NAVY CHILDREN SCHOOL

Split Up Syllabus (2023-24)

Class –XI Computer Science (083)

1. **Pre-requisites**: Basic handling of computer system.

2. **Learning Outcomes**: Student should be able to
   - develop basic computational thinking
   - explain and use data types
   - appreciate the notion of algorithm
   - develop a basic understanding of computer systems - architecture, operating system and cloud computing
   - explain cyber ethics, cyber safety and cybercrime
   - Understand the value of technology in societies along with consideration of gender and disability issues

3. **Distribution of Marks:**

Unit No.	Unit Name	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

4. **Monthly Split up syllabus:**

Month	Chapter	Content/Practical/Assignment	Practical / Projects
June/ July	1. Computer Systems and Organisation	• Basic Computer Organisation: Introduction to computer system, hardware, software, input device, output device, CPU, memory (primary, cache and secondary), units of memory (bit, Byte, KB, MB, GB, TB,PB) • Types of software: system software (operating systems, system utilities, device drivers), programming tools and language translators (assembler, compiler & interpreter), application software • Operating system (OS): functions of operating system, OS user interface	Identifying various components of Computer
	2. Boolean Logic	• Boolean logic: NOT, AND, OR, NAND, NOR, XOR, truth table, De Morgan's laws and logic circuits	Making logical gates and proving theorems

	3. Number System 4. Encoding Schemes	• Number system: Binary, Octal, Decimal and Hexadecimal number system; conversion between number systems. • Encoding schemes: ASCII, ISCII and UNICODE (UTF8, UTF32)	Number System Conversion
August	5. Introduction to problem solving	• Steps for problem solving (analysing the problem, developing an algorithm, coding, testing and debugging). Representation of algorithms using flow chart and pseudo code, decomposition.	Writing Algorithms and preparing flowcharts for simple problems
	6. Getting Started with Python	• Familiarization with the basics of Python programming: Introduction to Python, features of Python, executing a simple "hello world" program, execution modes: interactive mode and script mode, Python character set, Python tokens (keyword, identifier, literal, operator, punctuator), variables, concept of l-value and r-value, use of comments.	Launching and working with python IDLE. Working in Interactive and script modes
	7. Python Fundamentals & Data Handling	• Knowledge of data types: number (integer, floating point, complex), boolean, sequence (string, list, tuple), none, mapping (dictionary), mutable and immutable data types • Operators: arithmetic operators, relational operators, logical operators, assignment operator, augmented assignment operators, identity operators(is, is not), membership operators(in, not in)	
	8. Python Expressions & Statements	• Expressions, statement, type conversion & input/output: precedence of operators, expression, evaluation of expression, python statement, type conversion (explicit & implicit conversion), accepting data as input from the console and displaying output	Use of operators, framing &evaluating expressions, type conversions, etc in Interactive mode
	9. Errors & Debugging	• Errors: syntax errors, logical errors, runtime errors	
	10. Flow of control: sequential & conditional flow, Loops	• Flow of control: introduction, use of indentation, sequential flow, conditional and iterative flow control • Conditional statements: if, if-else, if-elif-else, flowcharts, simple programs: e.g.: absolute value, sort 3 numbers and divisibility of a number • Iterative statements: for loop, range function, while loop, flowcharts, break and continue statements, nested loops, suggested programs: generating pattern, summation of series, finding the factorial of a positive number etc	Basic Programs, Programs that require decision making. Programs based on loops

Sept	11. Strings in Python	• Strings: introduction, indexing, string operations (concatenation, repetition, membership & slicing), traversing a string using loops, built-in functions: len(), capitalize(), title(), lower(), upper(), count(), find(), index(), endswith(), startswith(), isalnum(), isalpha(), isdigit(), islower(), isupper(), isspace(), lstrip(), rstrip(), strip(), replace(), join(), partition(), split()	Programs based on string manipulations
Oct/Nov	12.Lists 13.Tuples 14. Dictionary	• Lists: introduction, indexing, list operations (concatenation, repetition, membership & slicing), traversing a list using loops, built-in functions: len(), list(), append(), extend(), insert(), count(), index(), remove(), pop(), reverse(), sort(), sorted(), min(), max(), sum(); nested lists, suggested programs: 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: introduction, indexing, tuple operations(concatenation, repetition, membership & slicing), built-in functions: len(), tuple(), count(), index(), sorted(), min(), max(), sum(); tuple assignment, nested tuple, suggested programs: 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: introduction, accessing items in a dictionary using keys, mutability of dictionary (adding a new item, modifying an existing item), traversing a dictionary, built-in functions: len(), dict(), keys(), values(), items(), get(), update(), 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	Programs based on list operations Programs based on tuples Programs based on dictionaries

Dec/Jan	15. Introduction to Python Modules	• Introduction to Python modules: Importing module using 'import ' and using from statement, Importing math module (pi, e,sqrt, ceil, floor, pow, fabs, sin, cos, tan); random module (random, randint, randrange), statistics module (mean, median, mode)	Programs importing and using modules.
	16. Society, Laws and Ethics	• Digital Footprints • Digital society and Netizen: net etiquettes, communication etiquettes, social media etiquettes. • Data protection: Intellectual Property Right (copyright, patent, trademark), violation of IPR (plagiarism, copyright infringement, trademark infringement), open source softwares and licensing (Creative Commons, GPL and Apache) • Cyber-crime: definition, hacking, eavesdropping, phishing and fraud emails, ransomware, preventing cyber crime • Cyber safety: safely browsing the web, identity protection, confidentiality, cyber trolls and bullying. • Safely accessing web sites: malware, viruses, Trojans, adware • E-waste management: proper disposal of used electronic gadgets • Indian Information Technology Act (IT Act) • Technology & Society: Gender and disability issues while teaching and using computers.	Understandi ng of Cyber laws and online ethics including safety measures to protect data and information available online
Jan/Feb		Revision for NES Common Final Exam	

1. **Blue Print:** To be followed strictly in accordance with the CBSE SQP for class XII to be released by CBSE on its website in due course of time.

2. **Practical Work:** As per the CBSE list of suggested Practical for the Academic Year 2023-24.

3. **Sample QP:** Annual QP to be set strictly in accordance with the CBSE SQP to be released by CBSE on its website in due course of time.