Academic Council AC/27.06.23/RS1
Item No: _____

SIES College of Arts, Science and Commerce, Sion (W)
Autonomous College

Syllabus for
Program: First Year Bachelor of Science (NEP) Course: Computer Science
Semester: I & II

With effect from
Academic Year 2023 -24

Preamble

Information and Communication Technology (ICT) has today become an integral part of all industry domains as well as fields of academics and research. The industry requirements and technologies have been steadily and rapidly advancing. Organizations are increasingly opting for open source systems. The students too these days are thinking beyond careers in the industry and aiming for research opportunities.

The B.Sc. Computer Science course structure therefore needed a fresh outlook and complete overhaul. A real genuine attempt has been made while designing the new syllabus for this 3 year graduate course. Not only does it prepare the students for a career in the Software industry, it also motivates them towards further studies and research opportunities.

The core philosophy of overall syllabus is to -

a. Form strong foundation of Computer science,
b. Introduce emerging trends to the students in gradual way,
c. Groom the students for the challenges of ICT industry

In the first year i.e. for semester I & II, the basic foundation of important skills required for software development is laid. The syllabus proposes to have 2 core subjects of Computer science and 2 core courses of Mathematics-Statistics. In Semester II the students would also be given industrial exposure via field projects/industrial visit. All core subjects are proposed to have theory as well as practical tracks. While the Computer Science courses will form fundamental skills for solving computational problems, the Mathematics & Statistics course will inculcate research oriented acumen.

The syllabus design for further semesters encompasses more advanced and specialized courses of Computer Science.

We sincerely believe that any student taking this course will get a very strong foundation and exposure to basics, advanced and emerging trends of the subject. We hope that the students' community and teachers' fraternity will appreciate the treatment given to the courses in the syllabus.

We wholeheartedly thank all experts who shared their valuable feedback and suggestions in order to improvise the contents, we have sincerely attempted to incorporate each of them. We further thank the Chairperson and members of the Board of Studies for their confidence in us. Special thanks to the Department of Computer Science and colleagues from various colleges, who volunteered or have indirectly helped design certain specialized courses and the syllabus as a whole.

Program Outcomes and Program Specific Outcomes

B.Sc. Computer Science

SR.NO	Details
PO 1	Recall and explain acquired scientific knowledge in a comprehensive manner and apply the skills acquired in their chosen discipline. Interpret scientific ideas and relate its interconnectedness to various fields in science.
PO 2	Evaluate scientific ideas critically, analyse problems, explore options for practical demonstrations, illustrate work plans and execute them, organize data and draw inferences.
PO 3	Explore and evaluate digital information and use it for knowledge upgradation. Apply relevant information so gathered for analysis and communication using appropriate digital tools.
PO 4	Ask relevant questions, understand scientific relevance, hypothesize a scientific problem, construct and execute a project plan and analyze results.
PO 5	Take complex challenges; work responsibly and independently, as well as in cohesion with a team for completion of a task. Communicate effectively, convincingly and in an articulate manner.
PO 6	Apply scientific information with sensitivity to values of different cultural groups. Disseminate scientific knowledge effectively for upliftment of the society.
PO 7	Follow ethical practices at the workplace and be unbiased and critical in interpretation of scientific data. Understand the environmental issues and explore sustainable solutions for it.
PO 8	Keep abreast with current scientific developments in the specific discipline and adapt to technological advancements for better application of scientific knowledge as a lifelong learner.

SR.NO	Details
PSO 1	Apply knowledge of computational mathematics, statistics and programming acquired in the field of Computer Science.
PSO 2	Identify, analyze complex problems in the real world and formulate innovative solutions to those problems.
PSO 3	Compare and apply hardware and software technologies for implementing reliable optimized solutions catering to need and available resources.
PSO 4	Apply software development, managerial, Professional, and soft skills in industry
PSO 5	Understand the global needs and prepare themselves for the changing needs worldwide adapting an ability to engage in life-long learning.
PSO 6	Become a responsible, ethical citizen and explore environmental issues to develop sustainable solutions for it.

F.Y.B.Sc. Computer Science CORE COURSE Syllabus
Credit Based System and Grading System
Academic year 2023-2024

Semester – I						
Course Code	**Course Type**	**Course Title**	**Credits**	**Lectures/Week**		
				Theory	**Practical (2 lectures)**	**Total**
SIUCSMJ111	Major	Programming with Python	3	3		3
SIUCSMJP111	Major Practical	Practical of SIUCSMJ111	1		1	1
SIUCSMN111	Minor Subject	Fundamentals of Mathematics and Statistics-I	3	3		3
SIUCSMNP111	Minor Practical	Practical of SIUCSMN111	1		1	1

Semester – II						
Course Code	**Course Type**	**Course Title**	**Credits**	**Lectures/Week**		
				Theory	**Practical (2 lectures)**	**Total**
SIUCSMJ121	Major Subject	Data Structures and fundamentals of Algorithm	3	3		3
SIUCSMJP121	Major Practical	Practical of SIUCSMJ121	1		1	1
SIUCSMN121	Minor Subject	Fundamentals of Mathematics and Statistics-II	3	3		3
SIUCSMNP121	Minor Practical	Practical of SIUCSMN121	1		1	1

Semester I – Theory

Course	Title	Lectures	Credits
SIUCSMJ111	**Programming with Python- I**	**3 per week (60 min per lec)**	**3**

Objectives

The objective of this paper is to introduce various concepts of programming to the students using Python.

Course Outcomes:

- **CO1**: Students should be able to understand the concepts of programming before actually starting to write programs.
- **CO2**: Students should be able to develop logic for Problem Solving.
- **CO3**: Students should be made familiar with the basic constructs of programming such as data, operations, conditions, loops, functions etc.
- **CO4**: Students should be able to apply the problem solving skills using syntactically simple language i.e. **Python (version: 3.X or higher)**

Unit I	**Introduction:** The Python Programming Language, History, features, Installing Python, Running Python program, Debugging : Syntax Errors, Runtime Errors, Semantic Errors, Experimental Debugging, Formal and Natural Languages **Variables and Expressions**: Values and Types, Variables, Variable Names and Keywords, Type conversion, Operators and Operands, Expressions, Interactive Mode and Script Mode, Order of Operations. Conditional Statements: if, if-else, nested if –else Looping: for, while, nested loops Control statements: Terminating loops, skipping specific conditions. **Compound Data types:** Strings, Lists, Tuples, Dictionaries	15 L
Unit II	**Functions And Modules**: Defining a function, calling a function, Advantages of functions, types of functions, function parameters, Formal parameters, Actual parameters, global and local variables, Anonymous functions, List comprehension Importing module, Creating & exploring modules. Anonymous functions. List comprehensions. Gentle introduction to object-oriented programming; using the built-in dir() function, enumerate the methods of strings, tuples, lists, dictionaries. Using these methods for problem-solving with compound types. **Python File Input-Output:** Opening and closing files, various types of file modes, reading and writing to files, manipulating directories. Iterables, iterators and their problem solving applications.	15 L

Unit III	**Exception handling:** What is an exception, various keywords to handle exceptions such try, catch, except, else, finally, raise. **Regular Expressions:** Concept of regular expression, various types of regular expressions, using match function. **Database connectivity in Python**: Installing MySQL connector, accessing connector module, using connect, cursor, execute & close functions, reading single & multiple results of query execution, executing different types of statements, executing transactions, understanding exceptions in database connectivity.	15 L

Text books:

1. Magnus Lie Hetland, Beginning Python: From Novice to Professional, Apress
2. Paul Gries, et al., Practical Programming: An Introduction to Computer Science Using Python 3, Pragmatic Bookshelf, 2/E 2014

Additional References:

1. Charles Dierbach, *Introduction to Computer Science using Python*, Wiley, 2013
2. Paul Gries , Jennifer Campbell, Jason Montojo, *Practical Programming: An Introduction to Computer Science Using Python 3*, Pragmatic Bookshelf, 2/E 2014
3. Adesh Pandey, *Programming Languages – Principles and Paradigms,* Narosa, 2008

Course	Title	Lectures	Credits
SIUCSMJP111	**Practicals on Programming with Python**	**2 per week (60 min per lec)**	**1**
1	Installing and setting up the Python IDLE interpreter. Executing simple statements like expression statements (numeric and Boolean types), assert, assignment, delete statements; the print function for output, the input function.		
2	Programs based on conditional constructs(if, if else, if elif else, nested if)		
3	Programs based on for statement and the range function, using break and continue statements		
4	Programs based on the while statement		
5	Programs related to string manipulation		
5	Programs related to lists and list comprehensions		
6	Programs related to dictionaries		
7	Programs related to functions		
8	Programs to read and write files.		
9	Program to demonstrate exception handling.		
10	Program to demonstrate the use of regular expressions.		
11	Programs related to database handling		

Course	Title	Lectures	Credits
SIUCSMN111	**Fundamentals of Mathematics and Statistics- I**	**3 per week (60 min per lec)**	**3**

Objectives:

The purpose of the course is to familiarize the prospective learners with mathematical structures that are fundamentally discrete. This course introduces sets and functions, forming and solving recurrence relations and different counting principles. These concepts are useful to study or describe objects or problems in computer algorithms and programming languages.

Course Outcomes:

- **CO1:** To provide an overview of the theory of discrete objects, starting with relations and partially ordered sets.
- **CO2:** Study about recurrence relations, generating function and operations on them.
- **CO3:** Give an understanding of graphs and trees, which are widely used in software.
- **CO4**: Provide basic knowledge about models of automata theory and the corresponding formal languages.
- **CO5**: Enable learners to know descriptive statistical concepts
- **CO6:** Enable study of probability concept required for Computer learners

Unit I	**Recurrence Relations** (a) **Functions:** Definition of function. Domain, co domain and the range of a function. Direct and inverse images. Injective, surjective and bijective functions. Composite and inverse functions. (b) **Relations**: Definition and examples. Properties of relations , Partial Ordering sets, Linear Ordering Hasse Diagrams , Maximum and Minimum elements, Lattices (c) **Recurrence Relations:** Definition of recurrence relations, Formulating recurrence relations, solving recurrence relations- Backtracking method, Linear homogeneous recurrence relations with constant coefficients. Solving linear homogeneous recurrence relations with constant coefficients of degree two when characteristic equation has distinct roots and only one root, Particular solutions of non linear homogeneous recurrence relation, Solution of recurrence relation by the method of generation functions, Applications- Formulate and solve recurrence relation for Fibonacci numbers, Tower of Hanoi, Intersection of lines in a plane, Sorting Algorithms.	15L
	Counting Principles , Languages and Finite State Machine (a) **Permutations and Combinations and counting Principles:** Partition and Distribution of objects, Permutation with distinct and indistinct objects, Binomial numbers, Combination with identities: Pascal Identity, Vandermonde's Identity, Pascal triangle, Binomial theorem, Combination with indistinct objects.	

	(b) **Counting Principles:** Sum and Product Rules, Two-way counting, Tree diagram for solving counting problems, Pigeonhole Principle (without proof); Simple examples, Inclusion Exclusion Principle (Sieve formula) (Without proof). (c) **Languages, Grammars and Machines:** Languages , regular Expression and Regular languages, Finite state Automata, grammars, Finite state machines, Gödel numbers, Turing machines.	
Unit II	**Graphs and Trees** (a) **Graphs :** Definition and elementary results, Adjacency matrix, path matrix, Representing relations using digraphs, Warshall's algorithm-shortest path , Linked representation of a graph, Operations on graph with algorithms - searching in a graph; Insertion in a graph, Deleting from a graph, Traversing a graph- Breadth-First search and Depth-First search. (b) **Trees:** Definition and elementary results. Ordered rooted tree, Binary trees, Complete and extended binary trees, representing binary trees in memory, traversing binary trees, binary search tree, Algorithms for searching and inserting in binary search trees, Algorithms for deleting in a binary search tree.	15L
Unit III	(a) **Data Presentation and Data Aggregation** ● **Data types :** attribute, variable, discrete and continuous variable ● **Data presentation :** frequency distribution, histogram to give, curves, stem and leaf display ● **Measures of Central tendency:** Mean, Median, mode for raw data, discrete, grouped frequency distribution. ● **Measures dispersion:** Variance, standard deviation, coefficient of variation for raw data, discrete and grouped frequency distribution, quartiles, quantiles Real life examples (b) **Moments, Skewness and Kurtosis** ● **Moments:** raw moments, central moments, relation between raw and central moments ● **Measures of Skewness and Kurtosis:** based on moments, quartiles, relation between mean, median, mode for symmetric, asymmetric frequency curve. (c) **Correlation and Regression:** bivariate data, scatter plot, correlation, nonsense correlation, Karl pearson's coefficients of correlation, independence. ● **Linear regression:** fitting of linear regression using least square regression, coefficient of determination, properties of regression coefficients (only statement)	15L

Text books:

1. Trivedi, K.S.(2001) : Probability, Statistics, Design of Experiments and Queuing theory, with applications of Computer Science, Prentice Hall of India, New Delhi

Additional References:

1. Ross, S.M. (2006): A First course in probability. 6th Edn Pearson
2. Kulkarni, M.B., Ghatpande, S.B. and Gore, S.D. (1999): common statistical tests. Satyajeet Prakashan, Pune
3. Gupta, S.C. and Kapoor, V.K. (1987): Fundamentals of Mathematical Statistics, S. Chand and Sons, New Delhi
4. Gupta, S.C. and Kapoor, V.K. (1999): Applied Statistics, S. Chand and Son's, New Delhi
5. Montgomery, D.C. (2001): Planning and Analysis of Experiments, wiley.

Course	Title	Lectures	Credits
SIUCSMNP111	**Practicals of Fundamentals of Mathematics and Statistics- I**	**2 per week (60 min per lec)**	**1**
1	Write a Program on to implement a.Hasse Diagram b.Poset c.Lattice d.Tower of Hanoi		
2	Write a Program on to implement A. Permutation B.Combination C.Counting Principle		
3	Write a Program on to implement A. Breadth First Search B. Depth First Search		
4	Write a program on Binary Trees to perform A. Insertion B. Traversal		
5	Write a program to implement A. Basics functions B. Frequency Distribution and Data Presentation		
6	Write a program to implement Measure of Central Tendency A. Mean B. Median C. Mode		
7	Measure of Dispersion A. Standard Deviation B. Coefficient of Variance C.Quartiles		
8	Moments, Skewness and Kurtosis A. Central and Raw Moments B.Skewness C.Kurtosis		
9	Correlation and Regression A. Karl Pearson's coefficient of correlation B. Linear Regression		

****

Semester II - Theory

Course	Title	Lectures	Credits
SIUCSMJ121	**Data Structures and fundamentals of Algorithm**	**3 per week (60 min per lec)**	**3**

Objectives:
To explore and understand the concepts of Data Structures and its significance in programming. Provide a holistic approach to design, use and implement abstract data types. Understand the commonly used data structures and various forms of its implementation for different applications using Python.

Expected Learning Outcomes:

- **CO1**: Learn about Data structures, its types and significance in computing
- **CO2**: Explore about Abstract Data types and its implementation
- **CO3**: Ability to program various applications using different data structure in Python

Unit I	**Abstract Data Types:** Introduction, Bags, Iterators. Application **Arrays:** Array Structure, Python List, Two Dimensional Arrays, Matrix Abstract Data Type, Application **Sets and Maps:** Sets-Set ADT, Selecting Data Structure, List based implementation **Maps**: Map ADT, List Based Implementation, Application **Searching and Sorting:** Searching-Linear Search, Binary Search, Sorting-Bubble, Selection and Insertion Sort, Merge Sort, Quick Sort, Radix Sort	15L
Unit II	**Linked Structures**: Introduction, Singly Linked List-Traversing, Searching and Removing Nodes **Stacks**: Stack ADT, Implementing Stacks-Using Python List, Using Linked List, Stack Applications-Infix to postfix conversion, Evaluating Postfix Expressions **Queues**: Queue ADT, Implementing Queue-Using Python List, Circular Array, Using List, Priority Queues- Priority Queue ADT, Bounded and unbounded Priority Queues **Advanced Linked List**: Doubly Linked Lists-Organization and Operation, Circular Linked List-Organization and Operation	15L

Unit III	**Recursion:** Recursive Functions, Properties of Recursion, Its working, Recursive Applications **Hash Table:** Introduction, Hashing-Linear Probing, Clustering, Rehashing, Separate Chaining, Hash Functions **Binary Trees:** Tree Structure, Binary Tree-Properties, Implementation and Traversals, Heaps and Heapsort,Search Trees **Graphs**: Definition, Implementation in Python, Traversal algorithms (DFS and BFS) **Algorithm Analysis:** Introduction to algorithm, Why to analysis algorithm, Running time analysis, How to Compare Algorithms, Rate of Growth, Commonly Used Rates of Growth, Types of Analysis, Asymptotic Notation, Big-O Notation, Omega-Ω Notation, Theta-Θ Notation, Asymptotic Analysis, Properties of Notations, Commonly used Logarithms and Summations, Performance characteristics of algorithms, Master Theorem for Divide and Conquer,Master Theorem for Subtract and Conquer , Evaluating Python Code	15L

Text book:

1) *Data Structure and algorithm Using Python*, Rance D. Necaise, 2016 Wiley India Edition

2) *Data Structure and Algorithm in Python*, Michael T. Goodrich, Robertom Tamassia, M. H. Goldwasser, 2016 Wiley India Edition

Additional References:

1) *Data Structure and Algorithmic Thinking with Python*- Narasimha Karumanchi, 2015, Careermonk Publications

2) *Fundamentals of Python: Data Structures*, Kenneth Lambert, Delmar Cengage Learning

Course	Title	Lectures	Credits
SIUCSMJP121	**Practicals of Data Structures and fundamentals of Algorithm**	**2 per week (60 min per lec)**	**1**
1	Implement Linear Search to find an item in a list.		
2	Implement binary search to find an item in an ordered list.		
3	Implement Sorting Algorithms A. Bubble sort B. Insertion sort		
4	Implement use of Sets and various operations on Sets.		
5	Implement working of Stacks. (pop method to take the last item added off the stack and a push method to add an item to the stack)		
6	Implement Program for A. Infix to Postfix conversion B. Postfix Evaluation		
7	Implement the following A. A queue as a list which you add and delete items from. B. A circular queue. (The beginning items of the queue can be reused).		
8	Implement Linked list and demonstrate the functionality to add and delete items in the linked list.		
9	Implement Binary Tree and its traversals.		
10	Recursive implementation of A. Factorial B. Fibonacci		
11	Write a Python program to sort n names using Quick sort algorithm. Discuss the complexity of algorithms used.		
12	Write a Python program to sort n numbers using Merge sort algorithm. Discuss the complexity of algorithms used.		
13	Write a Python program for inserting an element into a binary tree.		
14	Write a Python program for deleting an element (assuming data is given) from a binary tree.		

Course	Title	Lectures	Credits
SIUCSMN121	**Fundamentals of Mathematics and Statistics-II**	**3 per week (60 min per lec)**	**3**

Objective:
The objective of this paper is to explore the style of structured programming to give the idea to the students how programming can be used for designing real-life applications by reading/writing to files, GUI programming, interfacing database/networks and various other features.

Expected Learning Outcomes

- **CO1**: Students should be able to understand how to read/write to files using python.
- **CO2**: Students should be able to catch their own errors that happen during execution of programs.
- **CO3**: Students should get an introduction to the concept of pattern matching.
- **CO4**: Students should be made familiar with the concepts of GUI controls and designing GUI applications.
- **CO5**: Students should be able to connect to the database to move the data to/from the application.
- **CO6**: Students should know how to connect to computers, read from URL and send email.

Unit I	**Derivatives And Its Applications:** Review of Functions, limit of a function, continuity of a function, derivative function. Derivative In Graphing And Applications: Analysis of Functions: Increase, Decrease, Concavity, Relative Extrema; Graphing Polynomials, Rational Functions, Cusps and Vertical Tangents. Absolute Maxima and Minima, Applied Maximum and Minimum Problems, Newton's Method.	15 L
Unit II	**Integration And Its Applications:** An Overview of the Area Problem, Indefinite Integral, Definition of Area as a Limit; Sigma Notation, Definite Integral, Evaluating Definite Integrals by Substitution, Area Between Two Curves, Length of a Plane Curve. Numerical Integration: Simpson's Rule. Modeling with Differential Equations, Separation of Variables, Slope Fields, Euler's Method, FirstOrder Differential Equations and Applications. **Partial Derivatives And Its Applications:** Functions of Two or More Variables Limits and Continuity Partial Derivatives, Differentiability, Differentials, and Local Linearity, Chain Rule, Directional Derivatives and Gradients, Tangent Planes and Normal, Vectors, Maxima and Minima of Functions of Two Variables.	15 L

Unit III	**Standard distributions:** random variable; discrete, continuous, expectation and variance of a random variable, pmf, pdf, cdf, reliability, Introduction and properties without proof for following distributions; binomial, normal, chi-square, t, F. Examples **Hypothesis testing:** one sided, two sided hypothesis, critical region, p-value, tests based on t, Normal and F, confidence intervals. Analysis of variance : one-way, two-way analysis of variance **Non-parametric tests:** need of non-parametric tests, sign test, Wilicoxon's signed rank test, run test, Kruskal-Walis tests. Post-hoc analysis of one-way analysis of variance : Duncan's test Chi-square test of association	15 L

Text Book:

1. Trivedi, K.S.(2009) : Probability, Statistics, Design of Experiments and Queuing theory, with applications of Computer Science, Prentice Hall of India, New Delhi

Additional References:

1. Ross, S.M. (2006): A First course in probability. 6th Edn Pearson
2. Kulkarni, M.B., Ghatpande, S.B. and Gore, S.D. (1999): Common statistical tests. Satyajeet Prakashan, Pune
3. Gupta, S.C. and Kapoor, V.K. (2002) : Fundamentals of Mathematical Statistics, S. Chand and Sons, New Delhi
4. Gupta, S.C. and Kapoor, V.K. (4th Edition) : Applied Statistics, S. Chand and Son's, New Delhi
5. Montgomery, D.C. (2001): Planning and Analysis of Experiments, Wiley.

Course	Title	Lectures	Credits
SIUCSMNP121	**Practicals of Fundamentals of Mathematics and Statistics- II**	**2 per week (60 min per lec)**	**1**
1	Increasing, decreasing, concave up and concave down functions		
2	Relative maxima, relative minima, absolute maxima, absolute minima		
3	Newton's method to find approximate solution of an equation		
4	Numerical integration using Simpson's rule		
5	Solution of a differential equation, Euler's method, Runge Kutta Method.		
6	Calculation of Partial derivatives of functions		
7	Maxima and minima of functions of two variables		
8	Problems based on binomial distribution		
9	Problems based on normal distribution		
10	Parametric test		
11	Non parametric tests- I		
12	Non- Parametric tests – II		

****