Prescribed Books for Computer Science:
1. **Computer Science Theory XI – SUMITA ARORA**
2. **Computer Science XI (REETA SAHU) – SARASWATI - Reference**

Computer Science

(Revised)
CLASS-XI
Code No. 083
2020-21

1. Learning Outcomes

- Ability to understand and apply basic computational thinking.
- Ability to understand the notion of data types and data structures and apply in different situations.
- Ability to appreciate the notion of an algorithm and apply its structure including how algorithms handle corner cases.
- Ability to develop a basic understanding of computer systems - architecture, operating system, mobile and cloud computing.
- Ability to work in the cyber world with understanding of cyber ethics, cyber safety and cybercrime
- Ability to make use the value of technology in societies, gender and disability issues and the technology behind biometric ids.

2. Distribution of Marks

Unit No.	Unit Name	Theory Marks	Periods	
			Theory	**Practical**
I	Computer Systems and Organisation	10	10	5
II	Computational Thinking and Programming - 1	45	50	35
III	Society, Law and Ethics	15	20	----
	Total	70	80	40

Unit I: Computer Systems and Organisation

- Basic computer organisation: description of a computer system and mobile system, CPU, memory, hard disk, I/O, battery.
- Types of software: Application software, System software and Utility software.
- Memory Units: bit, byte, MB, GB, TB, and PB.
- Boolean logic: NOT, AND, OR, NAND, NOR, XOR, NOT, truth tables and De Morgan's laws, Logic circuits
- Number System: numbers in base 2, 8, 16 and binary addition.
- Encoding Schemes : ASCII, ISCII and Unicode

- Concept of Compiler and Interpreter
- Operating System (OS) - need for an operating system, brief introduction to functions of OS, user interface

Unit II: Computational Thinking and Programming - 1

Introduction to Problem solving: Problem solving cycle - Analysing a problem, designing algorithms and representation of algorithm using flowchart and pseudo-code.

Familiarization with the basics of Python programming: a simple "hello world" program, the process of writing a program (Interactive & Script mode), running it and print statements; simple data-types: integer, float and string.

- Features of Python, Python Character Set, Token & Identifiers, Keywords, Literals, Delimiters, Operators.
- Comments: (Single line & Multiline/ Continuation statements), Clarity & Simplification of expression
- Introduce the notion of a variable and methods to manipulate it (concept of L-value and R-value even if not taught explicitly).
- Knowledge of data types and operators: accepting input from the console, assignment statement, expressions, operators and their precedence.
- Operators & types: Binary operators-Arithmetic, Relational Operators, Logical Operators, Augmented Assignment Operators.
- Execution of a program, errors- syntax error, run-time error and logical error.
- Conditional statements: if, if-else, if-elif-else; simple programs: e.g.: absolute value, sort 3 numbers and divisibility of a number.
- Notion of iterative computation and control flow: for(range(),len()), while, using flowcharts, suggested programs: calculation of simple and compound interests, finding the factorial of a positive number etc.
- Strings: Traversal, operations – concatenation, repetition, membership; functions/methods–len(), capitalize(), title(), upper(), lower(), count(), find(), index(), isalnum(), islower(), isupper(), isspace(), isalpha(), isdigit(), split(), partition(), strip(), lstrip(), rstrip(), replace(); String slicing.
- Lists: Definition, Creation of a list, Traversal of a list. Operations on a list - concatenation, repetition, membership; functions/methods–len(), list(), append(), extend(), insert(), count(), index(), remove(), pop(), reverse(), sort(), min(), max(), sum(); Lists Slicing; Nested lists; finding the maximum, minimum, mean of numeric values stored in a list; linear search on list of numbers and counting the frequency of elements in a list.

- Tuples: Definition, Creation of a Tuple, Traversal of a tuple. Operations on a tuple - concatenation, repetition, membership; functions/methods – len(), tuple(), count(), index(), sorted(), min(), max(), sum(); Nested tuple; Tuple slicing; finding the minimum, maximum, mean of values stored in a tuple; linear search on a tuple of numbers, counting the frequency of elements in a tuple.
- Dictionary: Definition, Creation, Accessing elements of a dictionary, add an item, modify an item in a dictionary; Traversal, functions/methods – len(), dict(), keys(), values(), items(), get(), update(), del(), del, clear(), fromkeys(), copy(), pop(), popitem(), setdefault(), max(), min(), count(), sorted() copy(); Suggested programs : count the number of times a character appears in a given string using a dictionary, create a dictionary with names of employees, their salary and access them.
- Introduction to Python modules: Importing math module (pi, e, sqrt, ceil, floor, pow, fabs, sin, cos, tan); random module (random, randint, randrange), statistics module (mean, median, mode).

Unit III: Society, Law and Ethics

- Cyber safety: safely browsing the web, identity protection, confidentiality, social networks, cyber trolls and bullying.
- Appropriate usage of social networks: spread of rumours, and common social networking sites (Twitter, LinkedIn, and Facebook) and specific usage rules.
- Safely accessing web sites: adware, malware, viruses, trojans
- Safely communicating data: secure connections, eavesdropping, phishing and identity verification.
- Intellectual property rights, plagiarism, digital rights management, and licensing (Creative Commons, GPL and Apache), open source, open data, privacy.
- Privacy laws, fraud; cyber-crime- phishing, illegal downloads, child pornography, scams; cyber forensics, IT Act, 2000.
- Technology and society:
  - understanding of societal issues and cultural changes induced by technology.
  - E-waste management: proper disposal of used electronic gadgets.
  - Identity theft, unique ids and biometrics.
  - Gender and disability issues while teaching and using computers.

3. Practical

S.No.	Area	Marks (Total=30)
1.	**Lab Test (12 marks)**	
	Python program (60% logic + 20% documentation + 20% code quality)	**12**
2.	**Report File + Viva (10 marks)**	
	Report file: Minimum 20 Python programs	**7**
	Viva voce	**3**
3.	**Project (8 marks)** (that uses most of the concepts that have been learnt See CS-XII for the rules regarding the projects)	

4. Suggested Practical List

Python Programming

- Input a welcome message and display it.
- Input two numbers and display the larger / smaller number.
- Input three numbers and display the largest / smallest number.
- Given two integers x and n, compute x^n.
- Write a program to input the value of x and n and print the sum of the following series:

  - $1 + x + x^2 + x^3 + x^4 + \ldots\ldots x^n$
  - $1 - x + x^2 - x^3 + x^4 + \ldots\ldots x^n$
  - $x + \dfrac{x^2}{2} - \dfrac{x^3}{3} + \dfrac{x^4}{4} + \ldots\ldots \dfrac{x^n}{n}$
  - $x + \dfrac{x^2}{2!} - \dfrac{x^3}{3!} + \dfrac{x^4}{4!} + \ldots\ldots \dfrac{x^n}{n!}$

- Determine whether a number is a perfect number, an armstrong number or a palindrome.
- Input a number and check if the number is a prime or composite number.
- Display the terms of a Fibonacci series.
- Compute the greatest common divisor and least common multiple of two integers.
- Count and display the number of vowels, consonants, uppercase, lowercase characters in string.
- Input a string and determine whether it is a palindrome or not; convert the case of characters in a string.
- Find the largest/smallest number in a list/tuple
- Input a list of numbers and swap elements at the even location with the elements at the odd location.
- Input a list/tuple of elements, search for a given element in the list/tuple.
- Input a list of numbers and test if a number is equal to the sum of the cubes of its digits. Find the smallest and largest such number from the given list of numbers.
- Create a dictionary with the roll number, name and marks of n students in a class and display the names of students who have marks above 75.

COMPUTER SCIENCE - 083

CLASS XI

Topic reduced
Unit I: Computer Systems and Organisation ● Encoding Schemes : UTF8, UTF32 ● Concept of cloud computing and cloud services (SaaS,IaaS,PaaS), cloud (public/private), Blockchain technology
Unit II: Computational Thinking and Programming - 1 Decomposition – concept, need for decomposing a problem, examples of problem solving using decomposition. ● Sorting algorithm: bubble and insertion sort; count the number of operations while sorting.
Suggested Practical List Input a list of elements, sort in ascending/ descending order using Bubble/ Insertion sort.

CLASS XII

Unit I: Computational Thinking and Programming - 2 ● Recursion – simple algorithms with recursion : print a message forever, sum of first n natural numbers, factorial, Fibonacci numbers, recursion on arrays : binary search ● Idea of efficiency : performance measurement in terms of the number of operations. ● Data-structures: Lists as covered in Class XI, Stacks – Push, Pop using a list, Queues – Insert, Delete using a list. (One of the data structure Stack or Queue. Note : While setting the question paper a students will have an option between Stack and Queue.)
Unit II: Computer Networks ● Web Scripting Client side (VB Script, Java Script, PHP) and Server side (ASP, JSP, PHP), Web 2.0 (for social networking) ● E-commerce payment transactions using online banking, mobile banking, payment apps and services.
Unit III: Database Management CREATE TABLE, DROP TABLE, ALTER TABLE, UPDATE ....SET, INSERT, DELETE
1. Suggested Practical List: Python Programming ● Recursively find the factorial of a natural number ● Write a recursive code to find the sum of all elements of a list. ● Write a recursive code to compute the nth Fibonacci number