The University of Burdwan

Syllabus of 3 Years Degree / 4 YearsHonours in Computer Science

Under Curriculum and Credit Framework for Undergraduate Programme (CCFUP) in Computer Science as per NEP, 2020.

Preamble

The new curriculum of the four year undergraduate program under NEP, for computer science aims to develop the core competence in computing and problem solving amongst its graduates. Informally, "Learning to learn" has been the motto of the department since its inception. The curriculum thus focuses on building theoretical foundations in computer science to enable its pupils to think critically when challenged with totally different and new problems. It imbibes the following Student-Centric features of NEP 2020:

Flexibility to Exit: In order to support early exits, the curriculum aims to develop employability skills early. As programming is at the heart of computing it is proposed to have two programming courses early so that the students candevelop good programming skills in the first year. At the same time students are familiarized with the hardware of computers early on.

Employability: Industry demand in the IT sector has changed considerably in the past few years. With the humongous amount of data coming from all the domains like medical data, social networking data, astronomical data, education, etc., automating information extraction and analysis of data is the only way forward to leverage the available data for the future. This curriculum aims to equip the students with tools and techniques of Artificial Intelligence, Machine Learning and a pathway on Data Science if the student so desires. Having said this, there is no replacement for the foundational courses like programming, data structures and algorithms. With two courses on programming and two courses on data structures and algorithms together, a strong foundation will be laid down for problem solving.

Multidisciplinary/Minor: The curriculum provides two pathways one of computer science (CS) , minor and the other of interdisciplinary, to the students from other disciplines. Those who want to earn a minor in CS will be required to choose the first pathway whereas those who simply want to apply IT in the domain of their interest can choose the second pathway.

Research: With the option to obtain specialization in an area of their choice, the curriculum prepares the students to take up research projects in their final year.

Discipline Specific Graduate Attributes (DSGA)

1. Proficiency in writing readable, correct, efficient, and secure programs of modestcomplexity.
2. Ability to design efficient algorithms using appropriate data structures for new problems.
3. Understanding of computer architecture, operating systems, computer networks anddatabase management systems and their role in the performance of software applications.
4. Understanding of theoretical foundations and limits of computing.
5. Ability to develop good quality software by following the processes of softwaredevelopment life cycle.
6. Ability to extract information and analyze large volumes of data employing a range oftechniques for artificial intelligence and learning.
7. Ability to develop an end to end compiler using compiler designing tools and techniques.
8. Ability to protect the data and software from various types of cyber attacks.

SEMESTER WISE COURSE STRUCTURE

Semester	Course Type & Course Code	Name of the Course	Credit	Lect.	Tuto.	Pract./ Viva	Full Marks	Distribution of Marks		
								Theory	Pract. / Tuto./ Viva-voce	Internal Assessment
I	Major/DS Course (Core) **COMP 1011**	Computer Fundamentals & Digital Logic	4	3	0	1	75	40	20	15
	Minor Course	From other allied discipline	4				75			15
	Multi/Interdisciplinary	From other discipline	3				50			10
	Ability Enhancement Course (AEC) [L_1-1 MIL] **AEC 1041**	Arabic/ Bengali/ Hindi/ Sanskrit/ Santali/ Urdu or Equvlnt. Course from SWAYAM or UGC recognized others	2	2	0	0	50	40	0	10
	Skill Enhancement Course (SEC) **COMP 1051**	Python Programming	3	0	0	3	50	NIL	40	10
	Common Value Added (CVA) Course **CVA1061**	Environmental Science/ Education	4	3/3	1/0	0/1	100	60	20	20
	Total		**20**				**400**			

Semester	Course Type	Name of the Course	Credit	Lect.	Tuto.	Pract. /Viva	Full Marks	Distribution of Marks		
								Theory	Pract. / Tuto./ Viva-voce	Internal Assessment
II	Major/DS Course (Core) **COMP 2012**	Programming Fundamentals using C	4	3	0	1	75	40	20	15
	Minor Course	From other allied discipline	4				75			15
	Multi/Interdisciplinary	From other discipline	3				50			10
	Ability Enhancement Course (AEC)[L_2-1] **AEC 2042**	English or Equvlnt. Course from SWAYAM or UGC recognized others	2	2	0	0	50	40	0	10
	Skill Enhancement Course (SEC) **COMP 2052**	System Administration & Maintenance	3	0	0	3	50	NIL	40	10
	Common Value Added (CVA) Course **CVA 2062**		4	3/3	1/0	0/1	100	80/60	0/20	20
	Total		**20**				**400**			

Students exiting the programme after securing 40 credits will be awarded UG Certificate in Computer Science provided they secure 4 credits in work based vocational courses offered during summer term or internship / Apprenticeship in addition to 6 credits from skill-based courses earned during first and second semester.

Semester	Course Type	Name of the Course	Credit	Lect.	Tuto.	Pract. /Viva	Full Marks	Distribution of Marks		
								Theory	Pract. / Tuto./ Viva-voce	Internal Assessment
III	Major/DS Course (Core) **COMP 3013**	Discrete Structure	5	5	0	0	75	60	NIL	15
	Major/DS Course (Core) **COMP 3014**	Data Structures	5	4	0	1	75	40	20	15
	Minor Course	Intermediate Level Course (Voc. Edn. & Trng.)	4				75			15
	Multi/Interdisciplinary	From other discipline	3				50			10
	Ability Enhancement Course (AEC) [L_1-2 MIL] **AEC 3043**	Arabic/ Bengali/ Hindi/ Sanskrit/ Santali/ Urdu] or Equvlnt. Course from SWAYAM or UGC recognized others	2	2	0	0	50	40	0	10
	Skill Enhancement Course (SEC) **COMP 3053**	PHP Programming	3	0	0	3	50	NIL	40	10
	Total		**22**				**375**			

Semester	Course Type	Name of the Course	Credit	Lect.	Tuto.	Pract./Viva	Full Marks	Distribution of Marks		
								Theory	Pract. / Tuto./ Viva-voce	Internal Assessment
IV	Major/DS Course (Core) **COMP 4015**	Design & Analysis of Algorithm	5	4	0	1	75	40	20	15
	Major/DS Course (Core) **COMP 4016**	Object Oriented Technology using JAVA	5	4	0	1	75	40	20	15
	Major/DS Course (Core) **COMP 4017**	Computer System Architecture	5	4	0	1	75	40	20	15
	Minor Course	Intermediate Level Course (Voc. Edn. & Trng.)	4				75			15
	Ability Enhancement Course (AEC)[L_2-2] **AEC 4044**	English or Equvlnt. Course from SWAYAM or UGC recognized others	2	2	0	0	50	40	0	10
	Total		**21**				**350**			

Students exiting programme after securing 83 credits will be awarded UG Diploma in the Computer Science provided they secure additional 4 credit in skill based vocational courses offered during first year or second year summer term.

Semester	Course Type	Name of the Course	Credit	Lect.	Tuto.	Pract. /Viva	Full Marks	Distribution of Marks		
								Theory	Pract. / Tuto./Viva-voce	Internal Assessment
V	Major/DS Course (Core) **COMP 5018**	Operating Systems	5	4	0	1	75	40	20	15
	Major/DS Course (Core) **COMP 5019**	Microprocessor	5	4	0	1	75	40	20	15
	Major/DS Course (Core) **COMP 50110**	Software Engineering	5	4	1	0	75	60	NIL	15
	Minor Course	From other allied discipline	4				75			15
	Internship (for all students) **COMP 5081**		2	0	0	2	50	Project - 30 + Viva - 20		
	Total		**21**				**350**			

Semester	Course Type	Name of the Course	Credit	Lect.	Tuto.	Pract./ Viva	Full Marks	Distribution of Marks		
								Theory	Pract. / Tuto./ Viva-voce	Internal Assessment
VI	Major/DS Course (Core) **COMP 60111**	Computer Networks	4	3	0	1	75	40	20	15
	Major/DS Course (Core) **COMP 60112**	Database Management Systems	4	3	0	1	75	40	20	15
	Major/DS Course (Core) **COMP 60113**	Theory of Computation	4	4	0	0	75	60	NIL	15
	Major/DS Course (Core) **COMP 60114**	UNIX/Linux Programming	4	3	0	1	75	40	20	15
	Minor Course	Intermediate Level Course (Voc. Edn. & Trng.)	4				75			15
	Total		**20**				**375**			

Students who want to undertake 3-year UG programme will be awarded UG Degree in Computer Science upon securing 124 credits.

Semester	Course Type	Name of the Course	Credit	Lect.	Tuto.	Pract. /Viva	Full Marks	Distribution of Marks		
								Theory	Pract. / Tuto./ Viva-voce	Internal Assessment
VII	Major/DS Course (Core) **COMP 70115**	Artificial Intelligence	6	4	0	2	75	40	20	15
	Major/DS Course (Core) **COMP 70116**	Internet Technology	6	4	0	2	75	40	20	15
	Major/DS Course (Core) **COMP 70117**	Numerical methods	6	4	0	2	75	40	20	15
	Major/DS Course (Core) **COMP 70118**	Soft Computing	6	4	0	2	75	40	20	15
	Minor Course	From other discipline of choice	4				75			15
	Total		**28**				**375**			

Semester	Course Type	Name of the Course	Credit	Lect.	Tuto.	Pract /Viva	Full Marks	Distribution of Marks		
								Theory	Pract. / Tuto./ Viva-voce	Internal Assessment
VIII For UG Hons. With Research Project/ Dissertation	Major/DS Course (Core) **COMP 80119**	Computer Graphics	6	4	0	2	75	40	20	15
	Minor Course	From other discipline of choice	4				75			15
	Research Project COMP 8091	Research Project/ Dissertation	12	0	0	12	225	Seminar Presentation, Preparation & Submission of Research Project/ Dissertation - 135 + Viva - 90		
VIII For UG Hons. without Research Project/ Dissertation	Major/DS Course (Core) **COMP 80119**	Computer Graphics	6	4	0	2	75	40	20	15
	Major/DS Course (Core) **COMP 80120**	Compiler Design	4	4	0	0	75	60	20	15
	Major/DS Course (Core) **COMP 80121**	Machine Learning	4	3	0	1	75	40	20	15
	Major/DS Course (Core) **COMP 80122**	Data Mining	4	3	0	1	75	40	20	15
	Minor Course	From other discipline of choice	4				75			15
	Total		**22**				**375**			
	Total Credit/ Marks		**174**				**3000**			

Students will be awarded UG Degree (Honours/ Honours with Research) in Computer Science provided they secure 174 credits.

Minor Courses Offered by Department of Computer Science(for other allied discipline)							
Semester	**Name of the Course**	**Course Code**	**Credit L-T-P**	**F.M**	**Theory**	**Practical**	**Internal**
Semester I:	Computer Fundamentals & Digital Logic	**COMP 1021**	3-0-1=4	75	40	20	15
Semester II:	Python Programming	**COMP 2022**	3-0-1=4	75	40	20	15
Semester V:	Database Management System	**COMP 5025**	3-0-1=4	75	40	20	15
Semester VII:	Operating Systems	**COMP 7027**	3-0-1=4	75	40	20	15
Semester VIII:	Computer System Architecture	**COMP 8028**	3-0-1=4	75	40	20	15

N.B.: Vocational Education & Training as 3 Minor Courses for 12 Credits will be offered in 3rd, 4th & 6th Semesters.

Multi / Interdisciplinary Courses Offered by Department of Computer Science(for other discipline)							
Semester	**Name of the Course**	**Course Code**	**Credit L-T-P**	**F.M**	**Theory**	**Practical**	**Internal**
Semester I:	Basic IT tools	**COMP 1031**	3-0-0=3	50	40	NIL	10
Semester II:	Introduction to Internet	**COMP 2032**	3-0-0=3	50	40	NIL	10
Semester III:	Introduction to Cyber Security	**COMP 3033**	3-0-0=3	50	40	NIL	10

Major/DS Course (Core)

COMP 1011: Computer Fundamentals & Digital Logic

Credit: 03 **45 Hours**

Course Objective

This course introduces the students to the fundamental concepts of digital computer organization and design. It aims to develop a basic understanding of the building blocks of the computer system and highlights how these blocks are organized together to architect a digital computer system. The course teaches the fundamentals of digital systems, applying the logic design and development techniques. This course forms the basis for the study of advanced subjects like Computer Architecture and Organization, Microprocessor through Interfacing, VLSI Designing etc.

Course Learning Outcomes

On successful completion of the course, students will be able to:

i. Acquire the basic knowledge of digital logic to understand digital electronics circuits.

ii. Prepare students to perform the analysis and design of various digital electronic circuits.

iii. Design and simplify combinational and sequential circuits using basic buildingblocks.

iv. Represent data in binary form, convert numeric data between different number systemsand perform arithmetic operations in binary.

v. Simulate the design of a basic computer using a software tool/ digital trainer kit.

Syllabus

Computer Fundamentals (15 Hours)

Introduction to Computer and Problem Solving: Information and Data. Hardware: CPU, Primary and Secondary storage, I/O devices, Bus structure Software: Systems and Application. Generation of Computers: Super, Mainframe, Mini and Personal Computer. Introduction to Programming Languages: Machine Language, Assembly Language, High Level Language. Problem Solving: Flow Charts, Decision Tables and Pseudo codes. Number Systems and Codes: Number representation: Weighted Codes, Non-weighted codes, Positional, Binary, Octal, Hexadecimal, Binary Coded Decimal (BCD), Conversion of bases. Complement notions. Binary Arithmetic, Binary Codes: Gray, Alphanumeric, ASCII, EBCDIC; Single Error-Detecting and Correcting Codes, Hamming Codes, IEEE 754 floating point representation. Boolean algebra: Fundamentals of Boolean algebra, Switches and Inverters, Functionally Complete Gates (AND, OR, NOT), NAND, NOR, Switching function and Boolean Function. De Morgan's Theorem, Minterms, Maxterms, Truth table and minimization of switching function up to four variables, Algebraic and K-map method of Logic circuit synthesis: Two-level and Multi-level

Digital Logic(30 Hours)

Combinational Circuits: Realization of AND and OR Gates using diodes and NOT Gate using transistors, Standard Gate Assemblies, IC chips packaging nomenclature, Half and Full Adder(3 & bit), Multi-bit adders – Ripple carry and Carry Look Ahead Adder, Adder/subtractor, BCD-Adder, Data selectors/multiplexers – expansions, reductions, function realization, universal function 5 realization, multi-function realization, Decoders: function realization, De-multiplexer and function realization, Encoder, Priority Encoder, Parity bit Generator/checker, Gray Code Generator, Code Converters, Keyboard encoder, Seven segment display unit, Comparators. Sequential Circuits: Model of Sequential computing, Difference between Combinational and Sequential circuit, RS-Latch: using NAND and NOR Gates, RS Latch as a Static RAM Cell, Problems of Basic Latch circuits, Digital Clock – Duty Cycle, Rising time, Falling time, Clocked Flip Flops - SR, JK, D, T, Level Trigger and Edge Trigger, Excitation Functions of each flip-flops, Flip-flops with Preset and Clear, Application of Flip-flops: Asynchronous Counter(UP/DOWN) up to 4-bit counter, Decade Counter, Mod – n Counter, Finite State machine Model – State Transition Diagram and Table, Synchronous Counters – different mod counters, Ring counter, Johnson's Counter, Registers, Registers with parallel load, Shift Registers.

Practical :: Digital Circuit Design

Credit: 01 **30 Hours**

Combinational Circuits:

1) Implement Half Adder/Half Subtractor/Full Adder/Full Subtractor using Logic Gates. Realize a logic function using basic/universal gates in SOP and POS form. Study the functionalities of 7483 and design a BCD adder using 7483 or equivalent.
2) Design of two level AND – OR, NAND –NAND, NOR-NOR circuits to realize any truth table. Realize XOR in two level and multilevel.
3) Design a 4 bit 2's complement adder – subtractor unit using 7483 or equivalent and XOR gates.
4) Design a circuit to convert BCD numbers to corresponding gray codes.
5) Design a 4:1 MUX using NAND gates. Study of 74153 and 74151. Design Full Adder/Subtractor using MUX.
6) Design a 2:4 decoder using NAND gates. Study of 74155 and 74138. Design Full Adder/Subtractor using decoders.
7) Design a parity generator/checker using basic gates.
8) Design magnitude comparator using basic/universal gates. Study of 7485.
9) Design a seven segment display unit.

Sequential Circuits:

1) Realize S-R, D, J-K and T flip-flop using basic gates. (Study the undefined state in S-R flip-flop).
2) Design a shift register (shift left and shift right) using flip-flops. (Study the functional characteristic of IC 74194 with emphasis on timing diagram).
3) Design Asynchronous and Synchronous counters. Study of IC 74193.
4) Study the functional characteristics of RAM IC chip. Study of open collector and tri-state output. Horizontal and vertical expansion of RAM chips by cascading. Use 74189, 7489, 2114 or any available chip.

Reference Books:

1. Digital Logic and Computer Design by M.Morris Mano, PHI
2. Digital Fundamentals by Floyd, Pearson Education
3. Digital Principle and Applications by Malvino & Leach, TMH
4. P. K. Sinha & Priti Sinha , Computer Fundamentals‖, BPB Publications, 2007.
5. Dr. Anita Goel, Computer Fundamentals, Pearson Education, 2010.

Skill Enhancement Course (SEC)

Credit: 03 **90 Hours**

COMP 1051: Programming in Python (Practical)

Course Objective

The course is designed to introduce programming concepts using Python to students. The course aims to develop structured as well as object-oriented programming skills using Python. The course also aims to achieve competence amongst its students to develop correct and efficient Python programs to solve real life problems.

Course Learning Outcomes

On successful completion of the course, students will be able to:

1. Develop, document, and debug modular Python programs of reasonable complexity.
2. Implement arrays and user defined functions in Python.
3. Solve real life problems of reasonable complexity using suitable and efficientprogramming constructs in Python.
4. Solve real life problems of reasonable complexity using the concepts of object orientedprogramming in Python.

Syllabus

Planning the Computer Program: Concept of problem solving, Problem definition, Program design, Debugging, Types of errors in programming, Documentation.(20 Hrs)

Techniques of Problem Solving: Flowcharting, decision table, algorithms, Structured programming concepts, Programming methodologies viz. top-down and bottom-up programming.(20 Hrs)

Overview of Programming: Structure of a Python Program, Elements of Python (10 Hrs)

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, List , Tuple , Set and Dictionary.(20 Hrs)

Creating Python Programs : Input and Output Statements, Control statements(Branching, Looping, Conditional Statement, Exit function, Difference between break, continue and pass.), Defining Functions, default arguments.(20Hrs)

Reference Books

1. T. Budd, Exploring Python, TMH, 1st Ed, 2011
2. Python Tutorial/Documentation www.python.or 2015
3. Allen Downey, Jeffrey Elkner, Chris Meyers , How to think like a computer scientist : learning with Python , Freely available online.2012
4. http://docs.python.org/3/tutorial/index.html
5. http://interactivepython.org/courselib/static/pythonds
6. http://www.ibiblio.org/g2swap/byteofpython/read/

Software Lab Based on Python:

Section: A (Simple programs)

- Write a menu driven program to convert the given temperature from Fahrenheit to Celsius and vice versa depending upon users choice.
- 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 <80Grade C:
  Percentage>=60 and <70Grade D:
  Percentage>=40 and <60 Grade E:
  Percentage<40

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)

Minor Courses(For other discipline)::

Semester-I

COMP 1021: Computer Fundamentals & Digital Logic

Credit: 03 **45 Hours**

Course Objective

This course introduces the students to the fundamental concepts of digital computer organization and design. It aims to develop a basic understanding of the building blocks of the computer system and highlights how these blocks are organized together to architect a digital computer system. The course teaches the fundamentals of digital systems, applying the logic design and development techniques. This course forms the basis for the study of advanced subjects like Computer Architecture and Organization, Microprocessor through Interfacing, VLSI Designing etc.

Course Learning Outcomes

On successful completion of the course, students will be able to:

i. Acquire the basic knowledge of digital logic to understand digital electronics circuits.

ii. Prepare students to perform the analysis and design of various digital electronic circuits.

iii. Design and simplify combinational and sequential circuits using basic buildingblocks.

iv. Represent data in binary form, convert numeric data between different number systemsand perform arithmetic operations in binary.

v. Simulate the design of a basic computer using a software tool/ digital trainer kit.

Syllabus

Computer Fundamentals (15 Hours)

Introduction to Computer and Problem Solving: Information and Data. Hardware: CPU, Primary and Secondary storage, I/O devices, Bus structure Software: Systems and Application. Generation of Computers: Super, Mainframe, Mini and Personal Computer. Introduction to Programming Languages: Machine Language, Assembly Language, High Level Language. Problem Solving: Flow Charts, Decision Tables and Pseudo codes. Number Systems and Codes: Number representation: Weighted Codes, Non-weighted codes, Positional, Binary, Octal, Hexadecimal, Binary Coded Decimal (BCD), Conversion of bases. Complement notions. Binary Arithmetic, Binary Codes: Gray, Alphanumeric, ASCII, EBCDIC; Single Error-Detecting and Correcting Codes, Hamming Codes, IEEE 754 floating point representation. Boolean algebra: Fundamentals of Boolean algebra, Switches and Inverters, Functionally Complete Gates (AND, OR, NOT), NAND, NOR, Switching function and Boolean Function. De Morgan's Theorem, Minterms, Maxterms, Truth table and minimization of switching function up to four variables, Algebraic and K-map method of Logic circuit synthesis: Two-level and Multi-level.

Digital Logic(30 Hours)

Combinational Circuits: Realization of AND and OR Gates using diodes and NOT Gate using transistors, Standard Gate Assemblies, IC chips packaging nomenclature, Half and Full Adder(3 & bit), Multi-bit adders – Ripple carry and Carry Look Ahead Adder, Adder/subtractor, BCD-Adder, Data selectors/multiplexers – expansions, reductions, function realization, universal function 5 realization, multi-function realization, Decoders: function realization, De-multiplexer and function realization, Encoder, Priority Encoder, Parity bit Generator/checker, Gray Code Generator, Code Converters, Keyboard encoder, Seven segment display unit, Comparators. Sequential Circuits: Model of Sequential computing, Difference between Combinational and Sequential circuit, RS-Latch: using NAND and NOR Gates, RS Latch as a Static RAM Cell, Problems of Basic Latch circuits, Digital Clock – Duty Cycle, Rising time, Falling time, Clocked Flip Flops - SR, JK, D, T, Level Trigger and Edge Trigger, Excitation Functions of each flip-flops, Flip-flops with Preset and Clear, Application of Flip-flops: Asynchronous Counter(UP/DOWN) up to 4-bit counter, Decade Counter, Mod – n Counter, Finite State machine Model – State Transition Diagram and Table, Synchronous Counters – different mod counters, Ring counter, Johnson's Counter, Registers, Registers with parallel load, Shift Registers.

Practical :: Digital Circuit Design

Credit: 01 **30 Hours**

Combinational Circuits:

1) Implement Half Adder/Half Subtractor/Full Adder/Full Subtractor using Logic Gates. Realize a logic function using basic/universal gates in SOP and POS form. Study the functionalities of 7483 and design a BCD adder using 7483 or equivalent.
2) Design of two level AND – OR, NAND –NAND, NOR-NOR circuits to realize any truth table. Realize XOR in two level and multilevel.
3) Design a 4 bit 2's complement adder – subtractor unit using 7483 or equivalent and XOR gates.
4) Design a circuit to convert BCD numbers to corresponding gray codes.
5) Design a 4:1 MUX using NAND gates. Study of 74153 and 74151. Design Full Adder/Subtractor using MUX.
6) Design a 2:4 decoder using NAND gates. Study of 74155 and 74138. Design Full Adder/Subtractor using decoders.
7) Design a parity generator/checker using basic gates.
8) Design magnitude comparator using basic/universal gates. Study of 7485.
9) Design a seven segment display unit.

Sequential Circuits:

i. Realize S-R, D, J-K and T flip-flop using basic gates. (Study the undefined state in S-R flip-flop).
ii. Design a shift register (shift left and shift right) using flip-flops. (Study the functional characteristic of IC 74194 with emphasis on timing diagram).
iii. Design Asynchronous and Synchronous counters. Study of IC 74193.
iv. Study the functional characteristics of RAM IC chip. Study of open collector and tri-state output. Horizontal and vertical expansion of RAM chips by cascading. Use 74189, 7489, 2114 or any available chip.

Reference Books

a. Digital Logic and Computer Design by M.Morris Mano, PHI
b. Digital Fundamentals by Floyd, Pearson Education
c. Digital Principle and Applications by Malvino & Leach, TMH
d. P. K. Sinha & Priti Sinha , Computer Fundamentals‖, BPB Publications, 2007.
e. Dr. Anita Goel, Computer Fundamentals, Pearson Education, 2010.

Multi/Interdisciplinary courses

(For other discipline)

Semester-I

Credit: 03 **45 Hours**

COMP 1031: Basic IT Tools(Theory)

Course Objective

The goal of this course is to present overview of IT tools used in day to day use of computers and data base operations. The Course has been designed to provide knowledge on various hardware and software components of computer, operating system, various packages used for different applications, data base concepts & operations and various issues related to IT and application of IT.

Course Learning Outcomes:

On successful completion of the Course, a student will :

i. Acquire the foundation level knowledge required to understand computer and its operations.
ii. Understand the hardware and software components of the computer.
iii. Understand the basic concept of operating system and get knowledge about various different operating systems.
iv. Understand to use the packages of word processing, spread sheet and presentation in detail.
v. Understand various data base concepts and operations.
vi. Understand the issues related to IT and IT applications
vii. Prepare research and academic related presentations.

Syllabus

Introduction – Introduction to computers – Evolution – Generation of Computers – Computers Hierarchy – Applications of Computers. (5 Hrs)

Windows Basics – Introduction to word – Editing a document - Move and Copy text - Formatting text & Paragraph – Enhancing document – Columns, Tables and Other features.(10 Hrs)

Introduction to worksheet and shell – getting started with Excel – Editing cell & using Commands and functions – Moving & Copying , Inserting & Deleting Rows & Columns - Printing work sheet.(5 Hrs)

Creating charts – Naming ranges and using statistical, math and financial functions, database in a worksheet – Additional formatting commands and drawing toolbar – other commands & functions – multiple worksheet and macros.(10 Hrs)

Introduction to Database Development: Database Terminology, Objects, Creating Tables, working with fields, understanding Data types, Changing table design, Assigning Field Properties, Setting Primary Keys, Select data with queries: Creating simple Query by design & by wizard (10 Hrs)

Overview of Power point – presenting shows for corporate and commercial using Power point –Introduction to Desktop publishing – Computer viruses – Introduction to Internet – Web features.(5 hrs)

Reference Books:

i. Swinford, E., Dodge, M., Couch, A., Melton, B. A. (2013). Microsoft OfficeProfessional 2013. United States: O'Reilly Media.
ii. Wang, W. (2018). Office 2019 For Dummies. United States: Wiley. Microsoft Lambert, J. (2019). Microsoft Word 2019 Step by Step. United States: Pearson Education.
iii. Jelen, B. (2013). Excel 2013 Charts and Graphs. United Kingdom: Que.
iv. Alexander, M., Jelen, B. (2013). Excel 2013 Pivot Table Data Crunching. UnitedKingdom: Pearson Education.
v. Alexander, M., Kusleika, R. (2018). Access 2019 Bible. United Kingdom: Wiley.

Semester-II

Major/DS Course (Core)

COMP 2012: Programming Fundamentals using C

Credit: 03 **45 Hours**

Course Objective

The course is designed to provide complete knowledge of C language. Students will be able to develop logics which will help them to create programs, applications in C. Also by learning the basic programming constructs they can easily switch over to any other language in future.

Course Learning Outcomes: After successful completion of the Course a student will be able to:

i. Develop problem solving skills coupled with top-down design principles.
ii. Become skilled at developing simple algorithms and flow charts.
iii. Convert the algorithms into simple C programs.
iv. Understand code organization and functional hierarchical decomposition .
v. Develop simple C programs for solving real life problems.

Syllabus

Theory:

Introduction: Basic Structure, Character sets, Keywords, Identifiers, Constants, Variables, Data Types, Program Structure. Operators: Arithmetic, Relational, Logical and Assignment; Increment, Decrement and Conditional, Operator Precedence and Associations; Assignment, Initialization, Expressions. Expression evaluation and type conversion. Formatted input and output, Conditional statements, Branching and looping, Array. (15 Hrs)

Functions – Arguments passing, Return values and their types, recursion .String handling with arrays, String handling functions,. Enumerated data types. Structures. Arrays of structures. Arrays within structures, union (10 Hrs)

Pointers: Declaration and initialization, Accessing variables through pointer arithmetic, Pointers and arrays, String, Pointer to Functions and Structures, Dynamic Storage Allocation.
(15Hrs)

File handlings: Opening, Closing, I/O operations.(5 Hrs)

C language Practical:

Credit: 01 **30 Hours**

1. WAP to print the sum and product of digits of an integer.
2. WAP to reverse a number.
3. WAP to compute the sum of the first n terms of the following series S = 1+1/2+1/3+1/4+……
4. WAP to compute the sum of the first n terms of the following series S =1-2+3-4+5…….
5. Write a function that checks whether a given string is Palindrome or not. Use this function to find whether the string entered by user is Palindrome or not.
6. Write a function to find whether a given no. is prime or not. Use the same to generate the prime numbers less than 100.
7. WAP to compute the factors of a given number.
8. Write a macro that swaps two numbers. WAP to use it.

9. WAP to print a triangle of stars as follows (take number of lines from user):

```
    *
   ***
  *****
 *******
*********
```

10. WAP to perform following actions on an array entered by the user:

i) Print the even-valuedelements
ii) Print the odd-valued elements
iii) Calculate and print the sum and average of the elements of array
iv) Print the maximum and minimum element of array
v) Remove the duplicates from the array
vi) Print the array in reverse order
The program should present a menu to the user and ask for one of the options. The menu should also include options to re-enter array and to quit the program.

11.WAP that prints a table indicating the number of occurrences of each alphabet in the text entered as command line arguments.

12. Write a program that swaps two numbers using pointers.

13.Write a program in which a function is passed address of two variables and then alter its contents.

14.Write a program which takes the radius of a circle as input from the user, passes it to another function that computes the area and the circumference of the circle and displays the value of area and circumference from the main() function.

15. Write a program to find sum of n elements entered by the user. To write this program, allocate memory dynamically using malloc() / calloc() functions or new operator.

16. Write a menu driven program to perform following operations on strings:

a) Show address of each character in string
b) Concatenate two strings without using strcat function.
c) Concatenate two strings using strcat function.
d) Compare two strings
e) Calculate length of the string (use pointers)
f) Convert all lowercase characters to uppercase
g) Convert all uppercase characters to lowercase
h) Calculate number of vowels
i) Reverse the string

17. Given two ordered arrays of integers, write a program to merge the two-arrays to get an ordered array.

18. WAP to display Fibonacci series (i)using recursion, (ii) using iteration

19. WAP to calculate Factorial of a number (i)using recursion, (ii) using iteration

20. WAP to calculate GCD of two numbers (i) with recursion (ii) without recursion.

21. Create Matrix class using templates. Write a menu-driven program to perform following Matrix

operations (2-D array implementation):

22. Sum b) Difference c) Product d) Transpose

23. Write a menu-driven program, using user-defined functions to find the area of rectangle, square, circle and triangle by accepting suitable input paramters from user.

24. WAP to display the first n terms of Fibonacci series.

25. WAP to find factorial of the given number.

26. WAP to find sum of the following series for n terms: 1 – 2/2! + 3/3! --------- n/n!

27. WAP to calculate the sum and product of two compatible matrices.

Reference Books

1. C Programming by Karnighan, & Ritchie, PHI
2. Programming through C by Richard Johnsonbaugh and Martin Kalin, Pearson Education

Skill Enhancement Course (SEC)

COMP 2052: System Administration and Maintenance (Practical)

Credit: 03 **90 Hours**

Course Objective

System administration is the field of work in which someone manages one or more systems, be they software, hardware, servers or workstations. Its goal is ensuring the systems are running efficiently and effectively.

Course Learning Outcomes: After successful completion of the Course a student will be able to:

i. Troubleshoot and fix issues that compromise system performance or access to an IT service.

ii. Make regular system improvements, such as upgrades based on evolving end-user and business requirements.

iii. Maintain common system and network administration tasks and practices and how to implement and maintain standard services like email, file sharing, DNS and similar tasks.

Syllabus

Part I (Linux/Unix) 45 hrs

- □ Basics of operating system, services,
- □ Installation and configuration, maintenance
- □ What is linux/unix Operating systems, Kernel, API, cli, gui,
- □ Difference between linux/unix and other operating systems
- □ Features and Architecture
- □ Linux features, advantages, disadvantages

Part II (Windows) 45 hrs

- □ Windows as operating system, history, versions.
- □ PC hardware, BIOS, Devices and drivers,
- □ Kernal Configuration and building
- □ Application installation, configuration and maintenance
- □ Server services and Client services

Software Lab Based on System Administration and Maintenance Linux:

Linux Desktop tour. Configuring desktop environment and desktop settings.

Basic Commands :Terminal, shell, cat, ls, cd, date, cal, man, echo, pwd, mkdir, rm, rmdir ps, killPackage Installation Synaptic package manager

Windows:

Creating users – Admin and regular.

Path of their personal files. Adding and changing passwords. Difference between workgroup and domain.

Concept of roles .user profiles – creating and roaming Concept of Active Directory. Creating active directoryin windows.

Process and Disk management

Windows Task manager. File systems – NTFS, FAT.

Services Control Panel

C:/program Files, C:/system C:/windows
Add /remove new hardware (like printer), Add/remove new programmes.
Network Administration
Ipconfig, Ping, tracert, route, hostname, net, netstat, who am I , Set
manual IP address, check connectivity – ipv4, ipv6 Administrator Tools
Control Panel -> Administrative Tools
Computer Management, Local security Policy, Performance Monitor, Task Scheduler, Antivirus and firewall.
Misc Start->Accessories->System tools -> All options (Remote desktop, backup/restore etc.) LAN – Configuration, Switch, Router, sharing printer, files and folder over the network.

Reference Books

i. The Practice of System and Network Administration , Thomas A. Limoncelli **,** Christina J. Hogan **,** Strata R. Chalup

ii. Modern System Administration, Jennifer Davis

Minor Courses::(For other discipline)

Semester-II

COMP 2022: Python Programming

Credit: 03 **45 Hours**

Course Objective

The course is designed to introduce programming concepts using Python to students. The course aims to develop structured as well as object-oriented programming skills using Python. The course also aims to achieve competence amongst its students to develop correct and efficient Python programs to solve real life problems.

Course Learning Outcomes

On successful completion of the course, students will be able to:

1. Develop, document, and debug modular Python programs of reasonable complexity.
2. Implement arrays and user defined functions in Python.
3. Solve real life problems of reasonable complexity using suitable and efficientprogramming constructs in Python.
4. Solve real life problems of reasonable complexity using the concepts of object orientedprogramming in Python.

Syllabus

Theory:

Planning the Computer Program: Concept of problem solving, Problem definition, Program design, Debugging, Types of errors in programming, Documentation. (10 Hrs)

Techniques of Problem Solving: Flowcharting, decision table, algorithms, Structured programming concepts, Programming methodologies viz. top-down and bottom-up programming. (10 Hrs)

Overview of Programming: Structure of a Python Program, Elements of Python (5 L)

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. (10 Hrs)

Creating Python Programs : Input and Output Statements, Control statements(Branching, Looping, Conditional Statement, Exit function, Difference between break, continue and pass.), Defining Functions, default arguments. (10 Hrs)

Reference Books

1) T. Budd, Exploring Python, TMH, 1st Ed, 2011
2) Python Tutorial/Documentation www.python.org
3) Allen Downey, Jeffrey Elkner, Chris Meyers , How to think like a computer scientist :learning with Python , Freely available online.2012
4) http://docs.python.org/3/tutorial/index.html
5) http://interactivepython.org/courselib/static/pythonds
6) http://www.ibiblio.org/g2swap/byteofpython/read/

Software Lab Based on Python(Practical):

Credit :: 1 **30 Hours**

Section: A (Simple programs)

- Write a menu driven program to convert the given temperature from Fahrenheit to Celsius and vice versa depending upon users choice.
- 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 <80Grade C:
  Percentage>=60 and <70Grade D:
  Percentage>=40 and <60 Grade E:
  Percentage<40

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:
   velocity wrt time (v=u+at)
   distance wrt time (s=u*t+0.5*a*t*t)
   distance wrt velocity (s=(v*v-u*u)/2*a)

Multi/Interdisciplinary courses

(For Other discipline)

Semester-II

COMP 2032: Introduction to Internet

Credit: 03 **45 Hours**

Course Objective

This course is intended to teach the basics involved in publishing content on the World Wide Web. This includes the 'language of the Web' – HTML and the fundamental principles of how the Internet and the Web function.

Course Learning Outcomes

On successful completion of the course, students will be able to:

i. Discuss elementary Internet concepts and history.
ii. Make a successful Internet connection.
iii. Demonstrate simple principles of Internet Protocol (IP) addressing.
iv. Use and customize a web browser.
v. Comprehend the basics of the internet and web terminologies.

Syllabus

Introduction : Evolution of Internet, concept of Intranet and Internet, Applications of Internet, Types of Connectivity such as dial – up, leased, VSAT. etc., Internet Server and Clients module in various Operating Systems.(5 Hrs)

Usenet and Internet Relay Chart Introduction to World Wide Web: Evolution of WWW, Basics Features, WWW Browsers, WWW servers, HTTP & URL's. (5 Hrs)

WWW Browsers: Basic features, Bookmarks, history. Progress indicators, Personalization of Browsers, Printing displayed pages and forms, Saving Web pages, Netscape Communicators, Internet Explorer, Search and Downloads.(5 Hrs)

Search Engines: Technology overview, Popular Search Engines. How to register a website in search engine. (5 Hrs)

Internet Security: Overview of Internet Security threats, Firewalls, Introduction to AAA (5 Hrs)

HTML: (20 Hrs)

- **Unit-I: Introduction**
- **Unit-II: The Basics**
  - The Head, the Body
  - Colors, Attributes
  - Lists, ordered and unordered
- **Unit-III: Links**
  - Introduction
  - Relative Links, Absolute Links
  - Link Attributes

- Using the ID Attribute to Link Within a Document

- **Unit-IV: Images**
  - Putting an Image on a Page
  - Using Images as Links
  - Putting an Image in the Background

- **Unit V: – Tables**
  - Creating a Table
  - Table Headers
  - Captions
  - Spanning Multiple Columns
  - Styling Table

Reference Books

1. Internetworking with TCP/IP – by D.E.Comer, PHI
2. Introduction to HTML and CSS -- O'Reilly