

BCA/5 Years Integrated M.C.A. (4th Semester)

DSE4 030010411/060060408: Multi-paradigm Programming

Teaching Schedule

Objective: To develop an application through programming language which support multiple programming paradigm including object oriented, pattern matching, imperative, functional programming and protocols.

Course Outcomes: Upon completion of the course, students shall be able to

- C01: Apply the concepts of Lists, Dictionaries and Sequence to develop an application.
- C02: Utilize the concepts of object oriented, module and package.
- C03: Develop searching based application using re module.
- C04: Develop an application for handling Files and Directories.
- C05: Construct a module for sending mail using SMTP protocol.
- C06: Illustrate networking application using socket module.

Sub Unit	No. of Lecture (s)	Topics	Reference Chapter/ Additional Reading	Teaching Methodology to be used	Active Learning Activities	Evaluation parameter
1. Introduction						
1.1	1	Multi-paradigm Programming Language: Object-oriented, Imperative and Functional programming	http://people.cs.clemson.edu/~turner/courses/cs428/current/webct/content/pz/ch2/ch2_6.html	Chalk and Talk		
1.2	1	Lexical structure of Python	AM - #4 Page No. 33 – 38	Chalk and Talk		
1.3	1	Numbers and Operators	JP - #2 Page No. 15 – 28	Chalk and Talk		
1.4		Variables	JP - #3 Page No. 31 – 34	Chalk and Talk		
1.5	2	Built-in Types: Tuples, Lists, Sets and Dictionaries	JP - #3 Page No. 34 – 47	Chalk and Talk	For weak learners exercise shall be given to improve their logical skill	

1.6	1	print statement	AM - #4 Page No. 61 – 62 JP - #1 Page No. 08 – 13	Chalk and Talk		
2. Basic flow statement and Classes						
2.1	2	Making Decision: if statement and repetition	JP - #4 Page No. 51 – 65	Chalk and Talk		
2.2	1	Handling Errors: try statement	JP - #4 Page No. 65 – 67	Chalk and Talk		
2.3	1	Functions	JP - #5 Page No. 71 – 88	Chalk and Talk		Quiz
2.4	1	Classes and Object	JP - #6 Page No. 93 – 107	Chalk and Talk		
2.5	2	Module Objects: import statement, from statement	AM - #7 Page No. 139 – 143 JP - #10 Page No. 159	Chalk and Talk		
2.6	1	Packages	JP - #7 Page No. 118 – 121	Chalk and Talk		
3. Strings and Regular Expressions						
3.1	2	Methods of String objects: find(), join(), replace(), split(), splitlines(), title(), strip()	AM - #9 Page No. 186 – 191	Chalk and Talk, Topic Slides	For advanced learner: Exercise given to solve without using built-in methods	
3.2	2	String module	AM - #9 Page No. 191 – 193	Chalk and Talk, Topic Slides		
3.3	3	String formatting	AM - #9 Page No. 193 – 197	Chalk and Talk		Unit Test-1
3.4	3	Regular expression and the re Module: match() and search()	AM - #9 Page No. 201 – 212	Chalk and Talk, Demonstration		
4. Files and Directories						
4.1	2	File Objects	JP - #8 Page No. 127 – 131	Chalk and Talk		
4.2	1	Path and Directories	JP - #8 Page No. 131 – 132	Chalk and Talk		
4.3	1	os Module	JP - #8 Page No. 132 – 140	Chalk and Talk, Topic Slides		
4.4	2	Retrieval of data from XML file	JP - #15 Page No. 265 – 284	Chalk and Talk,		

				Demonstration, Topic Slides		
5. Protocols and Mail Server						
5.1	1	Comparing Protocols and Programming Languages	JP - #16 Page No. 289 – 290	Chalk and Talk		
5.2	3	Internet Protocol Stack	JP - #16 Page No. 290 – 291	Chalk and Talk		
5.3	2	MIME Multi-part Messages	JP - #16 Page No. 297 – 302	Chalk and Talk, Topic Slides		
5.4	3	Sending mail with SMTP	JP - #16 Page No. 303 – 306	Chalk and Talk, Demonstration, Topic Slides		Unit Test-2
6. Socket Programming						
6.1	1	Socket: Introduction	AM - #20 Page No. 520 – 523	Chalk and Talk		
6.2	3	Socket Class and Methods: socket(), connect(), send(), error() and close()	AM - #20 Page No. 523 – 528	Chalk and Talk, Demonstration	For all: Class activity to develop chat based system	
6.3	3	SocketServer Module: handle(), request() and server()	AM - #20 Page No. 528 – 532	Chalk and Talk, Demonstration, Topic Slides		
6.4	2	Event-Driven Socket Program: select Moduleselect()	AM - #20 Page No. 533 – 544	Chalk and Talk, Demonstration, Topic Slides		Internal
References :						
<ol style="list-style-type: none"> 1. J. Payne, "Beginning Python", Wrox.[JP] 2. A. Martelli, Python in a nutshell, O Reilly.[AM] 						
Note: # denotes chapter number.						

Course objectives and Course outcomes mapping:

To develop an application through object oriented concepts: C01, C02

To support multiple programming paradigm such as pattern matching and imperative: C01, C03, C04

To use functional programming and protocols: C05, C06

Course units and Course outcome mapping:

Unit No.	Unit	Course outcome
----------	------	----------------

		C01	C02	C03	C04	C05	C06
1	Introduction	✓					
2	Basic flow statement and Classes	✓	✓				
3	Strings and Regular expressions	✓	✓	✓			
4	File and Directories		✓		✓		
5	Protocols and Mail Server		✓			✓	
6	Socket Programming						✓

Programme Outcomes:

P01: Proficiency in and ability to identify problems related to computer science as well as design and apply computational knowledge to solve them.

P02: Ability to design, develop, test and maintain system, component, product or process as per needs and specification.

P03: Understanding of professional and ethical role and responsibility.

P04: Recognition of the need for and ability towards life-long learning

P05: Knowledge of programming languages, database systems, operating systems, software engineering, Web & Mobile technology and relevant modern issues.

P06: Ability to demonstrate the use of modern tools, models and languages to solve problems related to software development.

P07: Ability to communicate and present knowledge effectively.

Course outcome and Programme Outcome mapping:

Course Outcomes	Program Outcomes					
	P01	P02	P03	P04	P05	P06
C01		✓				✓
C02		✓		✓		✓
C03	✓	✓		✓		✓
C04		✓	✓	✓	✓	✓
C05		✓	✓	✓	✓	✓
C06		✓	✓	✓	✓	✓

Computing Environment:

A student must have the following computing environment available in laboratory as well as in his/her personal laptop.

- Python IDLE 3.5

Number of Practical Problems in Journal: 16

Total sets to be developed for each division: 2

Unit Number	Number of Questions	Time required to implement and debug the question (in hours)	Minimum required of Journal Certification
1& 2	2	4	2
3	3	10	3
4	4	12	4
5	3	10	3
6	4	12	3
TOTAL	16	48	15

Concept Linkage:

Unit/ Sub-Unit	Prior concept linkage	Post concept linkage
2: Making Decisions, 3:Repetitions, 5:Functions	060060111(Fundamentals of Programming)	060060506(Introduction to server side programming): 1.1, 1.2, 3.1, 3.2
3: Classes and Objects, 4: Inheritance	060060306(Java Programming)	