John Wilson Education Society's

Wilson College (Autonomous)

Chowpatty, Mumbai-400007
RE-ACCREDITED 'A' grade by NAAC

Affiliated to the

UNIVERSITY OF MUMBAI

Syllabus for F.Y
Program: BSc.I.T

(Major:Information Technology & Minor: Electronics)

Programme Code: WSITC

Choice Based Credit System (CBCS) with effect from Academic year 2023–2024

PROGRAM OUTLINE 2023-2024

YEAR	SEM	COURSE CODE	COURSE TITLE	CREDITS
FY	I			
	Major	WSITCMJ111	Programming Principles with C (Theory)	2
		WSITCMJ112	Operating System (Theory)	2
		WSITCMJ113	Programming Principles with C (Practicals)	2
	Minor	WSITCMN111	Digital Electronics	2
		WSITCMN112	Computer Networks(Theory)	2
		WSITCMN113	Computer Networks(Practicals)	2
	OE/GE	WSITCOE111	Fundamentals of Computers (Elective-I)	2
	OE/GE	WSITCOE112	Awareness Of Cyber Security (Elective-II)	2
	SEC	WSITCSE111	Operating System Practical	2
	II			
	Major	WSITCMJ121	OOPs with C++(Theory)	2
		WSITCMJ122	Web Designing & development(Theory)	2
		WSITCMJ123	OOPs with C++(Practical)	2
	Minor	WSITCMN121	Microprocessor and Architecture(Theory)	2
		WSITCMN122	Embedded Systems (Theory)	2
		WSITCMN123	Embedded Systems(Practical)	2
	OE/GE	WSICTOE121	Website Design (Elective-I)	2
	SEC	WSITCSE121	Digital Electronics Practicals	2

PROGRAMME SPECIFIC OUTCOME (PSOs)

After completing three years course of in Information Technology, the learner will be able to:

1. Equip with the professional & technical skills essential for making a career in the Front-end Developer, Back-end Developer, Software Tester, Programmer, Network Administrator, Security Expert etc.
2. Students would demonstrate the ability to apply research principles in a variety of creative, organizational, professional venues.
3. Apply the knowledge of engineering and management principles to manage projects effectively in diverse environments as a member or a leader in the team
4. Learners will understand technology as a system of interrelated forces, including automation, Robotics, Big data analytics, Advance Programming, Networking, Security, Cyber law, Regulatory constraints, and ethical concerns.
5. Learners will be able to create and design technological products, including website, software, robots, RFID, Hardware Automation, Also, will be able to become Software Engineer, Tester and Pen testers

PREAMBLE

With the introduction of Choice Based Credit System (CBCS) by the esteemed University of Mumbai from academic year 2016-17, the existing syllabus of FYBSCIT is restructured according to the CBCS pattern.

In the last two decades, technology has made a paradigm shift in the way society functions. Media, being the important segment of the society plays a pivotal role in the political, sociological, psychological, and economical aspects in society. In addition to the knowledge of the technical skills of mass media, the current syllabus orients to the theoretical framework relating to media-audience relationship.

The first year of BSCIT course introduces the students to various fields available under the umbrella of mass Programming, Mathematics, Communication Skills, Software’s, Database, Website Design.

The fundamentals of Information Technology prepare the students for advanced theoretical and practical related to field introduced. In the second year of BSCIT, followed by the elaboration on Information Technology research concepts and methodologies. Inclusion of field-based learning begins in the third year of BSCIT, with specialization in Information Technology.

WILSON COLLEGE (AUTONOMOUS) NEP SYLLABUS FOR BSC.I.T.

PROGRAM(s): FYBScIT		**SEMESTER: I**			
Course:- **Programming Principles with C (Theory)**	**Major**	**Course Code: WSITCMJ111**			
Teaching Scheme					**Evaluation Scheme**
Lectures (Hours per week)	**Practical (Hours per week)**	**Tutorial (Hours per week)**	**Credit**	**Continuous Internal Assessment (CIA)** **(Marks- 40)**	**Semester End Examination (Marks- 60)**
2	**-**	**-**	**2**	**40**	**60**

Learning Objectives:

1) Enhancing student logical reasoning through C programming language.
2) Clarifying fundamental concepts through relevant examples.
3) Adopting alternative strategies for problem-solving.
4) Mastering error handling techniques and identifying effective solutions.
5) Troubleshooting and debugging code for seamless execution.

Course Outcomes:

1) Master the foundational principles of programming.
2) Logic Development through Algorithms and Flowcharts.
3) Exploring and Mastering Different Data Types.
4) Mastering Input and Output Functions for Effective Programming.
5) Strengthening Advanced Concepts through Practical Programming Examples.

DETAILED SYLLABUS

Unit	**Sub unit**	**Course/ Unit Title**	**Lectures (hours)**
	1	**Introduction:**	
I	1.1	History of C,Applications of C, Features of C Language,Meaning of AlgorithmTypes of Programming Languages,Structure of C	**5**

		Program. pseudo code statements and flowchart symbols, Compilation and Execution of a Program, Character Set, identifiers and keywords, data types and sizes, constants and its types, variables, Rules for Variables, Algorithms	
	1.2	**Operators and Expressions:**	**4**
		Arithmetic operators, unary operators, relational and logical operators, assignment operators, assignment operators, the conditional operator, library functions	
II	2.1	Block Structure, Initialization, C Preprocessor Control Flow: Statements and Blocks, If-Else, Else-If, Switch, Loops- While and For LoopsDo-while, Break and Continue, Goto and Labels	**4**
	2.2	Functions and Program Structure: Basics of functions. User defined and Library functions, Function parameters, Return values, Recursion External variables, Scope Rules, Standard Input and Output, Formatted Output-printf() and Formatted Input- scanf(), Line Input and Output, Error Handling- StdErr and Exit, Header Files	**6**
III	3.1	Pointer and Arrays Pointer and Addresses, Pointer and Function Arguments, Pointer and Arrays, Address Arithmetic, Character Pointers and Functions, Pointer Arrays: Pointers and Functions, Multidimensional Array, Command-line Arguments, Pointers to Functions, Dynamic memory allocation	**6**
IV	4.1	Structures:Basics of `aa`structures, Structures and Functions, Arrays of Structures, Pointers to Structures, Unions, Bit-fields, File management in C: Defining and Opening file, Closing a file, Input / Output operations on file, Error handling in C, Random access to files, Command line arguments.	**5**

https://www.guru99.com/software-engineer-book.html

Book Title	**Author Name**	**Latest Edition**	**Publisher**
Clean Code	Robert C. Martin	1st edition	PHI
Design Patterns	Erich Gamma	1st edition	Addison Wesley
Patterns of Enterprise Application Architecture	Martin Fowler	1st edition	Addison-Wesley
Enterprise Integration Patterns	Gregor Hohpe	1st edition	Pearson Addison-Wesley Professional
Code Complete	Steve Mcconnell	2nd edition	Microsoft Press US

WILSON COLLEGE (AUTONOMOUS) NEP SYLLABUS FOR BSC.I.T.

PROGRAM: FYBSCIT		**SEMESTER: I**			
Course:- **Operating Systems(Theory)**	**Major**	**Course Code:-WSITCMJ112**			
Teaching Scheme					**Evaluation Scheme**
Lectures (Hours per week)	**Practical (Hours per week)**	**Tutorial (Hours per week)**	**Credit**	**Continuous Internal Assessment (CIA) (Marks-40)**	**Semester End Examination (Marks-60)**
2	**-**	**-**	**2**	**40**	**60**

Learning Objectives:

1) To understand the basic concepts and functions of operating systems.
2) To understand Processes, Threads and Deadlocks
3) To analyze Scheduling algorithms
4) To analyze memory management schemes.
5) To understand I/O management and File systems.
6) To be familiar with the basics of Linux system and Mobile OS like iOS and Android.

Course Outcomes:

At the end of the course, the students should be able to:

1) Analyze various scheduling algorithms.
2) Understand deadlock, prevention and avoidance algorithms.
3) Compare and contrast various memory management schemes.
4) Understand the functionality of file systems.
5) Perform administrative tasks on Linux Servers.
6) Compare iOS and Android Operating Systems

DETAILED SYLLABUS

Course Code/ Unit	Subunit	Course/ Unit Title	Credits/ Lecture’s
I		**Introduction**	
	1.1	**Introduction of Operating-Systems Structures:** Definition of Operating system, Operating System’s role, Operating-System Operations, Functions of Operating System.	3L
	1.2	**Operating-System Structures:** Operating- System Services, User and Operating-System Interface, System Calls, Types of System Calls, Operating-System Structure	3L
II			
	2.1	**Processes:** Process Concept, Process Scheduling, Operations on Processes, Interprocess Communication **Threads:** Overview, Multicore Programming, Multithreading Models **Process Synchronization:** General structure of a typical process, race condition, The Critical-Section Problem, Peterson’s Solution, Synchronization Hardware, Mutex Locks, Semaphores, Classic Problems of Synchronization, Monitors	6L
	2.2	**CPU Scheduling:** Basic Concepts, Scheduling Criteria, Scheduling Algorithms (FCFS, SJF, SRTF, Priority, RR, Multilevel Queue Scheduling, Multilevel Feedback Queue Scheduling), Thread Scheduling **Deadlocks:** System Model, Deadlock Characterization, Methods for Handling Deadlocks, Deadlock Prevention, Deadlock Avoidance, Deadlock Detection, Recovery from Deadlock	5L
III			

	3.1	**Main Memory:** Background, Logical address space, Physical address space, MMU, Swapping, Contiguous Memory Allocation, Segmentation, Paging, Structure of the Page Table **Virtual Memory:** Background, Demand Paging, Copy-on-Write, Page Replacement, Allocation of Frames, Thrashing	04
	3.2	**Mass-Storage Structure:** Overview, Disk Structure, Disk Scheduling, Disk Management **File-System Interface:** File Concept, Access Methods, Directory and Disk Structure, File- System Mounting, File Sharing **File-System Implementation:** File-System Structure, File-System Implementation, Directory Implementation, Allocation Methods, Free-Space Management	5
IV		**Server and Mobile Operation System**	
	4.1	Win 19 R Server operating System: Architecture, Components, Services, Configuration	2
	4.2	Android OS: Architecture, Components, Services, Configuration	2

Books and References:

Sr. No.	Title	Author/s	Publisher	Edition	Year
1	Modern Operating Systems	Andrew S. Tanenbaum, Herbert Bos	Pearson	4th	2014
2	Operating Systems – Internals and Design Principles	Willaim Stallings	Pearson	8th	2009
3	Operating System Concepts	Abraham Silberschatz, Peter B. Galvineg Gagne	Wiley	8th	
4	Operating Systems	Godbole and Kahate	McGraw Hill	3rd	

PROGRAM(s): FYBSCIT		**SEMESTER: I**			
Course:- **Programming Principles with C (Practical)**	**Major**	**Course Code: WSITCMJ113**			
Teaching Scheme					**Evaluation Scheme**
Lectures (Hours per week)	**Practical (Hours per week)**	**Tutorial (Hours per week)**	**Credit**	**Continuous Internal Assessment (Marks- 40)**	**Semester End Examination (Marks- 60)**
-	**4**	**-**	**2**	**40**	**60**

Learning Objectives:

1) To develop the logical ability of the student.
2) Basic concepts to be cleared using suitable examples.
3) Different approach towards the problem.
4) To handle the errors and find suitable solutions.
5) Debugging the code.

Course Outcomes:

1) Learn the basic principles of programming.
2) Develop logic using algorithms and flowchart.
3) Acquire the information about data types.
4) Understanding of input and output functions.
5) Enhance advanced concepts using programs.

Course Code/Unit	**Sub unit**	**Course/ Unit Title**	**Credits** **2**
	1	a.Write an algorithm and draw flowchart to print the given no. is even or odd. b.Write an algorithm and draw flowchart for sum of 1 to 5 numbers.	
	2	a. Write a program using a while loop to reverse the digits of a number. b. Write a program to calculate the factorial of a given number.. Write a program to print the Fibonacci series.	

	3	a. Write a program in C to check entered character vowel or consonant b. Write a program to C program to print day name of week using switch-case. c.Write a program to read three values from keyboard and print out the largest of them without using if statement.	
	4	a. Write a program to print the pattern of asterisks as shown below : * * * * * * * * * * b. Write a program to print the pattern of asterisks as shown below : * * * * * * * * * * * * * * *	
	5	a. Write a program to print area of square using function. b. Write a program using recursive function. c. Write a program to square root, abs() value using function.	
	6	a. Write a program to print roll no and names of 10 students using an array. b. Write a program to sort the elements of an array in ascending or descending order.	
	7	b. Write a program to find if the given string is palindrome or not.	
	8	a. Write a program to display the values using different data types and its address using a pointer. b. Write a program to perform addition and subtraction using a pointer.	
	9	Write a program to print the structure using Title Author	

		Subject Book ID Print the details of two students.	
	10	Create a mini project on “Bank management system” . The program should be menu driven.	

Wilson College

FIDES SPES CARITAS

विश्वास आशा प्रेम

PROGRAM: FYBSCIT		**SEMESTER: I**			
Course:Digital Electronics(Theory)		**Course Code: WSITCMN111**			
Teaching Scheme					**Evaluation Scheme**
Lectures (Hours per week)	**Practical (Hours per week)**	**Tutorial (Hours per week)**	**Credit**	**Continuous Internal Assessment (CIA) (Marks- 40)**	**Semester End Examination (Marks- 60)**
2	**-**	**_**	**2**	**40**	**60**

Learning Objectives:

1) Introduce the concept of digital and binary systems.
2) Be able to design and analyze combinational logic circuits.
3) Be able to design and analyze sequential logic circuits.
4) Understand the basic software tools for the design and implementation of digital circuits and systems.
5) Reinforce theory and techniques taught in the classroom through experiments and projects in the laboratory

Course Outcomes:

After the end of the course, the learner will be able to:

1) Examine the structure of number systems and perform the conversion among different number systems
2) Illustrate reduction of logical expressions using boolean algebra, k-map and tabulation method and implement the functions using logic gates
3) Realize combinational circuits for given application
4) Design and analyzes synchronous and asynchronous sequential circuits using flip-flops
5) Implement combinational logic circuits using programmable logic devices

DETAILED SYLLABUS

Course Code/ Unit	Sub unit	Course/ Unit Title	Lectures
I		**Introduction to Number system**	
	1.1	**Number System**: numbering systEm, binary number system, octal number system, hexadecimal number system,	3
	1.2	Binary Arithmetic: Binary addition, Binary subtraction,Negative number representation, Subtraction using 1's complement and 2's complement, Binary multiplication and division,BCD multiplication and division	4
II		**Boolean Algebra and Logic Gates:**	
	2.1	Boolean Algebra and Logic Gates: Introduction, Logic (AND OR NOT), Boolean theorems, Boolean Laws, De Morgan's Theorem, Perfect Induction, Reduction of Logic expression using Boolean Algebra, Deriving	4
	2.2	Boolean expression from given circuit, exclusive OR and Exclusive NOR gates, Universal Logic gates,Implementation of other gates using universal gates.	4
III		**Code Conversion and Arithmetic Circuits**	
	3.1	Conversion from one number system to another,floating point numbers, weighted codes binary coded decimal, non-weighted codes Excess – 3 code, Gray code, Code conversion. Error detection and correction.	4
	3.2	Arithmetic Circuits: Introduction, Adder, BCD Adder, Binary Subtractors, BCD Subtractor,	4

		Comparator.	
IV		**Combinational Logic Circuits**	
	4.1	Combinational Logic Circuits: Introduction, Multi-input, multi-output Combinational circuits, Code converters design and implementations.	4
	4.2	Multiplexer, Demultiplexer, Encoder and Decoder	3

References:

1. Digital Principles and Applications Malvino and Leach Tata McGraw Hill
2.2. Make Electronics Charles Platt O’Reilly 1 st 2010
3. Modern Digital Electronics R. P. Jain Tata McGraw Hill 3 rd
4. Digital Electronics: Principles, Devices and Applications, Anil K. Maini Wiley 2007
5.Digital Electronics and Logic Design N. G. Palan Technova

PROGRAM: FYBScIT		**SEMESTER: II**			
Course: Computer Networks		**Course Code: WSITCMN112**			
Teaching Scheme					**Evaluation Scheme**
Lectures (Hours per week)	**Practical (Hours per week)**	**Tutorial (Hours per week)**	**Credit**	**Continuous Internal Assessment (CIA) (Marks- 40)**	**Semester End Examination (Marks- 60)**
2	**-**	**_**	**2**	**40**	**60**

Learning Objectives:

1) To introduce students to the tools, techniques, components used for communication.
2) To understand how digital communication works.
3) Introduce building the network topology

Course Outcomes:

After the end of the course, the learner will be able to:

1) Gain knowledge, understanding of communication history, and evolution.
2) Development of Mass Communication with networking.
3) Develop the I.T Infrastructure

DETAILED SYLLABUS

Course Code/ Unit	Subunit	Course/ Unit Title	Credits/ Lecture’s
I		**Introduction to Computer Network**	
	1.1	Computer Network, Evolution of Computer Networks Different types of Computer Network, Difference between LAN, MAN and WAN, Hardware Devices used for Networking: Network Interface Card (NIC), Modem, Hub, Switch L1 and L2 switches, Comparison between switch and hub, Bridge, Router, Gateway.	6L

	1.2	Standards and administration. Network Models: Protocol layering, TCP/IP protocol suite, The OSI model.	5L
II		**Network Layer**	
	2.1	IPv4 Addresses, IPv4 Protocol, ARP,ICMP, IPv6	3L
	2.2	RIP, OSPF, BGP	3L
III		**Transport Layer**	
	3.1	UDP, TCP	2L
	3.2	Wireless LANs: Introduction, IEEE 802.11 project, Bluetooth, WiMAX, Cellular telephony, Satellite networks.	3L
IV		**Application Layer & Introduction to Network Security**	
	4.1	WWW, HTTP, SMTP, POP3, MIME, IMAP	2L
	4.2	DHCP, TELNET, SSH, FTP,	2L
	4.3	Attacks on Computer Network	2L
	4.4	Cryptography and Network Security, Internet Security	2L

References:

1. TCP/IP Protocol Suite Behrouz A. Forouzan Tata McGraw Hill Fourth Edition 2010
2. Data Communication and Networking Behrouz A. Forouzan Tata McGraw Hill Fifth Edition 2013
3. Computer Networks Andrew Tanenbaum Pearson Fifth 2013

PROGRAM: FYBScIT		**SEMESTER: II**			
Course:-Computer Networks Practicals		**Course Code: WSITCMN113**			
Teaching Scheme					**Evaluation Scheme**
Lectures (Hours per week)	**Practical (Hours per week)**	**Tutorial (Hours per week)**	**Credit**	**Continuous Internal Assessment (CIA) (Marks- 40)**	**Semester End Examination (Marks- 60)**
-	**4**	**_**	**2**	**40**	**60**

Learning Objectives:

1) To introduce students to the tools, techniques, components used for communication.
2) To understand how digital communication works.
3) Introduce building the network topology

Course Outcomes:

After the end of the course, the learner will be able to:

1) Gain knowledge, understanding of communication history, and evolution.
2) Development of Mass Communication with networking.
3) Develop the I.T Infrastructure

DETAILED SYLLABUS

Unit	Subunit	Course/ Unit Title	Credits(2)
1		Understanding IPv4 and IPv6	
2		Configure RIP	
3		Configure OSPF	
4		Configure Static routing	
5		Configure DHCP	
6		Configure DNS	
7		Ping, Traceroute, arp command	
8		Use wireshark for HTTP & ICMP Trace	
9		Configure Authentication & authorization	
10		Security Checkup	

PROGRAM: FYBScIT		**SEMESTER: I**			
Course: Fundamentals of Computers (Elective-I)		**Course Code: WSITCOE111**			
Teaching Scheme					**Evaluation Scheme**
Lectures (Hours per week)	**Practical (Hours per week)**	**Tutorial (Hours per week)**	**Credit**	**Continuous Internal Assessment (CIA) (Marks- 40)**	**Semester End Examination (Marks- 60)**
4	**-**	**_**	**2**	**40**	**60**

Learning Objectives:

1) To understand the structure, function and characteristics of computer systems.
2) To understand the design of the various functional units and components of computers.
3) To identify the elements of modern instruction sets and their impact on processor design.
4) To explain the function of each element of a memory hierarchy.
5) To identify and compare different methods for computer I/O.

Course Outcomes:

1) Understand basic concepts and terminology of information technology.
2) Have a basic understanding of personal computers and their operations.
3) Understand various software and hardware, and various security issues.
4) Familiarize students with complete fundamentals and the packages commonly used in computing software.
5) Gain writing skills and various presentation aspects using word processing software**.**

DETAILED SYLLABUS

Unit	Subunit	Course/ Unit Title	Lectures
I		**Fundamentals of Computer:**	
	1.	**Introduction to Computer:** Definition, Characteristics of Computers, Basic Applications of Computer, Generations of	**5**

		computers **Components of Computer System:** Central Processing Unit (CPU), input devices & output devices, Advantages of computer system	
	2.	**Computer Memory:** primary and secondary memory **Data processing:** concepts of data processing, Definition of Information and data,	**3**
II	1.	**Operating System:** Definition & functions, Characteristics of Operating system .	**2**
	2.	**Basics of Windows.** Multiprogramming, Multi Tasking, Multiprocessing, Time Sharing.	**2**
III	1.	**Computer Networking:**Introduction of computer network, Classifications of computer network, uses of computer network, advantages & dis-advantages of computer network.	**4**
	2.	**Components of Computer Network:** Classification of Computer Network Based on Topology.	**4**
IV	1.	**Data Representation:** Binary Number system, Octal, Hexadecimal, Decimal and their conversion, Complements.	**5**
	2.	**Introduction to MS Office:** MS Word,MS PowerPoint,MS Excel (Introduction,Features and Applications).	**5**

PROGRAM: FYBScIT		**SEMESTER: I**			
Course: Awareness of cyber Security(Elective-II)		**Course Code: WSITCOE112**			
Teaching Scheme					**Evaluation Scheme**
Lectures (Hours per week)	**Practical (Hours per week)**	**Tutorial (Hours per week)**	**Credit**	**Continuous Internal Assessment (CIA) (Marks- 40)**	**Semester End Examination (Marks- 60)**
2	**-**	**_**	**2**	**40**	**60**

Learning Objectives:

1. Student will be able to understand network
2. Will be able to differentiate between the different types of networks and the ip ranges
3. Student will be able to understand the types of attack related to cyber world
4. Student will be able to understand the legal action against the infringement of cyber law

Course Outcomes:

After the end of the course, the learner will be able to:

1. Will be able to create their own network infrastructure
2. Will be able to recognize the attacks
3. Will be able to examine the cyber security cases

DETAILED SYLLABUS

Unit	Subunit	Course/ Unit Title	Lectures
I		Introduction to computer network IPv4 & IPv6 addressing, Subnetting, Subnet mask	**3**
II		Standard & Protocol used for networking transmission Control Protocol (TCP), Internet Protocol (IP), User Datagram Protocol (UDP) and network standards such as TCP/IP, HTTP, FTP,	**5**

III		Different Types of attacks Phishing, Malware (Malicious Software)Ransomware, Spyware, Trojans, Viruses, Distributed Denial of Service (DDoS)	**10**
IV		Cyber Laws and Security	
		Introduction to cyber security, cyber security Ethics,web Technology and Security,Introduction to cryptography.	**12**

PROGRAM: FYBScIT		**SEMESTER: I**			
Course:Operating system Practicals		**Course Code: WSITCSE111**			
Teaching Scheme					**Evaluation Scheme**
Lectures (Hours per week)	**Practical (Hours per week)**	**Tutorial (Hours per week)**	**Credit**	**Continuous Internal Assessment (CIA) (Marks- 40)**	**Semester End Examination (Marks- 60)**
-	**4**	**_**	**2**	**40**	**60**
Learning Objectives:					
Course Outcomes: After the end of the course, the learner will be able to:					

DETAILED SYLLABUS

Course code WUSITC2P 2	**Operating system Practicals**	**Credit s**
1	**Installation of virtual machine software**	2
2	**Installation of Linux operating system (RedHat / Ubuntu) on virtual machine.**	
3	**Installation of Windows operating system on virtial machine.**	
4	**Linux commands:** Working with Directories: a. pwd, cd, absolute and relative paths, ls, mkdir, rmdir, b. file, touch, rm, cp. mv, rename, head, tail, cat, tac, more, less, strings, chmod	

5	**Linux commands:** Working with files: a. ps, top, kill, pkill, bg, fg, b. grep, locate, find, locate. c. date, cal, uptime, w, whoami, finger, uname, man, df, du, free, whereis, which. d. Compression: tar, gzip	
6	**Windows (DOS) Commands** – 1 a. Date, time, prompt, md, cd, rd, path. b. Chkdsk, copy, xcopy, format, fidsk, cls, defrag, del, move.	
7	**Windows (DOS) Commands – 2** a. Diskcomp, diskcopy, diskpart, doskey, echo b. Edit, fc, find, rename, set, type, ver	
8	**Working with Windows Desktop and utilities** a. Notepad b. Wordpad c. Paint d. Taskbar e. Adjusting display resolution f. Using the browsers g. Configuring simple networking h. Creating users and shares	
9	**9. Working with Linux Desktop and utilities** a. The vi editor. b. Graphics c. Terminal d.Adjusting display resolution e. Using the browsers f. Configuring simple networking g. Creating users and shares	
10	**Installing utility software on Linux and Windows**	

Wilson College

FIDES SPES CARITAS

विश्वास आशा प्रेम

SEMESTER- II

WILSON COLLEGE (AUTONOMOUS) NEP SYLLABUS FOR BSC.I.T.

PROGRAM: FYBSCIT	**SEMESTER: II**				
Course:OOPs with C++	**Course Code: WSITCMJ121**				
Teaching Scheme					**Evaluation Scheme**
Lectures (Hours per week)	**Practical (Hours per week)**	**Tutorial (Hours per week)**	**Credit**	**Continuous Internal Assessment (CA)(Marks-40)**	**Semester End Examination (Marks-60)**
2	**-**	**_**	**2**	**40**	**60**

Learning Objectives:

1) To make students familiar with object-oriented programming language.
2) To introduce the concept and process of writing and editing in oops.
3) To introduce students to different ways of delivering the information with data structure, inheritance, polymorphism concept

Course Outcomes:

After the end of the course, the learner will be able to:

1) Understand and incorporate the different writing styles in OOPS.
2) Employ the concepts OOPS.
3) Convert the textual data into machine readable data.

DETAILED SYLLABUS

Unit	**Subunit**	**Course/ Unit Title**	**Lecture s**
I	1.1	**Object Oriented Methodology:** Introduction, Advantages and Disadvantages of Procedure Oriented Language, what is Object Oriented? What Is Object Oriented Development? Object Oriented Themes, Benefits and Application of OOPS	3L

	1.2	**Principles of OOPS:** OOPS Paradigm, Objects, Classes, Data Abstraction and Data Encapsulation, Inheritance, Polymorphism, Dynamic Binding.	2L
	1.3	**Classes and Objects**: Simple classes (Class specification, class members accessing), Defining member functions, passing object as an argument, Returning object from functions, friend classes	3L
II		**Constructor, Destructor, Polymorphism, Virtual Function**	
	2.1	Constructors and Destructors: Introduction, Default Constructor, Parameterized Constructor and examples, Destructors	3L
	2.2	Polymorphism: Concept of function overloading, overloaded operators, overloading unary and binary operators, overloading comparison operator,overloading arithmetic assignment operator, Data Conversion between objects and basic types	3L
		Virtual Functions: Introduction and need, Pure Virtual Functions, Static Functions, this Pointer,abstract classes, virtual destructors	4L
III		**Inheritance & Exception Handling**	
		Inheritance: Introduction, understanding inheritance, Advantages provided by inheritance, choosing the access specifier, derived class constructors, class hierarchies, multiple inheritance, multilevel inheritance.	4L
		Exception Handling: Introduction, Exception Handling Mechanism, Concept of throw & catch	2L
IV		**Files**	
		Working with Files: Introduction, File Operations, Various File Modes.	2L

		Conditional Statements: Order of Operations. if,if- else, nested if –else **Looping:** for, while, nested loops.	2L
		Templates and Conditional Statement:- Templates: Introduction, Function Template and examples, Class Template and examples	2L

References:-

1. Object Oriented Analysis and Design Timothy Budd TMH 3rd 2012
2. Mastering C++ K R Venugopal, Rajkumar Buyya, T Ravishankar Tata McGraw Hill 2nd Edition 2011 3. C++ for beginners B. M. Hirwani SPD 2013
3. Effective Modern C++ Scott Meyers SPD
4. Object Oriented Programming with C++ E. Balagurusamy Tata McGraw Hill 4th
5. Learning Python Mark Lutz O’ Reilly 5th 2013
6. Mastering Object Oriented Python Steven F. Lott Pact Publishing 2014

PROGRAM: FYBSCIT		**SEMESTER: II**			
Course:Web Designing & development		**Course Code: WSITCMJ122**			
Teaching Scheme					**Evaluation Scheme**
Lectures (Hours per week)	**Practical (Hours per week)**	**Tutorial (Hours per week)**	**Credit**	**Continuous Internal Assessment (CA) (Marks-40)**	**Semester End Examination (Marks-60)**
2		**_**	**2**	**40**	**60**

Learning Objectives:

1) To make students familiar with object-oriented programming language.
2) To introduce the concept and process of writing and editing in oops.
3) To introduce students to different ways of delivering the information with data structure, inheritance, polymorphism concept

Course Outcomes:

After the end of the course, the learner will be able to:

1) Understand and incorporate the different writing styles in OOPS.
2) Employ the concepts OOPS.
3) Convert the textual data into machine readable data.

DETAILED SYLLABUS

Unit	Details	Lectures

I	**Introduction, Basic Principles of Web Development** Planning: Contemplate and appraise client requirements, identifying technologies used for website development, Estimating all the milestones and the costs for the project Golden Rules of Web Designing Designing Navigation Bar: Steps to create simple top navigation bar, Creating a responsive Top Navigation	7L
II	**HTML5** Why HTML5? Introduction, Formatting text by using tags, using lists and backgrounds, Creating hyperlinks and anchors. Style sheets, CSS formatting text using style sheets, formatting paragraphs using style sheets. HTML5 Tables, Forms and Media: Creating tables: creating simple table, specifying the size of the table, specifying the width of the column, merging table cells, using tables for page layout, formatting tables: applying table borders, applying background and foreground fills, changing cell padding, spacing and alignment, creating user forms: creating basic form, using check boxes and option buttons, creating lists, additional input types in HTML5	8L
III	**Introduction Creating Style Sheet CSS** Syntax CSS – Inclusion CSS Rules Overriding Handling old browsers CSS Comments CSS Properties Text Properties List Properties Border Properties Font Properties CSS Styling Background Text Format, Controlling Fonts Working with block elements and objects, Working with Block elements (BOX Model), Setting the Margins, setting padding, Working with borders Styling the content using as a block, Working with Objects, Working with Lists and Tables	7L

IV	**Java Script** Java Script Introduction: Introduction, Client-Side JavaScript, Server-Side JavaScript, JavaScript Objects, JavaScript Security Operators: Assignment Operators, Comparison Operators, Arithmetic Operators, % (Modulus), ++(Increment), --(Decrement), -(Unary Negation), Logical Operators, Short-Circuit Evaluation, String Operators, Special Operators, ?: (Conditional operator), (Comma operator), delete, new, this, void Statements: Break, comment, continue, delete, do…while, export, for, for...in, function, if...else, import, labeled, return, switch, var, while, with JavaScript (Properties and Methods of Each) : Array, Boolean, Date, Function, Math, Number, Object, String, regExp Document and its associated objects: document, Link, Area, Anchor, Image, Applet, Layer Eve	8L

References:

1. https://www.geeksforgeeks.org/html5-introduction/

PROGRAM: FYBSCIT		**SEMESTER: II**			
Course:OOPs with C++ Practicals		**Course Code: WSITCMJ123**			
Teaching Scheme					**Evaluation Scheme**
Lectures (Hours per week)	**Practical (Hours per week)**	**Tutorial (Hours per week)**	**Credit**	**Continuous Internal Assessment(CA) (Marks- 40)**	**Semester End Examination (Marks- 60)**
4		**_**	**2**	**40**	**60**

Learning Objectives:

1) To make students familiar with object-oriented programming language.
2) To introduce the concept and process of writing and editing in oops.
3) To introduce students to different ways of delivering the information with data structure, inheritance, polymorphism concept

Course Outcomes:

After the end of the course, the learner will be able to:

1) Understand and incorporate the different writing styles in OOPS.
2) Employ the concepts OOPS.
3) Convert the textual data into machine readable data.

DETAILED SYLLABUS

Course code WUSITC2P1	**OOPS WITH C++ Practical's**	**Credits(2)**
1	Programs based on Classes and methods.	
2	Programs based on friend functions.	

3	Programs based on Constructors and method overloading.	
4	Programs based on Operator Overloading.	
5	Programs based on Inheritance.	
6	Programs based on Virtual functions and abstract classes.	
7	Programs based on String handling.	
8	Programs based on Exception handling.	
9	Programs based on File handling.	
10	Programs based on Templates.	

WILSON COLLEGE (AUTONOMOUS) NEP SYLLABUS FOR BSC.I.T.

PROGRAM: FYBSCIT		**SEMESTER: II**			
Course:- Microprocessor and Architecture		**Course Code: WSITCMN121**			
Teaching Scheme					**Evaluation Scheme**
Lectures (Hours per week)	**Practical (Hours per week)**	**Tutorial (Hours per week)**	**Credit**	**Continuous Internal Assessment (CIA) (Marks- 40)**	**Semester End Examination (Marks- 60)**
2	**-**	**_**	**4**	**40**	**60**

Learning Objectives:

1) Understand the fundamentals of Microprocessors.
2) Understand the internal design of 8085 microprocessor
3) Design different interfacing applications using microcontrollers and peripherals.
4) Build systems using microprocessor for real time applications

Course Outcomes:

After the end of the course, the learner will be able to:

1) Describe the architecture & organization of 8085 Microprocessor.
2) Understand and classify the instruction set o f 8085/8086 microprocessor and distinguish the use of different instructions and apply it in assembly language programming.
3) Relate the addressing modes used in the instructions.
4) Familiarize the architecture and operation of Programmable Interface Devices and realize the programming & interfacing of it with 8085 microprocessor.

DETAILED SYLLABUS

Unit	Details	Lectures

I	**Microprocessor, microcomputers, and Assembly Language:** Microprocessor, Microprocessor Instruction Set and Computer Languages, From Large Computers to Single-Chip Microcontrollers, Applications. Microprocessor Architecture and Microcomputer System: Microprocessor Architecture and its operations, Memory, I/O Devices, Microcomputer System, Logic Devices and Interfacing, Microprocessor-Based System Application. 8085 Microprocessor Architecture and Memory Interface: Introduction, 8085 Microprocessor unit, 8085-Based Microcomputer, Memory Interfacing, Interfacing the 8155 Memory Segment, Illustrative Example: Designing Memory for the MCTS Project, Testing and Troubleshooting Memory Interfacing Circuit, 8085-Based Single-Board microcomputer.	7L
II	**Interfacing of I/O Devices** Basic Interfacing concepts, Interfacing Output Displays, Interfacing Input Devices, Memory Mapped I/O, Testing and Troubleshooting I/O Interfacing Circuits. Introduction to 8085 Assembly Language Programming: The 8085 Programming Model, Instruction Classification, Instruction, Data and Storage, Writing assembling and Execution of a simple program, Overview of 8085 Instruction Set, Writing and Assembling Program. Introduction to 8085 Instructions: Data Transfer Operations, Arithmetic Operations, Logic Operation, Branch Operation, Writing Assembly Languages Programs, Debugging a Program.	8L

III	**Programming Techniques With Additional Instructions:** Programming Techniques: Looping, Counting and Indexing, Additional Data Transfer and 16-Bit Arithmetic Instructions, Arithmetic Instruction Related to Memory, Logic Operations: Rotate, Logics Operations: Compare, Dynamic Debugging. Counters and Time Delays: Counters and Time Delays, Illustrative Program: Hexadecimal Counter, Illustrative Program: zero-to-nine (Modulo Ten) Counter, Generating Pulse Waveforms, Debugging Counter and Time-Delay Programs. Stacks and Subroutines: Stack, Subroutine, Restart, Conditional Call, Return Instructions, Advanced Subroutine concepts.	7L
IV	**Code Conversion, BCD Arithmetic, and 16-Bit Data Operations:** BCD-to-Binary Conversion, Binary-to-BCD Conversion, BCD-to- Seven-Segment-LED Code Conversion, Binary-to-ASCII and ASCII- to-Binary Code Conversion, BCD Addition, BCD Subtraction, Introduction To Advanced Instructions and Applications, Multiplication, Subtraction With Carry. **Software Development System and Assemblers:** Microprocessors-Based Software Development system, Operating System and Programming Tools, Assemblers and Cross-Assemblers, Writing Program Using Cross Assemblers.	8L

Books and References:

Sr. No. Title Author/s Publisher Edition Year

1. Microprocessor Architecture, Programming and Applications with the 8085 Ramesh Gaonkar PENRAM Fifth 2012
2. Computer System Architecture M. Morris Mano PHI 1998
3. Structured Computer Organization Andrew C. Tanenbaum PHI

PROGRAM: FYBSCIT	**SEMESTER: II**				
Course:- Embedded System	**Course Code: WSITCMN122**				
Teaching Scheme					**Evaluation Scheme**
Lectures (Hours per week)	**Practical (Hours per week)**	**Tutorial (Hours per week)**	**Credit**	**Continuous Internal Assessment (CIA) (Marks- 40)**	**Semester End Examination (Marks- 60)**
2	**-**	**_**	**2**	**40**	**60**

Learning Objectives:

1) To become aware of the core characteristics and quality attributes of embedded systems
2) To learn application specific and domain specific embedded systems, the embedded hardware and peripherals.
3) To design and program embedded system with 8051 microcontroller
4) To design and develop using the real time operating system

Course Outcomes:

After the end of the course, the learner will be able to:

1) gain knowledge about the history, classifications, applications and purpose of embedded systems.
2) become aware of application specific embedded systems
3) microcontroller and designing it, compile, link and debug an embedded program
4) real-time characteristics of operating systems, embedded system environment

DETAILED SYLLABUS

Unit	Details	Lectures
I	**Introduction:** Embedded Systems and general purpose computer systems, history, classifications, applications and purpose of embedded systems **Core of embedded systems:** microprocessors and microcontrollers, RISC and CISC controllers, Big endian and Little endian processors, Application specific ICs, Programmable logic devices, COTS, sensors and actuators, communication interface, embedded firmware, other system	**7L**

	components. **Characteristics and quality attributes of embedded systems:** Characteristics, operational and non-operational quality attributes.	
II	**Introduction to Embedded Systems –** Microprocessor & Microcontroller, Architecture of Microprocessor & Microcontroller **Application and Domain Specific:** Application specific – washing machine, domain specific - automotive. **Embedded Hardware:** Memory map, i/o map, interrupt map, processor family, external peripherals, memory – RAM , ROM, types of RAM and ROM, memory testing, CRC ,Flash memory. **Peripherals:** Control and Status Registers, Device Driver, Timer Driver - Watchdog Timers.	**8L**
III	**The 8051 Microcontrollers:** Microcontrollers and Embedded processors, Overview of 8051 family. 8051 Microcontroller hardware, Input/output pins, Ports, and Circuits, External Memory. **8051 Programming in C:** Data Types and time delay in 8051 C, I/O Programming, Logic operations, Data conversion Programs.	**7L**
IV	**Designing Embedded System with 8051 Microcontroller:** Factors to be considered in selecting a controller, why 8051 Microcontroller, Designing with 8051. **Programming embedded systems:** structure of embedded program, infinite loop, compiling, linking and debugging.	**8L**

References:

1. https://realtoughcandy.com/best-embedded-systems-books/

WILSON COLLEGE (AUTONOMOUS) NEP SYLLABUS FOR BSC.I.T.

PROGRAM: FYBSCIT		**SEMESTER: I**			
Course:Embedded Systems Practicals		**Course Code: WSITCMN123**			
Teaching Scheme					**Evaluation Scheme**
Lectures (Hours per week)	**Practical (Hours per week)**	**Tutorial (Hours per week)**	**Credit**	**Continuous Internal Assessment (CIA) (Marks- 40)**	**Semester End Examination (Marks- 60)**
-	**4**	**_**	**2**	**40**	**60**

Learning Objectives:

1. To configure the embedded system 8051
2. To demonstrate the use of embedded system
3. To understand the different port and connection with circuit

Course Outcomes:

After the end of the course, the learner will be able to:

1. Student will be able to understand the circuit
2. Student will be able to configure the circuit

Unit	**Subunit**	**Course/ Unit Title**	**Credits (2)**
1		Design and develop a reprogrammable embedded computer using 8051 microcontrollers and to show the following aspects. a. Programming b. Execution c. Debugging	
2		Configure timer control registers of 8051 and	

		develop a program to generate given time delay.	
3		Port I / O: Use one of the four ports of 8051 for O/P interfaced to eight LED’s. Simulate binary counter (8 bit) on LED’s	
4		Interface 8051 with D/A converter and generate triangular wave of given frequency on an oscilloscope.	
5		To demonstrate timer working in timer mode and blink LED without using any loop delay routine.	
6		To interface 8 LEDs at Input-output port and create different patterns.	
7		Serial I / O: Configure 8051 serial port for asynchronous serial communication with serial port of PC exchange text messages to PC and display on PC screen. Signify end of message by carriage return.	
8		To demonstrate interfacing of seven-segment LED display and generate counting from 0 to 99 with fixed time delay.	
9		To demonstrate use of general purpose port i.e. Input/ output port of two controllers for data transfer between them.	
10		Interface stepper motor with 8051 and write a program to move the motor through a given angle	

		in clockwise or counter-clockwise direction.	

WILSON COLLEGE (AUTONOMOUS) NEP SYLLABUS FOR BSC.I.T.

PROGRAM: FYBSCIT	**SEMESTER: II**				
Course:Website Design(Elective-I)	**Course Code: WSITCOE121**				
Teaching Scheme					**Evaluation Scheme**
Lectures (Hours per week)	**Practical (Hours per week)**	**Tutorial (Hours per week)**	**Credit**	**Continuous Internal Assessment (CIA) (Marks-40)**	**Semester End Examination (Marks-60)**
2	**_**	**_**	**2**	**40**	**60**

Learning Objectives:

Understand basic concepts of Internet and World Wide Web. Comprehend different HTML elements that can be used to develop static web pages. Become familiar with concept of stylesheets and various CSS effects. Peruse JavaScript as a tool to add dynamism to static HTML pages. Explore how server-side script works on the web.

Course Outcomes:

After the end of the course, the learner will be able to:

Analyze working of Internet. Gain an insight into designing web pages. Use different ways of styling web pages using CSS. Implement the basics of JavaScript in a web page.

Unit	**Sub-unit**	**Course/ Unit Title**	**Lecture s**
I	1.1	What is Internet? Applications of Internet, E-mail, Telnet, FTP, E-commerce and Ebusiness. Internet Service Providers, Domain Name Server, Internet Address,	3L
	1.2	World Wide Web (WWW): World Wide Web and its Evolution, Uniform Resource Locator 12 (URL), Browsers, Common Features of Browsers, Search Engine, Web Server, HTTP	05

		Protocol.	
		HTML:	
II	2.1	Introduction, Formatting Text by using Tags, Using Lists, Creating Hyperlinks and Bookmarks, Redirecting to another URL	**7**
		CSS:	
III	3.1	**I**mplementing Styles using CSS – Stylesheets, Formatting Text and Links using CSS, CSS Selectors, Changing Background, Adding Border, Margin and Padding, Setting Dimensions, Using Inline Container to mark up a part of a text.	**4**
	3.2	Arithmetic Operators,Conditional Statements –	**4**
		HTML Page Layout	
IV	4.1	Using Layout Elements, Semantic Elements, Creating, Positioning and Formatting Divisions, Floating Divisions next to each other, Responsive Web Design, Inline Frames.	**4**
	4.2	Introduction, Difference between Client-side and Server-side Scripting, JavaScript Variables and Constants, Data Types.	**3**

WILSON COLLEGE (AUTONOMOUS) NEP SYLLABUS FOR BSC.I.T.

PROGRAM: FYBSCIT	**SEMESTER: II**
Course:Digital Electronics Practicals	**Course Code: WSITCSE121**
Teaching Scheme	**Evaluation Scheme**

Lectures (Hours per week)	Practical (Hours per week)	Tutorial (Hours per week)	Credit	Continuous Internal Assessment(CA) (Marks- 40)	Semester End Examination (Marks- 60)
4		–	2	40	60

DETAILED SYLLABUS

UNIT	Digital Electronics Practical's	Credits 2
1.	Study of Logic gates using IC (AND, OR, NOT, XOR, NAND and NOR)	
2	NAND gate as universal gate (Implement AND, OR, NOT, XOR using NAND gates)	
3	NOR gate as universal gate (Implement AND, OR, NOT, XOR, XNOR using NOR gates.)	
4	Verifying De Morgan's laws.	
5	Design and implement Half adder and Full adder.	
6	Implement Adder and Subtractor Arithmetic circuits.	
7	Design and implement Binary – to – Gray code converter & Gray – to – Binary code converter.	
8	Design and implement Binary – to – BCD code converter	
9	Design and implement Binary – to – XS-3 code converter	
10	Design and implement 8:3 encoder.	
11	Design and implement 3:8 decoder. Implement the given expression using IC 74138 3:8 decoder.	

12	**Design and implement 4:1 multiplexer. Study of IC 74153, 74157. Implement the given expression using IC 74151 8:1 multiplexer.**	
13	**Design and implement 1:4 demultiplexer. Study of IC 74139**	

Modality of Assessment

Theory Examination Pattern: (for Discipline Specific Core papers)

A. Internal Assessment- 40%- 40 Marks per paper

Sr. No.	Evaluation Type	Marks
1	**Assignment/ Case study/ field visit report/ presentation/ project**	**40**
	Total	**40**

A.External Examination- 60%- 60 Marks per paper (for Discipline Specific Core papers) Semester End Theory Examination:

1. Duration - These examinations shall be of **two hours** duration.
2. Theory question paper pattern:

a) There shall be 3 questions each of 20 marks with different levels of difficulty.

b) All questions shall be compulsory with at least 50% internal choice within the questions. (For example, 4 out of 6 sub-questions or 3 out of 6 sub-questions to be solved).

c) All units will be given equal weightage.

Evaluation for Discipline Specific Core Practical Course:

A. Internal Examination: 40%- 40 Marks

Particulars	Marks
Regular assessment, performance and journal keeping	40
Total	**40**

B. External Examination: 60%- 60 Marks

Particulars	Marks
Practical exam	50

Viva	10
Total	**60**

PRACTICAL BOOK/JOURNAL

The students are required to perform a minimum number of practicals mentioned in the syllabus for certification of journal.

The students are required to present a duly certified journal for appearing at the practical examination, failing which they will not be allowed to appear for the examination.

Examination Pattern: (for Open Elective/Generic Elective) It will be assessment for total of 60 marks

Sr. No.	Evaluation Type	Marks
1	Assignment/ Case study/ field visit report/ presentation/ project	30
2	Assignment/ Case study/ field visit report/ presentation/ project	30
	Total	**60**

Examination Pattern: (SEC) Skill Enhancement Course is a practical course. There will be one single External Examination of 60

Marks

Particulars	Marks
Practical exam	50
Viva	10
Total	**60**