

UNIVERSITY OF MUMBAI



Syllabus for Sem – V & VI
Program: B.Sc.
Course: Applied Component
Computer Science

(Credit Based Semester and Grading System
with effect from the academic year 2018-2019)

SEMESTER V**Theory**

| | | | | |
|------------------|--|--|----------------------|-----------------------|
| USACCS501 | Advanced Microprocessor, Microcontrollers & Python Programing | | No of Credits | Lectures/ Week |
| Units | I | Advanced 8085 Microprocessor Programming | 2 | 4 |
| | II | Introduction to 8051 microcontroller , Instruction set & Programming | | |
| | III | Python Programming Language | | |
| | IV | Python Programming Language | | |

Practicals

| | | | |
|------------------|---|----------|----------|
| USACCS5P1 | Microprocessors , Microcontrollers , Python Programming | 2 | 4 |
|------------------|---|----------|----------|

SEMESTER VI**Theory**

| | | | | |
|------------------|------------|----------------------|----------------------|-----------------------|
| USACCS601 | | | No of Credits | Lectures/ Week |
| Units | I | 8051 microcontroller | 2 | 4 |
| | II | Python Programming | | |
| | III | Python Programming | | |
| | IV | Python Programming | | |

Practicals

| | | | |
|-----------------|--------------------------------------|----------|----------|
| USACC6P1 | Microcontroller , Python Programming | 2 | 4 |
|-----------------|--------------------------------------|----------|----------|

The revised syllabus under the credit based grading system in the subject of Computer Science (Applied Component) for Third Year B.Sc. Physics (Single/Twin major subject) **will be implemented from the academic Year 2018-19.**

The scheme of examination in the subject of Computer Science (Applied Component) will be as follows:

SEMESTER V & VI: THEORY

COURSE CODE: USACCS501 & USACS601

External Examination: 100 Marks

- Duration of each Theory paper will be three **hours**.
- Each theory paper shall consist of **five questions**, **one** from each unit and the **fifth question** will be from all the units. All questions are compulsory and will have internal choice.
- The theory examination will be conducted by the respective colleges and the marks will be forwarded to the University.

Objectives

The objective of this paper is to introduce the students to advanced microprocessors , microcontrollers and various concepts of programming to the students using Python as the programming language.

Expected learning outcomes

1. Develop a basic understanding of programming and the Python programming Language.
2. Students will use their problem solving abilities to implement programs in Python.
3. Familiarization about the basic constructs of programming such as data, operations, conditions, loops, functions etc.
4. Introduction to advanced topics in Python such as Exception Handling, Multithreaded programming, Graphical user interface & Database connectivity.

SEMESTER V
COURSE CODE: USACCS501

**Advanced 8085 Programing, Introduction to Microcontrollers & Python
Programming (Version 3.x for Python)**

| | | |
|-----------------|---|------------|
| Unit- I: | Advanced 8085 Programming and 8255 (PPI) | (15 lect.) |
| 1. | Introduction to advanced instructions and applications Ref. RG: 10.7 (entire) | |
| 2. | Stack and Subroutines: Stack, Subroutine Ref. RG: 9.1, 9.1.1, 9.2 &9.2.1 | |
| 3. | The 8255 Programmable Peripheral Interface : Block Diagram of the 8255, Mode 0 – Simple Input or Output mode, BSR (Bit Set/Reset Mode) Ref. RG: 15.1.1, 15.1.2& 15.1.3 | |
| Unit-II: | Introduction to Microcontrollers, Instruction Set & Programing | (15 lect.) |
| 1. | Introduction, Microcontrollers and Microprocessors, History of Microcontrollers and Microprocessors, Block diagram of 8051 Microcontroller*, Embedded Versus External Memory Devices, 8-bit & 16-bit Microcontrollers, CISC and RISC Processors, Harvard and Von Neumann Architectures, Commercial Microcontrollers. Ref. AVD-Ch: 1 Ref. MMM - For * Refer 1.2 The 8051 Microcontroller & Embedded Systems by M.A. Mazidi, J.G. Mazidi and R.D.Mckinlay, Second Edition, Pearson. | |
| 2. | 8051 Microcontrollers : Introduction, MCS-Architecture, Registers in MCS-51,8051 Pin Description, 8051 Connections, 8051 Parallel I/O Ports, Memory Organization. Ref. AVD-Ch: 2, 3. | |
| 3. | 8051 Instruction Set and Programming: MCS-51 Addressing Modes and Instructions: 8051 Addressing modes, MCS-51 Instruction Set, 8051 Instructions and Simple Programs, Using Stack Pointer Ref. AVD -Ch: 4 Ref. AVD: - Microcontrollers (Theory and Applications) by Ajay V Deshmukh, The Tata-McGraw-Hill Companies Ref. Intel's 8031/8051 Data sheet. | |

| | | |
|-------------------|--|------------|
| Unit- III: | Basics of Python , Functions & Conditional statements: | (15 lect.) |
| 1. | Introduction: What is a Program, The Python Programming Language, History, features, Installing Python, Running a Python program, the first program, Arithmetic operators , Values and types, Formal and Natural Languages AD: Think Python , 2nd Edition : Chapter 1. | |
| 2. | Variables , Expressions and Statements : Assignment statements , Variable Names and Keywords, Expressions and statements , Script mode , Order of Operations , String operations , Comments , Debugging : Syntax Errors, Runtime Errors, Semantic Errors. AD: Think Python , 2nd Edition : Chapter 2. | |
| 3. | Functions: Function basics, Function Calls, Math Functions, Composition, Adding New Functions, Definitions and Uses, Flow of Execution, Parameters and Arguments, Local variables and parameters, Stack Diagrams, Fruitful Functions and void Functions , return values , composition , incremental development , Boolean functions AD: Think Python , 2nd Edition : Chapter 3 AD: Think Python , 2nd Edition : Chapter 6 : Art : 6.1 , 6.2 , 6.3 and 6.4 | |
| 4. | Conditionals and recursion : Floor division and modulus, Boolean expression , Logical operators, Conditional expression, Alternative execution , chained conditionals , Nested conditionals, Recursion , Stack diagrams for recursive functions , infinite recursion , keyboard input. Programs on recursion AD: Think Python , 2nd Edition : Chapter 5 AD : Think Python , 2nd Edition : Chapter 6 : Art : 6.5 ,6,6 ,6,7 & 6.8 | |
| Unit-IV: | Iterations , Strings & Lists in Python | (15 lect.) |
| 1. | Iterations : Reassignment , updating variables , while statement, break statement AD: Think Python , 2nd Edition : Chapter 7 up to 7.4 | |
| 2. | Strings: A String is a Sequence, len built in function , for Loop traversal , String Slices, Strings Are Immutable, Searching, Looping and Counting, String Methods, The in Operator, String Comparisons. AD: Think Python , 2nd Edition : Chapter 8 | |
| 3. | Lists: A list is a sequence, Lists are mutable, Traversing a List, List operations , List slices, List methods , Deleting elements , Lists & Strings , Objects & Values , Aliasing , List arguments. AD: Think Python , 2nd Edition : Chapter 10 | |

| | |
|---|---|
| References: For units I and II:- | |
| 1. | RG: Ramesh Gaonkar "Microprocessor Architecture Programming and Applications with the 8085" (RG) 5th edition Penram. |
| 2. | AVD: Microcontrollers (Theory and Applications) by Ajay V Deshmukh, The Tata-McGraw-Hill Companies. |
| 3. | Intel's 8031/8051 Data sheet |
| References: For units III and IV:- | |
| 1. | Official Python Web site : https://www.python.org/ |
| 2. | AD : Think Python by Allen Downey , 2nd Edition |
| Additional References: | |
| 1. | The 8051 Microcontroller & Embedded Systems-Dr. Rajiv Kapadia (Jaico Pub. House) |
| 2. | 8051 Micro-controller by K.J.Ayala, Penram International. |
| 3. | Programming & customizing the 8051 microcontroller By MykePredko, TMH. |
| 4. | The 8051 Microcontroller & Embedded Systems by M.A. Mazidi, J.G. Mazidi and R.D.Mckinlay, Second Edition, Pearson. |
| 5. | Paul Gries, et al., Practical Programming: An Introduction to Computer Science Using Python 3, Pragmatic Bookshelf, 2/E 2014. |

In addition to the reference books **internet web-sites & MOOC** can be used wherever necessary.

PRACTICALS (USACCS5P1)

| Group A: Advanced 8085 microprocessor & Microcontrollers | | | | | | | | | |
|---|--|----------|-----------|------|------|------|-------------------|-------------------|--|
| A1: | Experiments for 8085 and 8255 Any one experiment from 1 & 2 and 3 & 4 | | | | | | | | |
| 1 | 16-bit Data manipulation (Addition, subtraction) Display result on Address field | | | | | | | | |
| 2 | Write ALP for Addition, Subtraction of two, 8-bit hex numbers. Note: Use Read Keyboard Utility for inputting the hex numbers and display for the result on the Address field. | | | | | | | | |
| 3 | Interfacing 8 LEDs to 8255 <ol style="list-style-type: none"> i. Write ALP to alternately flash 8 LEDs ii. Write ALP to display binary count in UP sequence on the 8 LEDs | | | | | | | | |
| 4 | Interfacing 8 switches and 8 LEDs to 8255 <ol style="list-style-type: none"> i. Write ALP read the status of the switches and display on the LEDs. ii. Write ALP so that when the first switch is made ON all the LEDs should glow and when the second switch is made OFF all the LEDs should become off. | | | | | | | | |
| A2: | Experiments for 8031/8051/89C51 Any two experiments from 1, 2 and 3 | | | | | | | | |
| 1 | 8031/51/89C51 assembly language programming: Simple data manipulation programs. (8/16-bit addition, subtraction, multiplication, 8/16-bit data transfer, cubes of nos., to rotate a 32- bit number, finding greatest/smallest number from a block of data, decimal / hexadecimal counter) | | | | | | | | |
| 2 | Study of IN and OUT port of 8031/51/89C51 by Interfacing switches, and LEDs: Interface 4 LEDs to one of the microcontrollers PORT. Write ALP to display bit pattern on LED's. <table style="width: 100%; border: none;"> <tr> <td style="text-align: center;">(i) 1000</td> <td style="text-align: center;">(ii) 1100</td> </tr> <tr> <td style="text-align: center;">0100</td> <td style="text-align: center;">0110</td> </tr> <tr> <td style="text-align: center;">0010</td> <td style="text-align: center;">0011 continuously</td> </tr> <tr> <td style="text-align: center;">0001 continuously</td> <td></td> </tr> </table> | (i) 1000 | (ii) 1100 | 0100 | 0110 | 0010 | 0011 continuously | 0001 continuously | |
| (i) 1000 | (ii) 1100 | | | | | | | | |
| 0100 | 0110 | | | | | | | | |
| 0010 | 0011 continuously | | | | | | | | |
| 0001 continuously | | | | | | | | | |

| | |
|--|--|
| 3 | <p>Interface 4 switches and 4 LEDs to one of the microcontrollers PORT. Develop various types of logic to control LEDs depending on the status/states/conditions of the 4 switches.</p> <p>For example: Write ALP such that when all four switches are ON all four LEDs should glow. But when any two switches become OFF then all four LEDs should become OFF. And so on.....</p> |
| Group B: Python Programming Exercises | |
| B1 : | Perform minimum two experiment |
| 1 | Write a program to generate the Fibonacci series. |
| 2 | Write a program to generate if a three digit number entered is an Armstrong number or not |
| 3 | Write a function that reverses the user defined value. |
| 4 | Write a recursive function to print the factorial for a given number. |
| B2 : | Perform minimum one experiment |
| 1. | Write a function that takes a character (i.e. a string of length 1) and returns True if it is a vowel, False otherwise. |
| 2. | Define a function that computes the length of a given list or string. |
| B3: | Perform minimum two experiments |
| 1. | Write a program that takes two lists and returns True if they have at least one common member. |
| 2. | Write a Python program to print a specified list after removing the 0th, 2nd, 4th and 5th elements. |
| 3. | Write a Python program to clone or copy a list |

SEMESTER VI
COURSE CODE: USACCS601

Microcontrollers & Python Programming (Version 3.x for Python)

| | | |
|-----------------|---|------------|
| Unit- I: | 8051 microcontroller: Timer/Counters, Serial Communication, Interrupts | (15 lect.) |
| 1. | <p>Programming 8051 Timers, Counter Programming, Basics of Serial Communication, 8051 Connection to RS232, 8051 Serial Port Programming in assembly, 8051 Interrupts, Programming Timer Interrupts, Programming External Hardware Interrupts</p> <p>Ref. MMM - Ch 9: 9.1, 9.2, 10.1, 10.2, 10.3, 11.1, 11.2, 11.3</p> <p>Ref. MMM - The 8051 Microcontroller & Embedded Systems by M.A. Mazidi, J.G. Mazidi and R.D.Mckinlay, Second Edition, Pearson</p> <p>Ref. Intel's 8031/8051 Data sheet</p> | |
| Unit-II: | Dictionaries, Tuples, File & Exception Handling in Python | (15 lect.) |
| 1. | Dictionaries: Creating a Dictionary, Accessing Values in a dictionary, Updating Dictionary, Deleting Elements from Dictionary, Properties of Dictionary keys, Operations in Dictionary, Built-In Dictionary Functions, Built-in Dictionary Methods, in operator. | |
| 2. | Tuples: Tuples, Accessing values in Tuples, Tuple Assignment, Tuples as return values, Variable-length argument tuples, Basic tuples operations, Concatenation, Repetition, in Operator, Iteration, Built-in Tuple Functions | |
| 3. | Files: Text Files, The File Object Attributes, Directories | |
| 4. | Exceptions: Built-in Exceptions, Handling Exceptions, Exception with Arguments, User-defined Exceptions. | |

| | | |
|-------------------|--|------------|
| Unit- III: | Object Oriented Programming , Modules & Multithreading in Python | (15 lect.) |
| 1. | Classes and Objects: Overview of OOP (Object Oriented Programming), Class Definition, Creating Objects, Instances as Arguments, Instances as return values, Built-in Class Attributes, Inheritance, Method Overriding, Data Encapsulation, Data Hiding | |
| 2. | Multithreaded Programming: Thread Module, creating a thread, synchronizing threads, multithreaded priority queue | |
| 3. | Modules: Importing module, Creating and exploring modules, Math module, Random module, Time module | |
| Unit-IV: | GUI & Database in Python Creating the GUI Form and Adding Widgets: | (15 lect.) |
| 1. | Widgets: Tkinter module , Label , Buttons , Checkbutton, Radiobuttons, Text box, Canvas , Entry, Frame, Text, Menu, LabelFrame, Scrolled Text Widgets , Message boxes , Spinbox . Handling Standard attributes and Properties of Widgets. | |
| 2. | Layout Management: Designing GUI applications with proper Layout Management features. | |
| 3. | Look and Feel Customization: Enhancing Look and Feel of GUI using different appearances of widgets. | |
| 4. | Storing Data in Our MySQL Database via Our GUI: Connecting to a MySQL database from Python, Configuring the MySQL connection, Designing the Python GUI database, Using the INSERT command, Using the UPDATE command, Using the DELETE command, Storing and retrieving data from MySQL database. | |

| | |
|--------------------------------|--|
| References: For unit I: | |
| 1. | MMM - The 8051 Microcontroller & Embedded Systems by M.A. Mazidi, J.G. Mazidi and R.D.Mckinlay, Second Edition, Pearson |
| 2. | Intel's 8031/8051 Data sheet |

| References: For units II, III and IV: | |
|--|--|
| 1. | Official Python Web site : https://www.python.org/ |
| 2. | AD : Think Python by Allen Downey , 2 nd Edition |
| 3. | Paul Gries , Jenifer Campbell , Jason Montojo : Practical Programming , Introduction to Computer Science using Python 3 , Second Edition. |
| 4. | Burkhard Meier : Python GUI Programming Cookbook , Packt Publishing |
| 5. | Michael H. Goldwasser, David Letscher ,Object-oriented Programming in Python , Michael H. Goldwasser |

In addition to the reference books **internet web-sites & MOOC** can be used wherever necessary.

PRACTICALS (USACCS6P1)

| Group A: Microcontrollers Any two experiments from the following | |
|---|--|
| 1 | Study of TIMER0/1 in Mode 0. |
| 2 | Study of TIMER0/1 in Mode 1. |
| 3 | Study of TIMER0/1 in Mode 2. |
| 4 | Study of External Interrupts INT0 / INT1. |
| Group B: Python Programming Exercises | |
| B1 : | Perform minimum two experiments (one from Dictionary & the other from file handling). |
| 1. | Write a Python script to sort (ascending and descending) a dictionary by value. |
| 2. | Write a Python script to concatenate following dictionaries to create a new one. |
| 3. | Write a Python program to sum all the items in a dictionary. |
| 4. | Write a Python program to read an entire text file. |

| | |
|-------------|---|
| 5. | Write a Python program to append text to a file and display the text. |
| 6. | Write a Python program to read last n lines of a file. |
| B2 : | Perform minimum one experiment |
| 1. | Design a class that store the information of student and display the same |
| 2. | Implement the concept of inheritance using python |
| 3. | Write a program to implement exception handling. |
| B3: | Perform minimum two experiments (one from GUI and the other from Database). |
| 1. | Try to configure the widget with various options like: <code>bg="red"</code> , <code>family="times"</code> , <code>size=18</code> |
| 2. | Try to change the widget type and configuration options to experiment with other widget types like Message, Button, Entry, Checkbutton, Radiobutton, Scale etc. |
| 3. | Design a simple database application that stores the records and retrieve the same. |
| 4. | Design a database application to search the specified record from the database. |
| 5. | Design a database application to that allows the user to add, delete and modify the records. |