Detailed Syllabus of Minor Courses

Programme	B. Sc. Computer Science				
Course Code	CSC1MN101				
Course Title	Exploring Computer Basics & Computational Thinking				
Type of Course	**Minor**				
Semester	I				
Academic Level	100-199				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	1. Foundation on Mathematics at Plus Two level 2. Foundation on Basic Science at Plus Two Level				
Course Summary					

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Understanding of computer hardware, software, and basic operation principles	U	C	Exams/ Assignments/ Quizzes/ Seminars/ Practical
CO2	Understand and identify computer hardware components	U, Ap	C	Exam/ Assignments/ Quizzes/ Seminars/ v
CO3	Understand how data is represented and manipulated within a computer system.	U	C	Exam/ Assignments/ Quizzes/ Seminars
CO4	Understand the basics of computer languages, operating systems, and their comparison	U	C	Exam/ Assignments/ Quizzes/ Seminars

CO5	Learn to design and implement algorithms to solve simple computational problems.	U	P	Exam/ Assignments/ Quizzes/ Seminars/ / Practical
CO5	Develop computational thinking skills essential for problem-solving in various domains	Ap	P	Exam/ Assignments/ Quizzes/ Seminars/ / Practical

* - Remember (R), Understand (U), Apply (Ap), Analyse (An), Evaluate (E), Create (C)

- Factual Knowledge(F) Conceptual Knowledge (C) Procedural Knowledge (P) Metacognitive Knowledge (M)

Detailed Syllabus:

Module	Unit	Content	Hrs	Marks
I	**History, Evolution of Computers, and Number System**		**8**	**15**
	1	Introduction to Computers, Characteristics of Computers	1	
	2	Generations of Computers	1	
	3	Classification of Computers: Super Computers, Main Frame Computers, Mini Computers, Micro Computers	1	
	4	Number Systems (Binary, Decimal, Octal, Hexadecimal) and Conversion	3	
	5	Computer Codes: BCD Code, Excess 3 Code, ASCII Code, Unicode, Gray Code	2	
II	**Basic Computer Organization and Concept of Hardware**		**14**	**20**
	6	● Basic Computer Organization: Input Unit, Storage Unit, Processing Unit, Control Unit, Output Unit	1	
	7	● Concept of hardware	1	
	8	● CPU: Arithmetic and Logic Unit, Control unit	1	
	9	● Memory: Primary Memory, Secondary Memory, Access Time, Storage Capacity-bit, byte, nibble, volatile memory	2	
	10	● Memory hierarchy: Register memory, Cache memory, RAM (Static, Dynamic), ROM(Masked ROM, PROM and EPROM), Secondary storage devices (Magnetic tape, Hard disk, SSD and CD drive)	5	

		● Inside CPU: SMPS, Motherboard, Processor, Storage Devices (HDD, SSD, RAM, ROM).	1	
	11	● Motherboard Components: Processor Slot, Cooling Fan, RAM, Expansion Slots (PCIe), Input/Output Ports, Chipset, BIOS/UEFI Chip, SATA/NVMe Slots, Network Interface, Ports- Ethernet, VGA port, HDMI port, USB port.	3	
III	**Input-Output Devices, Concept of Software**		**12**	**15**
	12	● Input Devices: keyboard, pointing devices (mouse, touchpad), Video digitizer, remote control, joystick, scanner, digital camera, microphone, sensor	2	
	13	● Output Devices: monitor, printer (laser, inkjet, dot-matrix), plotter, speaker, control devices (lights, buzzers, robotic arms, and motors)	2	
	14	● Types of Software: System Software vs. Application Software, Proprietary Vs Open Source	2	
	15	Operating Systems: Functions, types of OS (batch, multiprogramming, time-sharing, real-time, and distributed)	2	
	16	Programming Languages (Machine, assembly & High level),	2	
	17	language Translators (Assembler, Interpreter and Compiler)	2	
1V	**Problem-solving and logical Thinking**		**11**	**20**
	18	Introduction to Problem Solving: Understanding the importance of problem-solving in computer science, Identifying and defining problems in a computational context.	2	
	19	Algorithm and its characteristics	1	
	20	Algorithm Development: Steps involved in designing algorithms, Pseudocode is an intermediate step in algorithm development.	2	
	21	Flowchart Basics: Introduction to flowcharts as a visual representation of algorithms, Understanding flowchart symbols and their meanings	2	
	22	Drawing simple flowcharts	4	
V	**Hands-on Data Structures:** **Practical Applications, Case Study and Course Project**		**30**	

	1	**Hardware:** 1. Identify the given motherboard components. 2. Identify and describe various ports and connectors on the motherboard.	5	
	2	**Software:** 1. Check the hardware compatibility and Install an operating system on a given computer. 2. Install any device driver on a given computer system to communicate with peripheral devices like Printers, Scanner, etc..	5	
	3	**Design Algorithm and visualize it using RAPTOR software** Problem 1: Calculate the Sum of Two Numbers Problem 2: Find the Larger of Two Numbers Problem 3: Check if a Number is Even or Odd Problem 4: Calculate the Factorial of a Number Problem 5: Temperature Conversion Problem 6: Simple Interest Calculation Problem 7: Calculate the Sum of Digits in a Number Problem 8: Check if a Number is Positive, Negative, or Zero Problem 9: Determine if a Triangle is Equilateral, Isosceles, or Scalene Problem 10: Check if a Number is Prime or Composite	20	

Reference Books:

1. Brookshear, J. Glenn. Computer Science: An Overview. 13th ed., Pearson, 2014.

2. Norton, Peter. Introduction to Computers. 7th ed., McGraw-Hill, 2016.

3. Patterson, David A. and John L. Hennessy. Computer Organization and Design: The Hardware/Software Interface. 5th ed., Morgan Kaufmann, 2013.

4. Sedgewick, Robert, and Kevin Wayne. Algorithms. 4th ed., Addison-Wesley Professional, 2011.

5. Knuth, Donald E. The Art of Computer Programming, Volumes 1-4A Boxed Set. Addison-Wesley Professional, 2011.

6. Grover, Aditya Bhargava. Grokking Algorithms: An Illustrated Guide for Programmers and Other Curious People. Manning Publications, 2016.

Mapping of COs with PSOs and POs :

	PSO1	PSO2	PSO3	PSO4	PSO5	PSO6	PO1	PO2	PO3	PO4	PO5	PO6
CO1	1	2	-	-	-	-	-					

CO 2	1	2	-	-	-	-	-					
CO 3	1	2	-	-	-	-	-					
CO 4	-	2	2	2	-	-	-					
CO 5	-	2	2	2	-	-	-					
CO 6	-	2	2	2	-	1	-					

Correlation Levels:

Level	Correlation
-	Nil
1	Slightly / Low
2	Moderate / Medium
3	Substantial / High

Assessment Rubrics:

- Quiz / Assignment/ Quiz/ Discussion / Seminar
- Midterm Exam
- Programming Assignments (20%)
- Final Exam (70%)

Mapping of COs to Assessment Rubrics:

	Internal Exam	Assignment	Project Evaluation	End Semester Examinations
CO 1		✓		✓
CO 2	✓	✓		✓
CO 3	✓	✓		✓
CO 4	✓	✓		✓
CO 5	✓	✓		✓

CO 6	✓	✓	✓	✓

Programme	B. Sc. Computer Science				
Course Code	CSC2MN101				
Course Title	Foundations of C Programming				
Type of Course	**Minor**				
Semester	II				
Academic Level	100-199				
Course Details	Credit	Lecture per week	Tutorial per week	Practical per week	Total Hours
	4	3	-	2	75
Pre-requisites	1. Basic Computer Literacy 2. Basic Problem-Solving Skills				
Course Summary	This course teaches the basics of programming using the C language. C is a powerful and widely used programming language known for its efficiency and flexibility. Through this course, students will learn how to write, understand, and debug C code to solve various problems and build simple applications.				

Course Outcomes (CO):

CO	CO Statement	Cognitive Level*	Knowledge Category#	Evaluation Tools used
CO1	Demonstrate a solid understanding of fundamental programming concepts	An	P	Instructor-created lab exams / Quiz
CO2	Develop effective problem-solving skills by applying algorithmic thinking and logical reasoning.	An	P	Problem-solving assessments
CO3	Gain proficiency in writing, compiling, debugging, and executing C programs to implement algorithms, solve	Ap	P	Modeling Assignments