

UNIVERSITY OF MUMBAI



Syllabus For the Program: F.Y.B.Sc. Sem -I CBCS Course: Computer Science

**(Choice Based and Credit System with effect from the
academic year 2021-22)**

Preamble

The rise of Information and Communication Technology (ICT) has profoundly affected modern society. Increasing applications of computers in almost all areas of human endeavor has led to vibrant industries with concurrent rapid change in technology.

As the computing field advances at a rapid pace, the students must possess a solid foundation that allows and encourages them to maintain relevant skills as the field evolves. Specific languages and technology platforms change over time. Thus students must continue to learn and adapt their skills throughout their careers. To develop this ability, students will be exposed to multiple programming languages, tools, paradigms and technologies as well as the fundamental underlying principles throughout this programme.

The programme offers required courses such as programming languages, data structures, computer architecture and organization, algorithms, database systems, operating systems, and software engineering; as well as specialized courses in artificial intelligence, computer-based communication networks, distributed computing, information security, graphics, human-computer interaction, multimedia, scientific computing, web technology, and other current topics in computer science.

The core philosophy of this programme is to –

- ☐ Form strong foundations of Computer Science
- ☐ Nurture programming, analytical & design skills for the real world problems.
- ☐ Introduce emerging trends to the students in gradual way.
- ☐ Groom the students for the challenges of ICT industry

The students these days not only aspire for a career in the industry but also look for research opportunities. The main aim of this programme is to deliver a modern curriculum that will equip graduates with strong theoretical and practical backgrounds to enable them to excel in the workplace and to be lifelong learners. Not only does it prepare the students for a career in Software industry, it also motivates them towards further studies and research opportunities. Graduating students, can thus take up postgraduate programmes in CS leading to research as well as R&D, can be employable at IT industries, or can adopt a business management career.

In the first year i.e. for semester I & II, basic foundation of important skills required for software development is laid. The syllabus proposes to have four core subjects of Computer science and two core courses of Mathematics-Statistics. 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. Ability Enhancement Courses on Soft Skill Development will ensure an overall and holistic development of the students. The syllabus design for further semesters encompasses more advanced and specialized courses of Computer Science.

We sincerely believe that any student taking this programme will get 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 feedbacks and suggestions in order to improvise the contents; we have sincerely attempted to incorporate each of them. We further thank Chairperson and members of Board of Studies for their confidence in us.

Special thanks to Department of Computer Science and colleagues from various colleges, who volunteered

or have indirectly, helped designing certain specialized courses and the syllabus as a whole.

Academic year 2021-2022

Semester – I				
Course Code	Course Type	Course Title	Credits	Lectures/Week
USCS101	Core Subject	Digital Systems & Architecture	2	3
USCSP101	Core Subject Practical	Digital Systems & Architecture – Practical	1	3
USCS102	Core Subject	Introduction to Programming with Python	2	3
USCSP102	Core Subject Practical	Introduction to Programming with Python – Practical	1	3
USCS103	Core Subject	LINUX Operating System	2	3
USCSP103	Core Subject Practical	LINUX Operating System – Practical	1	3
USCS104	Core Subject	Open Source Technologies	2	3
USCSP104	Core Subject Practical	Open Source Technologies – Practical	1	3
USCS105	Core Subject	Discrete Mathematics	2	3
USCSP105	Core Subject Practical	Discrete Mathematics – Practical	1	3
USCS106	Core Subject	Descriptive Statistics	2	3
USCSP106	Core Subject Practical	Descriptive Statistics – Practical	1	3
USCS107	Ability Enhancement Course	Soft Skills	2	3

Semester I

Course Code	Course Title	Credits	Lectures /Week
USCS102	Introduction to Programming with Python	2	3
About the Course: This course is aims at introducing one of the fastest growing programming language of current time and enables learners to understand the fundamentals of programming with Python. Learners will be able to write programs to solve real-world problems, and produce quality code. It will help to develop strong skills of programming for implementing applications for emerging fields including data science and machine learning.			
Course Objectives: <ul style="list-style-type: none"> <input type="checkbox"/> To learn how to design and program Python applications. <input type="checkbox"/> To explore the innards of Python Programming and understand components of Python Program <input type="checkbox"/> To define the structure and components of a Python program. <input type="checkbox"/> To learn how to write loops and decision statements in Python <input type="checkbox"/> To learn about inbuilt input/output operations and compound data types in Python 			
Learning Outcomes: After successful completion of this course, students would be able to: <ul style="list-style-type: none"> <input type="checkbox"/> Ability to store, manipulate and access data in Python <input type="checkbox"/> Ability to implement basic Input / Output operations in Python <input type="checkbox"/> Ability to define the structure and components of a Python program. <input type="checkbox"/> Ability to learn how to write loops and decision statements in Python. <input type="checkbox"/> Ability to learn how to write functions and pass arguments in Python. <input type="checkbox"/> Ability to create and use Compound data types in Python 			
Unit	Topics	No of Lectures	
I	Overview of Python: History & Versions, Features of Python, Execution of a Python Program, Flavours of Python, Innards of Python, Python Interpreter, Memory Management in Python, Garbage Collection in Python, Comparison of Python with C and Java, Installing Python, Writing and Executing First Python Program, Getting Help, IDLE Data Types, Variables and Other Basic Elements: Comments, Docstrings, Data types- Numeric Data type, Compound Data Type, Boolean Data type, Dictionary, Sets, Mapping, Basic Elements of Python, Variables Input and Output Operations: Input Function, Output Statements, The print() function, The print("string") function, The print(variables list) function, , The print(object) function, The print(formatted string) function, Command Line Arguments Control Statements: The if statement, The if ... else Statement, The „if ...	15	

	elif ... else' Statement, Loop Statement- while loop, for loop, Infinite loop, Nested loop, The else suite, break statement, continue statement, pass statement, assert statement, return statement	
II	<p>Operators: Arithmetic operators, Assignment operators, Unary minus operator, Relational operators, Logical operators, Bitwise operators, Membership operators, Identity operators, Precedence of Operators, Associativity of Operators</p> <p>Arrays: Creating Arrays, Indexing and Slicing of Arrays, Basic Array Operations, Arrays Processing, Mathematical Operations on Array, Aliasing Arrays, Slicing and Indexing in NumPy Arrays, Basic slicing, Advanced Indexing, Dimensions of Arrays, Attributes of an Array, The ndim Attribute, The shape Attribute, The size Attribute, The itemsize Attribute</p> <p>Functions: Function definition and call, Returning Results, Returning Multiple Values from a Function, Built-in Functions, Difference between a Function and a Method, Pass Value by Object Reference, Parameters and Arguments, Formal and Actual Arguments, Positional Arguments, Keyword Arguments, Default Arguments, Arbitrary Arguments, Recursive Functions, Anonymous or Lambda Functions, Using Lambda with the filter() Function, Using Lambda with the map() Function, Using Lambda with the reduce() Function</p> <p>Modules: Introduction to Modules in Python</p>	15
III	<p>Strings: Creating Strings, Functions of Strings, Working with Strings, Length of a String, Indexing and Slicing, Repeating and Concatenating Strings, Checking Membership, Comparing Strings, Removing Spaces, Finding Substrings, Counting Substrings, Immutability, Splitting and Joining Strings, Changing Case, Checking Starting and Ending of a String, Sorting Strings, Searching in the Strings, Testing Methods, Formatting Strings, Finding the Number of Characters and Words, Inserting Substrings into a String</p> <p>List and Tuples: Lists, List Functions and Methods, List Operations, List Slices, Nested Lists, Tuples, Functions in Tuple</p> <p>Dictionaries: Creating a Dictionary, Operators in Dictionary, Dictionary Methods, Using for Loop with Dictionaries, Operations on Dictionaries, Converting Lists into Dictionary, Converting Strings into Dictionary, Passing Dictionaries to Functions, Sorting the Elements of a Dictionary using Lambda, Ordered Dictionaries</p>	15
<p>Textbooks:</p> <ol style="list-style-type: none"> 1. Practical Programming: An Introduction to Computer Science Using Python 3, Paul Gries , Jennifer Campbell, Jason Montojo, Pragmatic Bookshelf, 2nd Edition, 2014 2. Programming through Python, M. T Savaliya, R. K. Maurya & G M Magar, Sybgen Learning India, 2020 		

Additional References:

1. Python: The Complete Reference, Martin C. Brown, McGraw Hill, 2018
2. Beginning Python: From Novice to Professional, Magnus Lie Hetland, Apress, 2017
3. Programming in Python 3, Mark Summerfield, Pearson Education, 2nd Ed, 2018
4. Python Programming: Using Problem Solving Approach, ReemaThareja, Oxford Univeristy Press, 2017
5. Let Us Python, Yashwant. B. Kanetkar, BPB Publication, 2019

Course Code	Course Title	Credits	Lectures /Week
USCSP102	Introduction to Programming with Python – Practical	1	3
1	Write a program to design and develop python program to implement various control statement using suitable examples		
2	Write program in Python to define and call functions for suitable problem.		
3	Write Python program to demonstrate different types of function arguments.		
4	Write a Python program to demonstrate the precedence and associativity of operators.		
5	Write suitable Python program to implement recursion for problems such as Fibonacci series, Factorial, Tower of Hanoi etc.		
6	Write Python program to implement and use lambda function in python		
7	Write a python program to create and manipulate arrays in Python. Also demonstrate use of slicing and indexing for accessing elements from the array.		
8	Write a program to implement list in Python for suitable problem. Demonstrate various operations on it.		
9	Write a program to implement tuple in Python for suitable problem. Demonstrate various operations on it.		
10	Write a program to implement dictionary in Python for suitable problem. Demonstrate various operations on it.		

Evaluation Scheme

I. Internal Evaluation for Theory Courses – 25 Marks

(i) Mid-Term Class Test– 15Marks

- ☐ It should be conducted using any **learning management system** such as **Moodle** (Modular object-oriented dynamic learning environment)
- ☐ The test should have **15 MCQ's** which should be solved in a time duration of **30 minutes**.

(ii) Assignment/ Case study/ Presentations– 10 Marks

- ☐ Assignment / Case Study Report / Presentation can be uploaded on any **learning management system**.

II. External Examination for Theory Courses – 75 Marks

- ☐ Duration: **2.5 Hours**
- ☐ Theory question paper pattern:

	All questions are compulsory.		
Question	Based on	Options	Marks
Q.1	Unit I	<i>Any 4 out of 6</i>	20
Q.2	Unit II	<i>Any 4 out of 6</i>	20
Q.3	Unit III	<i>Any 4 out of 6</i>	20
Q.4	Unit I,II and III	<i>Any 5 out of 6</i>	15

- ☐ All questions shall be compulsory with internal choice within the questions.
- ☐ Each Question maybe sub-divided into subquestions as a, b, c, d, etc. & the allocation of Marks depends on the weightage of the topic.

III. Practical Examination

- ☐ Each core subject carries 50 Marks
40 marks + 05 marks (journal) + 05 marks (viva)
- ☐ Duration: **2 Hours** for each practical course.
- ☐ Minimum **80% practical** from each core subjects are required to be completed. ☐ **Certified Journal is compulsory for appearing at the time of Practical Exam** ☐ The final submission and evaluation of **journal in electronic form** using a Learning Management System / Platform can be promoted by college.
