University CBCS norm
Total No. of Lectures-Tutorials-Practical (in hours per week): **0-0-4**		

Practical: 60 Lab Periods

Memory 4096 words 16 bits per word	Instruction format 0 3 4 15 Opcode | Address

Basic Computer Instructions

Memory Reference	Register Reference	Input-Output

1. Create a machine based on the following architecture:
   Register Set

IR	DR	AC	AR	PC	FGI	FGO	S	I	E
0 15	0 15	0 15	011	011	1 Bit	1 Bit	1 Bit	1 bit	1 Bit

Symbol	Hex		Symbol	Hex	Symbol	Hex
AND	0xxx		CLA	E800	INP	F800
ADD	2xxx		CLE	E400	OUT	F400
ISZ	Cxxx		INC	E020		

AND_I	1xxx	Indirect Addressing	SPA	E010		
ADD_I	3xxx		SNA	E008		
LDA_I	5xxx		SZA	E004		
STA_I	7xxx		SZE	E002		
BUN_I	9xxx		HLT	E001		
BSA_I	Bxxx					
ISZ_I	Dxxx					

Refer to Chapter-5 of Morris Mano for description of instructions.

ii) Create the micro operations and associate with instructions as given in the chapter (except interrupts). Design the register set, memory and the instruction set. Use this machine for the assignments of this section.

iii) Create a Fetch routine of the instruction cycle.

iv) Simulate the machine to determine the contents of AC, E, PC, AR and IR registers in hexadecimal after the execution of each of following register reference instructions:

a. CLA
b. CLE
c. CMA
d. CME
e. CIR
f. CIL
g. INC
h. SPA
i. SNA
j. SZA
k. SZE
l. HLT

Initialize the contents of AC to $(A937)_{16}$, that of PC to $(022)_{16}$ and E to 1.

5. Simulate the machine for the following memory-reference instructions with I= 0 and address part = 082. The instruction to be stored at address 022 in RAM. Initialize the memory word at address 082 with the operand B8F2 and AC with A937. Determine the contents of AC, DR, PC, AR and IR in hexadecimal after the execution.

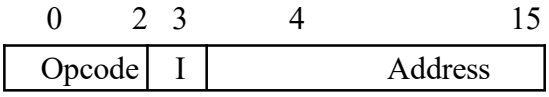
a. ADD
b. AND
c. LDA
d. STA
e. BUN
f. BSA
g. ISZ

6. Simulate the machine for the memory-reference instructions referred in above question with I= 1 and address part = 082. The instruction to be stored at address 026 in RAM. Initialize the memory word at address 082 with the value 298. Initialize the memory word at address 298 with operand B8F2 and AC with A937. Determine the contents of AC, DR, PC, AR and IR in hexadecimal after the

execution.

7. Modify the machine created in Practical 1 according to the following instruction format:

Instruction format



a. The instruction format contains a 3-bit opcode, a 1-bit addressing mode and a 12-bit address. There are only two addressing modes, I = 0 (direct addressing) and I = 1 (indirect addressing).

b. Create a new register I of 1 bit.

c. Create two new microinstructions as follows :

   i. Check the opcode of instruction to determine type of instruction (Memory Reference/Register Reference/Input-Output) and then jump accordingly.

   ii. Check the I bit to determine the addressing mode and then jumpaccordingly.

Programme/Class: **Bachelor in Science**	Year: **Third**	Semester: **Fifth**
Subject: **Computer Science**		
Course Code: B070501T	Course Title: **Analysis of Algorithm and Data Structures**	

Course outcomes:

CO 1: Understand that various problem solving categories exist such as; iterative technique, divide and conquer, dynamic programming, greedy algorithms, and understand various searching and sorting algorithms

CO 2: Employ a deep knowledge of various data structures when constructing a program..

CO 3: Design and construct simple object-oriented software with an appreciation for data abstraction and information hiding.

CO 4: Effectively use software development tools including libraries, compilers, editors, linkers and debuggers to write and troubleshoot programs.

Credits: **4**	**Core Compulsory**
Max. Marks: **25+75**	Min. Passing Marks: As per UGC/University CBCS norm

Total No. of Lectures-Tutorials-Practical (in hours per week): **4-0-0**

Unit	Topic	No. of Lectures
I	**Introduction**: Basic Design and Analysis techniques of Algorithms, Correctness of Algorithm, Algorithm Design Techniques: Iterative techniques, Divide and Conquer, Dynamic Programming, Greedy Algorithms.	7
II	**Sorting Techniques**: Elementary sorting techniques-Bubble Sort, Insertion Sort, Merge Sort, Advanced Sorting techniques-Heap Sort, Quick Sort, Sorting in Linear Time-Bucket Sort, Radix Sort and Count Sort	8
III	**Searching Techniques and Complexity Analysis:** Linearand Binary search, Medians & Order Statistics.	7
IV	**Arrays** Arrays: Single and Multi-dimensional Arrays,Sparse Matrices;	7
V	**Stacks and Queues :** Implementing stack using array and linked list, Prefix, Infix and Postfix expressions, Utility and conversion of these expressions from one to another; Array andLinked representation of Queue, De-queue, Priority Queues	8
VI	**Linked Lists:** Singly, Doubly and Circular Lists, representationof Stack and Queue as Linked Lists.	8
VII	**Recursion**: Developing Recursive Definition of Simple Problems and their implementation; Advantages and Limitations ofRecursion;	7
VIII	**Trees**: Introduction to Tree as a data structure; Binary Trees, Binary Search Tree, (Creation, and Traversals of Binary Search Trees)	8

Suggested Readings:

1. Cormen T.H., Leiserson Charles E., Rivest Ronald L., Stein Clifford, Introduction to Algorithms, PHI Learning Pvt. Ltd., 2009, 3rd Edition.
2. Basse Sara & A.V. Gelder, Computer Algorithm: Introduction to Design and Analysis, Pearson, 2000, 3rd Edition.
3. Drozdek Adam, "Data Structures and algorithm in C++", Cengage Learning, 2012, Third Edition.
4. Tenenbaum Aaron M., Augenstein Moshe J., Langsam Yedidyah, "Data Structures Using C and C++, PHI, 2009, Second edition.
5. Kruse Robert L., "Data Structures and Program Design in C++", Pearson.
6. **Suggestive digital platforms web links or online course-**

   https://www.oercommons.org/authoring/14873-data-structure/view

   https://www.oercommons.org/courses/data-structure-and-algorithms

   https://onlinecourses.swayam2.ac.in/cec19_cs04/preview **(online course)**

Suggested equivalent online courses:

1 https://nptel.ac.in/courses/106/102/106102064/

Programme/Class: **Bachelor In Science**	Year: **Third**	Semester: **Fifth**
Subject: **Computer Science**		
Course Code: B070502T	Course Title: **Soft Computing**	

Course outcomes:

Upon the completion of this course the student will have the knowledge of soft computing concepts and he can apply them for practical applications. He would be able to choose and design suitable Neural Network for real time problems. He can appropriately use fuzzy rules and reasoning to develop decision making and expert systems. He would know the importance of optimization techniques and genetic programming.

Credits: **4**	**Core Compulsory**
Max. Marks: **25+75**	Min. Passing Marks: As per UGC/University CBCS norm

Total No. of Lectures-Tutorials-Practical (in hours per week): **4-0-0**

Unit	Topic	No. of Lectures
I	**Introduction To Neural Networks**: Neural Networks Neuron, Nerve Structure And Synapse, Artificial Neuron And Its Model, Activation Functions.	7
II	**Neural Network Architecture**: Single Layer And Multilayer Feed Forward Networks, Recurrent Networks. Perception And Convergence Rule.Supervised Learning Network& Unsupervised Learning Network.	8
III	**Back Propogation Networks-I**: Perceptron Model, Solution, Single Layer, Multilayer Perception Model;	7
IV	**Back Propogation Networks**-II: Back Propogation Learning Methods, Effect Of Learning Rule Co-Efficient ;Back Propagation Algorithm, Applications.	8
V	**Fuzzy Logic Introduction-I**: Basic Concepts Of Fuzzy Logic, Fuzzy Sets And Crisp Sets, Fuzzy Set Theory And Operations, Properties Of Fuzzy Sets	7
VI	**Fuzzy Logic Introduction-II:** Fuzzy And Crisp Relations, Fuzzy To Crisp Conversion, Membership Functions, Interference In Fuzzy Logic, Fuzzy If-Then Rules, Fuzzyfications&Defuzzificataions.	8
VII	**Genetic Algorithm-I**: Basic Concepts, Working Principle, Procedures Of GA, Flow Chart Of GA	7

VIII	**Genetic Algorithm-II:** Genetic Representations, (Encoding), Genetic Operators, Mutation, Generational Cycle.	8

Suggested Readings:

1. S. Rajsekaran& G.A. VijayalakshmiPai, "Neural Networks,Fuzzy Logic and Genetic Algorithm:Synthesis and Applications" Prentice Hall of India,2003
2. Anderson, James, "Introduction to Neural Networks", PHI Publication, Delhi, India
3. N.P.Padhy,"Artificial Intelligence and Intelligent Systems" Oxford University Press, USA, 2005.
4. Simon Haykin,"Neural Netowrks and Learning Machines "Prentice Hall of India, 2005, Third Edition.

Suggested equivalent online courses:
https://www.classcentral.com/course/swayam-introduction-to-soft-computing-10053

Further Suggestions:
List of Programs in Soft Computing with Python:

Programme/Class: **Bachelor of Science**	Year: **Third**	Semester: **Fifth**
Subject: **Computer Science**		
Course Code: B070503P	Course Title: Lab on Algorithms and Data Structures with C++ based on Course code B070501T)	
Course outcomes: **CO 1:** Optimize the solution with respect to time complexity & memory usage **CO 2:** Assess how the choice of data structures and algorithm design methods impacts the performance of programs. **CO 3:** Choose the appropriate data structure and algorithm design method for a specified application. **CO 4:** Solve problems using data structures such as linear lists, stacks, queues, binary trees, binary search trees and writing programs for these solutions		
Credits: **2**	Max. Marks: **50**	Min. Passing Marks: As per UGC/University CBCS norm
Total No. of Lectures-Tutorials-Practical (in hours per week): **0-0-4**		

Practical List of on Analysis of Algorithms and Data Structures with C++:

1. Write a program that uses functions to perform the following:
   a) Create a singly linked list of integers.
   b) Delete a given integer from the above linked list.
   c) Display the contents of the above list after deletion.

2. Write a program that uses functions to perform the following:
   a) Create a doubly linked list of integers.
   b) Delete a given integer from the above doubly linked list.
   c) Display the contents of the above list after deletion.

3. Write a program that uses stack operations to convert a given infix expression into its postfix Equivalent, implement the stack using an array.

4. Write program to implement a double ended queue using
   i) array and
   ii) doubly linked list respectively.

5. Write a program that uses functions to perform the following:
   a) Create a binary search tree of characters.
   b) Traverse the above Binary search tree recursively in Postorder.

6. Write a program that uses functions to perform the following:
   a) Create a binary search tree of integers.
   b) Traverse the above Binary search tree non recursively in inorder.

7. Write program for implementing the following sorting methods to arrange a list of integers in ascending order:
    a) Insertion sort
    b) Merge
8. Write program for implementing the following sorting methods to arrange a list of integers in ascending order:
    a) Quick sort
    b) Selection sort

9. Write program to implement Insertion Sort (The program should report the number of comparisons)

10. Write program implement Merge Sort(The program should report the number of comparisons)

11. Write program implement Heap Sort (The program should report the number of comparisons)

12. Write program implement Randomized Quick sort (The program should report the number of comparisons)

13. Write program for creation and traversal of Binary Search Tree.

Programme/Class: **Bachelor of Science**	Year: **Third**	Semester: **Fifth**
Subject: **Computer Science**		
Course Code: B070504P	Course Title: Viva-voce examination conducted by external examiner at the end of the Session based on Course code B070502T	

Course outcomes:

CO 1: Understand the soft computing techniques and their applications.
CO 2: Understand the various neural network architectures.
CO 3: Understand the fuzzy systems.
CO 4: Understand the genetic algorithm concepts and their applications.

Credits: **2**	Max. Marks: **50**	Min. Passing Marks: As per UGC/University CBCS norm
Total No. of Lectures-Tutorials-Practical (in hours per week): **0-0-4**		

*Viva-voce will be based on Course code B070502T (**Soft Computing**). Teacher-in-Charge shall design some problems on Neural Network for real time problems, fuzzy rules and reasoning to develop decision making and expert systems and importance of optimization techniques and genetic programming. The students shall be required to systematically work out the solution of those problems.*

Programme/Class: **Bachelorof Science**	Year: **Third**	Semester: **Sixth**
Subject: **Computer Science**		
Course Code: B070601T	Course Title: **Data Communication and Computer Network**	

Course outcomes:

After the completion of the course the students will be able:

1. To develop understanding of computer networks and communication basics.
2. To understand design issues and services at different layers of reference models.
3. To learn various error detection/correction techniques, routing protocols, congestion control algorithms, and connection establishment/release.
4. To describe and analyze related technical, administrative, and social aspects of networking.

Credits: **4**	**Core Compulsory**
Max. Marks: **25+75**	Min. Passing Marks: As per UGC/University CBCS norm

Total No. of Lectures-Tutorials-Practical (in hours per week): **4-0-0**

Unit	Topic	No. of Lectures
I	**Introduction to Signals** Data and Information, Data communication, Characteristics of data communication, Components of data communication, Data Representation, Data Flow, Simplex, Half Duplex, Full Duplex, Analog and Digital Signals, Periodic and Aperiodic signals, Time and Frequency Domain, Composite Signals	7
II	**Basic concepts of Networks:** Components of data communication, standards and organizations, Network Classification, Network Topologies ; network protocol; layered network architecture; overview of OSI reference model; overview of TCP/IP protocol suite.	8
III	**Physical Layer** : Cabling, Network Interface Card, Transmission MediaDevices- Repeater, Hub, Bridge, Switch, Router, Gateway.	7
IV	**Data Link Layer** Designing issues, Framing and Data Link Control, Error detection schemes (parity, checksums, CRCs), Error correction schemes (Hamming codes, binary convolution codes), Data link layer protocols (Simplest, Stop & Wait ARQ, Go-Back-N ARQ, Selective Repeat ARQ, Sliding Window), MAC sublayer (Ethernet, ALOHA, CSMA family, Contention-free access/Token Ring).	8

V	**Network Layer** Design issues, Switching, Routing algorithms (Shortest path, Link state, Flooding, Broadcast, Multicast), Packet Scheduling, Internetworking, Internet Protocol (IPv4, IPv6), IP addressing, Internet Control Protocols (IMCP, ARP, DHCP), Mobile IP.	8
VI	**Transport Layer** Transport layer services, Connection establishment and teardown, TCP, UDP, Congestion Control, Quality of Service, Domain Name System, World Wide Web.	8
VII	**Application Layer** : Application layer protocols and services –Domain name system, HTTP, WWW, telnet, FTP, SMTP	7
VIII	**Network Security** : Common Terms, Firewalls, Virtual PrivateNetworks	7

Suggested Readings:

1. Andrew S. Tanenbaum and David J. Wetherall, "Computer Networks," Fifth Edition, Pearson, 2014.
2. William Stallings, "Data and Computer Communications", Tenth Edition, Pearson, 2013.
3. Behrouz A. Forouzan, "Data Communications and Networking," Fourth Edition, McGraw-Hill Higher Education, 2007

Programme/Class: **Bachelor In Science**	Year: **Third**	Semester: **Sixth**
Subject: **Computer Science**		
Course Code: B070602T	Course Title: **Cyber Security & Cyber Laws**	

Course outcomes: After the completion of the course the students will be able to:

1. Understand types of information, cyber threats, and national/international cyber security standards.
2. Do mathematical modeling and development of security techniques and information system.
3. Develop understanding of legal issues related to cyber security.
4. Apply ethical principles/responsibilities in cyber practices.

Credits: **4**	**Core Compulsory**
Max. Marks: **25+75**	Min. Passing Marks: As per UGC/University CBCS norm

Total No. of Lectures-Tutorials-Practical (in hours per week): **4-0-0**

Unit	Topic	No. of Lectures
I	**Introduction:** Introduction to Information System, Type of information system, Development of information system, CIA model of Information Characteristics, Introduction to Information Security, Need of Information Security, Cyber Security, Business need, Ethical and Professional issues of security.	7
II	Information Security Model, Component of an Information security, Aspect of information security, Security attacks (Active and Passive Attacks), Security mechanism and Security Services (X.800).	8
III	Information Security Techniques, Introduction to Cryptography: Terminology, cryptanalysis, Security of algorithms, Substitution Cipher and Transposition Cipher, Single XOR , One-way Pad,	7
IV	Cryptographic Protocols-I: Arbitrated and Adjudicated Protocol, One- Way Hash function,	8
V	Cryptographic Protocols-II: Public key cryptography, Digital Signature, Digital Watermarking Technique: Characteristics and Types.	7

VI	Security Policies, Why Policies should be developed, WWW policies, Email Security policies, Policy Review Process-Corporate policies- Sample Security Policies.	8

VII	**Cyber Laws I:** Information Security Standards, IT act 2000 Provisions, Introduction to digital laws,	7
VIII	**Cyber Laws II:** cyber laws, intellectual property rights, copyright laws, patent laws, software license.	8

Suggested Readings:

1. Michael E. Whitman and Herbert J. Mattord, "Principles of Information Security," Sixth Edition, Cengage Learning, 2017.
2. Douglas J. Landoll, "Information Security Policies, Procedure, and Standards: A Practitioner's Reference," CRC Press, 2016.
3. Harold F. Tipton, and Micki Krause, "Hand book of information security management," Sixth Edition, Archtech Publication, 2007.
4. William Stallings, "Cryptography and Network Security: Principles and Practice," Sixth Edition, Pearson, 2014.

Programme/Class: **Bachelorof Science**	Year: **Third**	Semester: **Sixth**
Subject: **Computer Science**		
Course Code: B070603P	Course Title: Lab on Computer Networks based on Course code B070601T	
Course outcomes: CO1 Understand and explain the concept of Data Communication and networks, layered architecture and their applications. CO2 Analyze and Set up protocol designing issues for Communication networks. CO3 Evaluate data communication link considering elementary concepts of data link layer protocols for error detection and correction. CO4 Apply various network layer techniques for designing subnets and supernets and analyze packet flow on basis of routing protocols. CO5 Estimate the congestion control mechanism to improve quality of service of networking application		
Credits: **2**	**Core Compulsory**	
Max. Marks: **50**	Min. Passing Marks: As per UGC/University CBCS norm	
Total No. of Lectures-Tutorials-Practical (in hours per week): **0-0-4**		

Software Lab based on Computer Networks:

Implement the concepts of Computer Networks such as:

1. Simulate Checksum Algorithm.
2. Simulate CRC Algorithm
3. Simulate Stop & Wait Protocol.
4. Simulate Go-Back-N Protocol.
5. Simulate Selective Repeat Protocol.

and so on….

Programme/Class: **Bachelorof Science**	Year: **Third**	Semester: **Sixth**
Subject: **Computer Science**		
Course Code: B070604P	Course Title: Viva-voce examination conducted by external examiner at the end of the Session based on Course code B070602T	

Course outcomes:

CO 1: Understand the various information system and need of their security.
CO 2: Understand the various information system models.
CO 3: Understand the various security policies.
CO 4: Understand the legal issues related to cyber security.

Credits: **2**	Max. Marks: **50**	Min. Passing Marks: As per UGC/University CBCS norm
Total No. of Lectures-Tutorials-Practical (in hours per week): **0-0-4**		

*Viva-voce will be based on Course code B070602T (**Cyber Security & Cyber Laws**). Teacher-in-Charge shall design some problems/case study on* information system, security policies and legal issues related to cyber security. *The students shall be required to systematically work out the solution of those problems.*