**Syllabus for Python Programming BCA368**



|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **School:** | | **School of Engineering and technology** | | | | | | |
| **Department** | | **Department of Computer Science and Engineering** | | | | | | |
| **Program:** | | BCA | | | | | | |
| **Branch:** | |  | | | | | | |
| 1 | Course Code | BCA318 |  | | | | | |
| 2 | Course Title | Python Programming | | | | | | |
| 3 | Credits | 2 | | | | | | |
| 4 | Contact Hours  (L-T-P) | 2 | | | 0 | | 0 | |
|  | Course Status | Core /Elective/Open Elective | | | | | | |
| 5 | Course Objective | The objective of this course is to:   * Explain the basic syntax of Python Program * Explain various programming constructs –data types, decision structures, control structures in python * Know how to use in-built data structures in python – Lists, Tuples, Dictionary * Know how to use libraries for string manipulation and File handling   Learn the fundamental principles of Object-Oriented Programming  Using such knowledge small project can be made | | | | | | |
| 6 | Course Outcomes | **At the end of this course students will be able to:**   1. Demonstrate the fundamentals of python 2. Analyze and implement the concept of python data structure 3. Design function for a problem using python programming 4. Formulate the understanding of file handling 5. Discuss and implement the OOPs concept 6. Create accurate logical solution of any given problem | | | | | | |
| 7 | Course Description | This course starts with an introduction to Python, History of Python and basics syntax for writing Python Program. As the course progresses the study of decision structure, control structure and in-built data structure are studied in detail. This course mainly focuses on OOPs concepts. This course also deals with File handling, and Module concept. | | | | | | |
| 8 | Outline syllabus | | | | | | | CO Mapping |
|  | **Unit 1** | **Introduction to Python** | | | | | |  |
| A | History, Features, Working with Python, Installing Python, basic syntax to write a program, The concept of data types | | | | | | CO1 |
| B | Variables, Constants, Identifiers, keywords, Arithmetic and Logical operators and Boolean expressions. Debugging, comments in the program | | | | | | CO1 |
| C | Conditional Statements : If, If-else, Nested if-else; Looping: For, While, Nested loops; Control Statements: Break, Continue, Pass | | | | | | CO1 |
|  | **Unit 2** | **Lists, Tuples and Dictionaries** | | | | | |  |
| A | Lists; Creation, Attributes, Accessing, Operations, Searching and sorting in Lists; Linear, Binary; Bubble, Selection, Insertion | | | | | |  |
| B | Tuple; Accessing, operations, working with Tuples | | | | | |  |
| C | Dictionaries; Notations, Accessing, Operations, Working with Dictionaries | | | | | |  |
|  | **Unit 3** | **Functions, Recursion &String** | | | | | |  |
| A | Defining, Calling, Types of functions , Passing parameters with call by value and call by reference, Global and local variables | | | | | |  |
| B | Recursion, Writing recursive functions, Factorial Using recursion , Fibonacci series Using Recursion | | | | | |  |
| C | String; Accessing, Manipulation /Operation, String methods, Slicing. | | | | | |  |
|  | **Unit 4** | **Module, File Handling & Exception Handling** | | | | | |  |
| A | Importing Module, Creating Module, Packages, Math and Random Module | | | | | |  |
| B | Need of File Handling, Different modes of operation, Opening, Writing, Reading, Closing | | | | | |  |
| C | Exception, Exception Handling, Try and Except clause, Finally clause, | | | | | |  |
|  | **Unit 5** | **Object Oriented Programming Concepts** | | | | | |  |
| A | Overview of OOP concepts, Class and objects, Attributes | | | | | |  |
| B | Adding methods to a class, Passing an Object as Parameter to a method, Overloading; Method Overloading | | | | | |  |
| C | Inheritance; Types of inheritance(single, Multiple, Multi-level) | | | | | |  |
|  | Mode of examination | Theory/Jury/Practical/Viva | | | | | |  |
|  | Weightage Distribution | CA | | MTE | | ETE | |  |
| 30% | | 20% | | 50% | |  |
|  | Text book/s\* | 1. Tony Gaddis, Starting Out with Python, 3rd edition, Pearson  2. Y. Daniel Liang, Introduction to Programming Using Python, Pearson  3. Jason R .Briggs, Python For Kids, San Francisco  4. E Balagurusamy, Introduction to Computing & Problem solving Using Python, TMH | | | | | |  |
|  | Other References | 1. Downey, Allen B., Think Python: How to Think Like a Computer Scientist. O’Reilly, 2012. Obtain free PDF at http://www.greenteapress.com/thinkpython/  2. Python Programming: An Introduction to Computer Science (Second Edition) John Zelle, ISBN 978-1-59028-241-0-9, Franklin, Beedle & Associates Inc., 2003.  3. Budd T A, Exploring Python , 2011, [Tata McGraw Hill Education](https://www.shimply.com/publisher/tata-mcgraw-hill-education) | | | | | |  |

CO and PO Mapping

|  |  |  |
| --- | --- | --- |
| S. No. | Course Outcome | Program Outcomes (PO) & Program Specific Outcomes (PSO) |
| 1. | Demonstrate the fundamentals of python | **PO 1,** **PO2,PO3,PO4,PO6,PO7,PO10,PSO1,PSO2** |
| 2. | Analyze and implement the concept of python data structure | **PO 1, PO2,PO3,PO4,PO6,PO7,PO10,PSO1,PSO2** |
| 3. | Design function for a problem using python programming | **PO 1, PO2,PO3,PO4,PO5, ,PO6,PO7,PO10,PSO1,PSO2** |
| 4. | Formulate the understanding of file handling | **PO 1, PO2,PO3,PO4,PO5,PO6,PO7,PO10,PSO1,PSO2** |
| 5. | Discuss and implement the OOPs concept | **PO 1, PO2,PO3,PO4,PO5,PO6,PO7,PO10,PSO1,PSO2** |
| 6. | Create accurate logical solution of any given problem | **PO 1, PO2,PO3,PO4,PO5,PO6,PO7,PO10,PSO1,PSO2** |

**PO and PSO mapping with level of strength for Course Name** Python Programming **(Course Code** BCA318**)**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Code\_ Course Name** | **CO’s** | **PO 1** | **PO**  **2** | **PO**  **3** | **PO4** | **PO 5** | **PO 6** | **PO 7** | **PO 8** | **PO 9** | **PO 10** | **PSO**  **1** | **PSO2** |
| **BCA318** | **CO1** | 1 | 1 | 1 | 1 | - | 2 | 2 | - | - | 2 | 1 | 1 |
| **CO2** | 2 | 1 | 1 | 1 | - | 2 | 2 | - | - | 2 | 1 | 2 |
| **CO3** | 1 | 2 | 1 | 2 | - | 2 | 2 | - | - | 2 | 2 | 2 |
| **CO4** | 2 | 2 | 3 | 2 | 2 | 2 | 2 | - | - | 2 | 2 | 2 |
| **CO5** | 2 | 2 | 2 | 2 | 2 | 2 | 2 | - | - | 2 | 2 | 2 |
| **CO6** | 3 | 3 | 3 | 2 | 2 | 2 | 2 | -- | - | 2 | 3 | 3 |

***Average of non-zeros entry in following table (should be auto calculated).***

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Course Code** | **Course Name** | **PO 1** | **PO2** | **PO 3** | **PO 4** | **PO 5** | **PO 6** | **PO 7** | **PO 8** | **PO 9** | **PO 10** | **PSO 1** | **PSO 2** |
| **BCA318** | **Python Programming** | **1.8** | **1.8** | **1.8** | **1.6** | **1** | **2** | **2** | **-** | **-** | **2** | **1.83** | **2** |

***Strength of Correlation***

***1.*** Addressed to ***Slight (Low=1) extent 2.*** Addressed to ***Moderate (Medium=2) extent***

***3.*** Addressed to ***Substantial (High=3) extent***