Computer Science
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	10
II	Computational Thinking and Programming - 1	45	80	60
III	Society, Law and Ethics	15	20	----
	Total	70	110	70

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, UTF8, UTF32, ISCII and Unicode.

- Concept of Compiler and Interpreter
- Operating System (OS) - need for an operating system, brief introduction to functions of OS, user interface
- Concept of cloud computing and cloud services (SaaS,IaaS,PaaS), cloud (public/private), Blockchain technology

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.

Decomposition - concept, need for decomposing a problem, examples of problem solving using decomposition.

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.

- Sorting algorithm: bubble and insertion sort; count the number of operations while sorting.

- 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:

  $\triangleright$ $1+x+x^2+x^3+x^4+\ldots\ldots\ldots x^n$

  $\triangleright$ $1-x+x^2-x^3+x^4-\ldots\ldots\ldots x^n$

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

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

- 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 of elements, sort in ascending/descending order using Bubble/Insertion sort.
- 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
CLASS-XII
Code No. 083
2020-21

1. Prerequisites

Computer Science- Class XI

2. Learning Outcomes

- Apply the concept of functions and recursion.
- Ability to create and use Python libraries.
- Apply the concept of file handling.
- Make use of the concept of efficiency in algorithms and computing in general.
- Ability to use basic data structures: Stacks and Queues.
- Explain the basics of computer networks.
- Ability to use connectivity between Python and SQL.

3. Distribution of Marks:

Unit No.	Unit Name	Theory Marks	Periods	
			Theory	Practical
I	Computational Thinking and Programming - 2	40	70	50
II	Computer Networks	10	15	---
III	Database Management	20	25	20
	Total	70	110	70

Unit I: Computational Thinking and Programming - 2

- Revision of the basics of Python covered in Class XI.
- Functions: scope, parameter passing, mutable/immutable properties of data objects, passing strings, lists, tuples, dictionaries to functions, default parameters, positional parameters, return values, functions using libraries: mathematical and string functions.
- File handling: Need for a data file, Types of file: Text files, Binary files and CSV (Comma separated values) files.

- Text File: Basic operations on a text file: Open (filename – absolute or relative path, mode) / Close a text file, Reading and Manipulation of data from a text file, Appending data into a text file, standard input / output and error streams, relative and absolute paths.
- Binary File: Basic operations on a binary file: Open (filename – absolute or relative path, mode) / Close a binary file, Pickle Module – methods load and dump; Read, Write/Create, Search, Append and Update operations in a binary file.
- CSV File: Import csv module, functions – Open / Close a csv file, Read from a csv file and Write into a csv file using csv.reader ( ) and csv.writerow( ).
- Using Python libraries: create and import Python libraries.
- 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.

Unit II: Computer Networks

- Evolution of Networking: ARPANET, Internet, Interspace Different ways of sending data across the network with reference to switching techniques (Circuit and Packet switching).
- Data Communication terminologies: Concept of Channel, Bandwidth (Hz, KHz, MHz) and Data transfer rate (bps, Kbps, Mbps, Gbps, Tbps).
- Transmission media: Twisted pair cable, coaxial cable, optical fiber, infrared, radio link, microwave link and satellite link.
- Network devices: Modem, RJ45 connector, Ethernet Card, Router, Switch, Gateway, WiFi card.
- Network Topologies and types: Bus, Star, Tree, PAN, LAN, WAN, MAN.
- Network Protocol: TCP/IP, File Transfer Protocol (FTP), PPP, HTTP, SMTP, POP3, Remote Login (Telnet) and Internet, Wireless / Mobile Communication protocol such as GSM, GPRS and WLL.
- Mobile Telecommunication Technologies: 1G, 2G, 3G, 4G and 5G; Mobile processors;
  Electronic mail protocols such as SMTP, POP3, Protocols for Chat and Video Conferencing: VoIP, Wireless technologies such as Wi-Fi and WiMax

- Network Security Concepts:
  Threats and prevention from Viruses, Worms, Trojan horse, Spams
  Use of Cookies, Protection using Firewall, https;
  India IT Act, Cyber Law, Cyber Crimes, IPR issues, hacking.
- Introduction To Web services: WWW, Hyper Text Markup Language (HTML), Extensible Markup Language (XML); Hyper Text Transfer Protocol (HTTP); Domain Names; URL; Website, Web browser, Web Servers; Web Hosting, 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

Database Concepts: Introduction to database concepts and its need.

Relational data model: Concept of domain, relation, tuple, attribute, degree, cardinality, key, primary key, candidate key, alternate key and foreign key;

Structured Query Language:

General Concepts: Advantages of using SQL, Data Definition Language and Data Manipulation Language;

Data Types: number / decimal, character / varchar / varchar2, date;

SQL commands: CREATE TABLE, DROP TABLE, ALTER TABLE, UPDATE ....SET...., INSERT, DELETE; SELECT, DISTINCT, FROM, WHERE, IN, BETWEEN, LIKE, NULL / IS NULL, ORDER BY,GROUP BY, HAVING;

SQL functions: SUM ( ), AVG ( ), COUNT ( ), MAX ( ) and MIN ( );

Joins: equi-join and natural join

Interface of Python with an SQL database
- Connecting SQL with Python
- Creating Database connectivity Applications
- Performing Insert, Update, Delete queries
- Display data by using fetchone(),fetchall(),rowcount

4. Practical

S. No.	Area	Marks (Total=30)
1	**Lab Test:** 1. Python program (60% logic + 20% documentation + 20% code quality) 2. Small Python program that sends a SQL query to a database and displays the result. A stub program can be provided.	7 5
2	**Report file:** Minimum 20 Python programs. Out of this at least 4 programs should send SQL commands to a database and retrieve the result	7
3	**Project** (that uses the concepts that have been learnt in Class 11 and 12)	8
4	**Viva voce**	3

5. 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.
- Read a text file line by line and display each word separated by a #.
- Read a text file and display the number of vowels/ consonants/ uppercase/ lowercase characters in the file.
- Create a binary file with name and roll number. Search for a given roll number and display the name, if not found display appropriate message.
- Create a binary file with roll number, name and marks. Input a roll number and update the marks.
- Remove all the lines that contain the character `a' in a file and write it to another file.
- Write a random number generator that generates random numbers between 1 and 6 (simulates a dice).
- Write a Python program to implement a stack and queue using a list data-structure.
- Take a sample of ten phishing e-mails (or any text file) and find most commonly occurring word(s)

Database Management

- Create a student table and insert data. Implement the following SQL commands on the student table:
  ALTER table to add new attributes / modify data type / drop attribute
  UPDATE table to modify data
  ORDER By to display data in ascending / descending order
  DELETE to remove tuple(s)
  GROUP BY and find the min, max, sum, count and average
- Similar exercise may be framed for other cases.
- Integrate SQL with Python by importing the MySQL module.

6. Project

The aim of the class project is to create something that is tangible and useful using Python / Python and SQL connectivity. This should be done in groups of two to three students and should be started by students at least 6 months before the submission deadline. The aim here is to find a real world problem that is worthwhile to solve.

Students are encouraged to visit local businesses and ask them about the problems that they are facing. For example, if a business is finding it hard to create invoices for filing GST claims, then students can do a project that takes the raw data (list of transactions), groups the transactions by category, accounts for the GST tax rates, and creates invoices in the appropriate format. Students can be extremely creative here. They can use a wide variety of Python libraries to create user friendly applications such as games, software for their school, software for their disabled fellow students, and mobile applications, Of course to do some of these projects, some additional learning is required; this should be encouraged. Students should know how to teach themselves.

The students should be sensitized to avoid plagiarism and violations of copyright issues while working on projects. Teachers should take necessary measures for this.