


CMR Institute of Technology, Bangalore		
Department(s) : MCA		
Semester: 02	Section(s):	
Subject: Python Programming	16MCA21	Lectures /Week :05
Course Instructor(s) : Kavitha K		
Course Duration : Febraury 2017 to May 2017		

Course Objectives: This subject provides students with

1. Basics python programming
2. Using Modules and functions
3. Using collections

Course Prerequisites:No prior knowledge of programming is required

Course Outcome (CO):

At the end of this course, the students will be able to

CO1: Understand and comprehend the basics of python programming.

CO2: Apply knowledge in real time applications.

CO3: Understands about files and its applications.

Lecture #	Book & Sections	Topics	Portions Coverage %	
			Individual	Cumulative
1 to 10	TB1: 1-73	Module1 : Introduction: Installing Python, Simple program using Python, Expressions and Values, Variables and Computer Memory, error detection, Multiple line statements, Designing and using functions, functions provided by Python, Tracing function calls in memory model, omitting return statement. Working with Text: Creating Strings of Characters, Using Special Characters in Strings, Creating a Multiline String, Printing Information, Getting Information from the Keyboard.	10	20
11 to 20	TB1: 77-123	Module 2: A Boolean Type , Choosing Statements to Execute, Nested If Statements , Remembering the Results of a Boolean Expression Evaluation , A Modular Approach to Program Organization, Importing Modules ,	20	40


		Defining Your Own Modules, Testing Code Semi automatically Grouping Functions Using Methods: Modules, Classes, and Methods , Calling Methods the Object- Oriented Way, Exploring String Methods, Underscores.		
21 to 30	TB1: 129-161	Module 3: Storing Collections of Data Using Lists: Storing and Accessing Data in Lists, modifying Lists, Operations on Lists, Slicing Lists, Aliasing, List Methods, Working with a List of Lists. Repeating Code Using Loops: Processing Items in a List, Processing Characters in Strings, Looping Over a Range of Numbers, Processing Lists Using Indices, Nesting Loops in Loops, Looping Until a Condition Is Reached, Repetition Based on User Input, Controlling Loops Using Break and Continue Reading and Writing	10	60
31 to 40	TB1: 171-218	Module 4: Kinds of files, Opening a File, Techniques for Reading Files, Files over the Internet, Writing Files, and Writing Algorithms That Use the File-Reading Techniques, Multiline Records. Storing Data Using Other Collection Types: Storing Data Using Sets, Storing Data Using Tuples, Storing Data Using Dictionaries, Inverting a Dictionary, Using the In Operator on Tuples, Sets, and Dictionaries, Comparing Collections.	10	80
41 to 50	TB1: 269-293 317-336	Module 5: Collection of New Information Object-Oriented Programming : Understanding a Problem Domain , Function “Isinstance,” Class Object, and Class Book , Writing a Method in Class Book, Plugging into Python Syntax: More Special Methods ,Creating Graphical User interface: Building a Basic GUI, Models, Views, and Controllers, Customizing the Visual Style Introducing few more Widgets, Object-Oriented GUIs, Keeping the Concepts from Being a GUI Mess.	10	100

Syllabus for Internal Assessment Test

Internal Assessment Test	Syllabus
T1	Class # 01 - 24

T2	Class # 25 - 48
T3	Class # 49 - 62

Literature Book Type	Code	Author & Title	Publication Info	
			Edition & Publisher	ISBN #
Text Book	TB1	Paul Gries , Practical Programming	2 nd Edition, Pragmatic Bookshelf	ISBN-13: 978-1-93778-545-1
Text Book	TB2	Allen Downey, Learning Python-How to think like a Computer Scientist	Green Tea Publisher	ISBN 0-9716775-0-6

CMR Institute of Technology, Bangalore		
Department(s): Master of Computer Applications		
Semester: 02	Section : A	
Object Oriented Programming using C++	16MCA22	Lectures/week: 06
Course Instructor(s): Ms. Neha Agrawal		
Course duration : 6 Feb 2017 – 21 May 2017		

Course Objectives:

The course presents basics of C++ programming including: Basics of C++ environment, Data representation, Control structures, Functions, Arrays, Pointers, Strings, and Classes that aims to:

1. Arm the students with the basic programming concepts.
2. Introduce different techniques pertaining problem solving skills
3. Arm the students with the necessary constructs of C++ programming.
4. And to emphasis on guided practical sessions

Prerequisites:

C programming language. Concepts of variables, data types, functions, pointers

Course Outcome (CO): At the end of this course, the students will be able to

CO1: Differentiate between object oriented programming and procedure oriented programming & Disseminate the importance of Object oriented programming

CO2: Apply C++ features such as Classes, objects, constructors, destructors, inheritance, operator overloading, and Polymorphism, Template and exception handling in program design and implementation.

CO3: Use C++ to demonstrate practical experience in developing object-oriented solutions.

CO4: Analyze a problem description and build object-oriented software using good coding practices and techniques.

CO5: Implement an achievable practical application and analyze issues related to object-oriented techniques in the C++ programming language

Class #	Chapter Title / Reference Literature	Topic	Percentage of portion covered	
			Reference	Cumulative
1	TB1 : 11 Chapter	Overview of C++: The Origins of C++ , What Is Object-Oriented Programming? ,Encapsulation ,Polymorphism , Inheritance. Some C++ Fundamentals	9%	9%
2		A Sample C++ Program ,A Closer Look at the I/O Operators, Declaring Local Variables ,No Default to int, The bool Data Type		
3		Old-Style vs. Modern C++ , The New C++ Headers , Namespaces, Working with an Old Compiler		
4		Introducing C++ Classes ,Function Overloading, Operator Overloading		
5		Inheritance ,Constructors and Destructors ,		
6		The C++ Keywords, The General Form of a C++ Program.		
7	TB1 :12 chapter	Classes and Objects: Classes, Structures and Classes Are Related, Unions and Classes Are Related , Anonymous Unions	11%	21%
8		Friend Functions, Friend Classes, Inline Functions, Defining Inline Functions Within a Class		
9		Parameterized Constructors, Constructors with One Parameter: A Special Case		
10		Static Class Members ,Static Data Members ,Static Member Functions		
11		When Constructors and Destructors Are Executed		
12		The Scope Resolution Operator ,Nested Classes, Local Classes,		
13	Passing Objects to Functions ,Returning Objects ,Object Assignment.			
14	TB1 :13 chapter	Arrays, Pointers, References, and the Dynamic Allocation Operators: Arrays of Objects, Creating Initialized vs. Uninitialized Arrays , Pointers to Objects,	9%	30%
15		Type Checking C++ Pointers , The this Pointer, Pointers to Derived Types		
16		Pointers to Class Members , References, Reference Parameters, Passing References to Objects		
17		Returning References, Independent References, References to Derived Types, Restrictions to References		
18		A Matter of Style, C++'s Dynamic Allocation Operators ,Initializing Allocated Memory ,Allocating Arrays		
19		Allocating Objects ,The nothrow Alternative ,The Placement Form of new		
20	TB1 :14 chapter	Function Overloading, Copy Constructors and Default Arguments: Function Overloading	9%	40%
21		Overloading Constructors		
22		Overloading a Constructor to Gain Flexibility, Allowing Both Initialized and Uninitialized Objects Copy Constructors		
23		Finding the Address of an Overloaded Function , The overload Anachronism		
24		Default Function Arguments, Default Arguments vs. Overloading		
25		Function Overloading and Ambiguity.		
26	TB1 :15 Chapter	Operator Overloading: Creating a Member Operator Function	9%	50%

27		Creating Prefix and Postfix Forms, of the Increment and decrement Operators		
28		Overloading the Shorthand Operators, Operator Overloading Restrictions		
29		Operator Overloading Using a Friend Function, Using a Friend to Overload ++ or --, Friend Operator Functions Add Flexibility		
30		Overloading new and delete, Overloading new and delete for Arrays, Overloading the nothrow Version of new and delete		
31		Overloading Some Special Operators, Overloading [] , Overloading (), Overloading ->, Overloading the Comma Operator.		
32	TB1 :16 chapter	Inheritances: Base-Class Access Control, Inheritance and protected Members	9%	60%
33		Protected Base-Class Inheritance		
34		Inheriting Multiple Base Classes		
35		Constructors, Destructors, and Inheritance, When Constructors and Destructors Are Executed		
36		Passing Parameters to Base-Class Constructors		
37		Granting Access, Virtual Base Classes.		
38	TB1 :17 chapter	Virtual Functions and Polymorphism: Virtual Functions, Calling a Virtual Function Through a Base, Class Reference	5%	66%
39		The Virtual Attribute Is Inherited, Virtual Functions Are Hierarchical		
40		Pure Virtual Functions, Abstract Classes, Using Virtual Functions, Early vs. Late Binding.		
41	TB1:18 chapter	Templates: Generic Functions, A Function with Two Generic Types, Explicitly Overloading a Generic Function, Overloading a Function Template	6%	72%
42		Using Standard Parameters with Template Functions , Generic Function Restrictions , Applying Generic Functions, A Generic Sort, Compacting an Array		
43		Generic Classes, An Example with Two Generic Data Types, Applying Template Classes: A Generic Array Class, Using Non-Type Arguments with Generic Classes		
44		Using Default Arguments with Template Classes, Explicit Class Specializations, The typename and export Keywords, The Power of Templates		
45	TB1 :19 chapter	Exception Handling: Exception Handling Fundamentals, Catching Class Types, Using Multiple catch Statements	8%	81%
46		Handling Derived-Class Exceptions, Exception Handling Options, Catching All Exceptions, Restricting Exceptions		
47		Rethrowing an Exception, Understanding terminate() and unexpected()		
48		Setting the Terminate and Unexpected Handlers, The uncaught_exception() Function		
49		The exception and bad_exception Classes Applying Exception Handling		
50	TB1 :20 Chapter	Standard C++ I/O Classes : Old vs. Modern C++ I/O, C++ Streams, The C++ Stream Classes, C++'s Predefined Streams, Formatted I/O	6%	87%
51		Formatting Using the ios Members, Setting the Format Flags, Clearing Format Flags, Overloading << and >>		
52		Creating Your Own Inserters, Creating Your Own Extractors, Creating Your Own Manipulator Functions		
53		C++ File I/O: fstream> and the File Classes, Opening and Closing a File, Reading and Writing Text Files, Unformatted and Binary I/O		

54	TB1 :21 Chapter	STL : An Overview of the STL, Containers	6%	94%
55		Algorithms, Iterators and Function Objects		
56		The String Class, Vector Class		
57		Lists, Maps.		
58	TB1 :24 chapter	Characters vs. Bytes, put() and get()	8%	100%
59		read() and write(), More get()		
60		Functions, getline() , Detecting EOF		
61		The ignore() Function, peek() and putback(), flush(), Random Access		
62		Obtaining the Current File Position , I/O Status, Customized I/O and Files.		


Syllabus for Sessionals :

Sessional #	Syllabus
T1	Class # 01 - 19
T2	Class # 20 -53
T3	Class # 54- 62

Literature

:

Book Type	Code	Author & Title	Publication info	
			Edition & Publisher	ISBN #
Text Book	TB1	Herbert Schildt: C++ The Complete Reference, 4th Edition,	Tata McGraw Hill, 2007.	ISBN=0071502394
Text Book	TB2	K R Venugopal, Rajkumar Buyya, T Ravishanker: Mastering C++	Tata McGraw Hill	10:0-07-463454-2
References	RB1	Al Stevens: C++ Programming, 7th Edition,	Wiley India Publications	ISBN-13: 978-81-265-0398-8
References	RB2	Stanley B.Lippmann, Josee Lajore: C++Primer, 4th Edition,	Addison Wesley, 2005.	ISBN 0-321-71411-3

CMR Institute of Technology, Bangalore			
Department(s): Master of Computer Applications			
Semester: 02	Section(s): A		
Database Management System	16MCA23	Lectures/week: 05	
Course Instructor(s): Ms. Gomathi T			
Course duration: Feb 2017 – May 2017			

Course Outcome:

By end of this course the students will be able to

CO1: Demonstrate the fundamentals of data models and conceptualize and depict a database system and make use of ER diagram in developing ER Model.

CO2: To summarize the SQL and relational database design.

CO3: Illustrate transaction processing, concurrency control techniques and recovery.

CO4: Inference the database design in the real world entities.

Class #	Chapter Title / Reference Literature	Topic	Percentage of portion covered	
			Reference	Cumulative
1	Unit 1 : INTRODUCTION	Characteristics of Database approach. Actors on the screen; Workers behind the scene		
2		Advantages of using DBMS approach. Data models, schemas and instances		
3		Three-schema architecture and data independence.		
4		Database Languages and interfaces, the database system environment, Centralized and Client server architecture		
5		Classification of Database Management System		
6		ER Model- Conceptual database using high level conceptual data models for database design.		
7		A sample database applications, Entity types, Entity set attributes and keys		
8		Relationship types, Relationship sets		
9		Roles and structural constraints, Weak Entity type		
10		Exercise on ER Diagram		
11		Exercise on ER Diagram		
12		Exercise on ER Diagram		
13	Unit 2 : RELATIONAL MODEL	Relational Model Concepts.		
14		Relational Model Constraints and Relational Database Schemas.		
15		Relational Model Constraints and Relational Database Schemas.		
16		Update Operations, Transactions and dealing with constraint violations;		
17		Update Operations, Transactions and dealing with constraint violations;		
18		Unary Relational Operations: SELECT and PROJECT; Relational Algebra Operations from Set Theory.		
19		Binary Relational Operations : JOIN and DIVISION		
20		Binary Relational Operations : JOIN and DIVISION		
21		Additional Relational Operations.		
22		Additional Relational Operations.		
23		Examples of Queries in Relational Algebra		
24		Examples of Queries in Relational Algebra		


25	Unit 3 : Introduction to SQL	Overview of the SQL Query Language, SQL Data Definition		
26		Basic structure of SQL Queries, Additional Basic Operations		
27		Null values, Aggregate functions, nested sub queries		
28		Modification of the database,		
29		Join Expression,		
30		Views and transactions		
31		Integrity constraints,		
32		SQL Data Types and Schemas ,Authorization		
33		SQL Data Types and Schemas ,Authorization		
34		Database programming issues and techniques		
35		Database programming issues and techniques		
36	Embedded SQL			
37	Unit 4 : Database Design	Informal Design Guidelines for Relation Schemas.		
38		Functional Dependencies.		
39		Functional Dependencies.		
40		Normal Forms Based on Primary Keys.		
41		Normal Forms Based on Primary Keys.		
42		General Definitions of Second and Third Normal Forms.		
43		Boyce-Codd Normal Form.		
44		Boyce-Codd Normal Form.		
45		Stored Procedures and functions		
46		Stored Procedures and functions		
47	Triggers			
48	Triggers			
49	Unit 5 : Transaction Management	Transaction Concept		
50		A Simple transaction Model		
51		Transaction atomicity and Durability		
52		Serializability		
53		Transaction isolation levels		
54		Implementation of isolation levels		
55		Concurrency control		
56		Lock based protocols		
57		Deadlock handling		
58		Recovery System: Failure Classification		
59		Storage		
60		Recovery and Atomicity		
61		Recovery algorithm		
62		Recovery algorithm		

Syllabus for Sessionals:

Sessional #	Syllabus
T1	Class #1 -17 ,Class # 26-40
T2	Class # 18 -25, Class # 41-62

Literature:

Book Type	Code	Author & Title	Publication info	
			Edition & Publisher	ISBN #
Text Book	TB1	Elmasri and Navathe, "Fundamentals of Database Systems".	5 th edition, Pearson Education	978-81-317-1625-0
Text Book	TB2	Raghu Ramakrishnan and Johannes Gehrke, "Database Management Systems.	3 rd Edition, McGraw-Hill, 2003.	0-07-123151-X
Reference Book	RB2	Silberschatz, Korth and Sudharshan: Data base System Concepts.	5 th Edition, Mc-GrawHill ,2006	0-07-295886-3
References	RB2	Database Principles Fundamentals of Design, Implementation and Management by Coronel, Morris	Rob-Cengage Learning,2012	978-1-133-31197-3

CMR Institute of Technology, Bangalore			
Department(s): Master of Computer Applications			
Semester: 02	Section(s): A		
Operating Systems	10MCA24	Lectures/week: 05	
Course Instructor(s): Ms. Helen Josephine VL			
Course duration: Feb 2017 – June 2017			

At the end of this course, the students will be able to

- CO1: Understand the Basics of Computer and Operating Systems Structure**
- CO2: Realize the concept of Process Management and Mutual Execution**
- CO3: Understand the concepts of the Deadlock and different approaches to memory management.**
- CO4: Learn the concepts of file system**
- CO5: Understand the concepts of Computer Security.**

Class #	Chapter Title / Reference Literature	Topic	Percentage of portion covered	
			Reference	Cumulative
1	TB2:1.1 TO 1.7 TB1: 1.1 TO 1.10 TB1:2.1 TO 2.6	Computer and Operating Systems Structure: Basic Elements, Processor Registers, Instruction Execution	17%	17%
2		The Memory Hierarchy, Cache Memory, I/O Communication Techniques, Introduction to Operating		

		System		
3		Mainframe Systems, Desktop System		
4		Multiprocessor Systems, Distributed Systems, Clustered Systems,		
5		Real - Time Systems, Handheld Systems		
6		Feature Migration, Computing Environments.		
7		System Components, Operating – System Services		
8		System Calls, System Programs		
9		System Structure, Virtual Machines		
10		System Design and Implementation		
11		System Generation		
12		Process Management and Mutual Execution: Process, Process States		
13		Process Description, Process Control		
14		Execution of the Operating System		
15		Security Issues, Processes and Threads		
16		Symmetric Multiprocessing(SMP), Micro kernels		
17	TB2:3.1 TO 3.04	CPU Scheduler and Scheduling	19%	36%
18	TB2:4.1 TO 4.3 TB2:5.1 TO 5.6	Principles of Concurrency		
19		Mutual Exclusion: Hardware Support		
20		Semaphores		
21		Monitors		
22		Message Passing		
23		Readers/Writes Problem		
24		Deadlock and Starvation : Principles of Deadlock		
25	TB2:6.1 TO 6.6 TB2: 7.1 TO 7.5 TB1:10.1 TO 10.6	Deadlock Prevention, Deadlock Avoidance,	20%	56%
26		Deadlock Detection,An Integrated Deadlock Strategy		
27		Dining Philosophers Problem		

28		Memory Management :Swapping		
29		Contiguous Memory Allocation		
30		Paging, Segmentation		
31		Segmentation with Paging		
32		Demand Paging, Examples		
33		Process Creation, Examples		
34		Page Replacement, Examples		
35		Allocation of Frames,Thrashing with Examples		
36		File System and secondary storage: File concept		
37		Access Methods		
38		Directory Structure, File – System Mounting,		
39		Directory Structure, File – System Mounting,		
40		File – System Mounting		
41		File – System Mounting		
42	TB1: 11.1 TO 11.6	File Sharing,		
43	TB1: 12.1 TO 12.5	Protection,	24%	80%
44	TB1: 14.1 TO 14.2	File – System Structure		
45		File – System Implementation		
46		Directory Implementation		
47		Allocation Methods		
48		Free –Space Management		
49		Disk Structure		
50		Disk Scheduling		
51		Computer Security and Case study of Linux Operating system : The Security Problem		
52	TB1: 19.1 TO 19.4	The Security Problem	20	100
53	TB1: 20.1 TO 20.9	User Authentication		
54		Program Threats		
55		System Threats		


56	Linux System Linux history
57	Design Principles
58	Kernel modules
59	Process , management
60	Scheduling, Memory management
61	File systems, Input and output
62	Inter-process communications

Syllabus for Sessionals :

Sessional #	Syllabus
T1	#1 - #30
T2	#31 - #62

Literature:

Book Type	Code	Author & Title	Publication info	
			Edition & Publisher	ISBN #
Text Book	TB1	Silberschatz, Galvin, Gagne, "Operating System Concepts" John Wiley,	Sixth Edition, 2004	978-0471417439
Text Book	TB2	William Stallings, "Operating Systems – Internals and Design Principles" Pearson,	6 th edition, 2012	10: 0133805913
References	RB1	Chakraborty , "Operating Systems" Jaico Publishing House	, 2011.	9788179929766
References	RB2	Dhananjay M. Dhamdhere, "Operating Systems – A Concept – Based Approach", Tata McGraw – Hill,	3rd Edition, 2012	9781259005589

CMR Institute of Technology, Bangalore		
Department(s): Master of Computer Applications		
Semester: 02	Section(s): A	
System Software	10MCA25	Lectures/week: 05
Course Instructor(s): Ms. Ashwini Patil		
Course duration: Feb 2017 – June 2017		

Course Objective:

To view some of the major tasks of the system software of a computer system, focusing on internal working of the hardware and software interface of a typical system..

Course Pre-requisites:

Student should have basic knowledge of mnemonic codes.

Course Outcome:

CO1: Understand the introductory concepts of system software, SIC and SIC/XE machine architecture.

CO2: Understand the design and implementation of Assemblers with implementation examples.

CO3: Design and implement the linkers and loaders, macro processors and respective implementation examples.

CO4: Learn the basic design and working of compilers.

Class #	Chapter #	Topics to be covered	% of portions covered	
			Reference Chapter	Cumulative
1	Unit –1 Machine Architecture & Introduction to Assemblers	Introduction to System Software & Machine architecture.	15.00	15.00
2		SIC/standard version -Machine Architecture – Memory, Registers, Data Formats, Instruction format, Addressing Modes, Instruction Set, Input and Output.		
3		SIC/XE –Machine Architecture – Memory, Registers, Data Formats, Instruction format, Addressing Modes, Instruction Set, Input and Output.		
4		SIC – Programming Examples (Data movement, arithmetic operation & Indexing, looping)		
5		Traditional CISC Machines-VAX Architecture		
6		RISC Machines-UltraSPARC Architecture		
7		Basic Assembler Functions A Simple SIC Assembler		
8		Assembler algorithm and Data Structure		
9		Pass I Algorithm		
10		Pass II Algorithm		
11		Machine-Dependent Assembler Features - Instructions Formats and Addressing Modes	25.00	40.00

12	Unit – 2 Assembler	Program Relocation criteria		
13		Machine-Independent Assembler Features- Literals		
14		Symbol Defining Statements		
15		Expressions		
16		Program Blocks		
17		ControlSection and Program Linking		
18		Assembler Design Options One-Pass Assemblers		
19		Multi Pass - Assemblers		
20		Implementation Example-MASM assembler		
21		SPARC assembler		
22		Unit 3- Loaders and Linkers		
23	Simple Bootstrap Loader			
24	Machine-Dependent Loader Features- Relocation			
25	Program Linking			
26	Algorithm and Data Structures for Linking Loader			
27	Machine-Independent Loader Features - Automatic Library Search			
28	Loader Options			
29	Loader Design options- Linkage Editors			
30	Dynamic Linking			
31	Bootstrap Loader			
32	Implementation Example-MSDOS Linker			
33	SunOS Linker			
34	Cary MPP Linker			
35	Unit-4 Macro Processor		Basic Macro Processors Functions Macro Definition and Expansion	20.00
36		Macro Processors – Data Structures		
37		Algorithm for a One – Pass Macro Processor		
38		Machine Independent Macro Processor features - Concatenation of Macro Parameters, Generation of Unique Labels		
39		Conditional Macro Expansion		

40		Keyword Macro Parameters		
41		Recursive Macro Expansion		
42		General- Purpose Macro Processors		
43		Macro Processing Within Language Translators		
44		Implementation Examples- MASM Macro Processor		
45		ANSIC Macro Language		
46		ELENA macro processor		
47	Unit-5 Compilers	Basic Compilers Functions- Grammars	20.00	100
48		Lexical Analysis		
49		Syntactic Analysis		
50		Code Generation		
51		Machine Dependent Compiler Features- Intermediate Form of the Program		
52		Machine dependent code Optimization		
53		Machine Independent Compiler Features- Structured variables		
54		Machine Independent code Optimization		
55		Compiler Design Options- Division into passes		
56		Interpreters		
57		P-code Compilers		
58		Compiler-Compilers		
59		SunOS C compiler,		
60		YACC Compiler-compiler.		

Syllabus for Sessionals :

Sessional #	Syllabus
T1	Class # 01 – 21
T2	Class # 21 – 40
T3	Class # 41 – 60

Literature:

Book Type	Code	Author & Title
Text Book	TB1	Leland.L.Beck: System Software, 3rd Edition, Addison-Wesley, 1997.
References	RB1	J.Nithyashri, "System Software", 2nd Edition, Tata McGraw Hill, 2010