Course Name : **Bachelor of Science**

Discipline   : **Computer Science**

(Those who joined in 2018 and after)

COURSE SCHEME:

SELF LEARNING COURSES

Semester	Course Name	Credit	External Mark = Total Mark	Subject Code
V	Microcontrollers and Embedded System Development in C	5	100 = 100	U1CSSL51
V	Introduction to Python Programming	5	100 = 100	U1CSSL52
V	Software Testing	5	100 = 100	U1CSSL53
V	Internet of Things	5	100 = 100	U1CSSL54

SELF - LEARNING COURSE - 1
Microcontrollers and Embedded System Development in C

Credit: 5   **Subject Code: U1CSSL51**

Total marks 100 (EXTERNAL)

Study Objective: This topic will motivate students to explore on their own about microcontroller programming basics and to design embedded system based applications.

Prerequisite: Knowledge on C language and computer organization.

Unit 1: 8051 Microcontrollers and Internal architecture

Microcontrollers & embedded processors, 8051 family overview, Inside 8051,Program counter and ROM, 8051 flag bits and the PSW register, 8051 register banks and stack.

Unit 2: 8051 Programming in C

Data types, Time delay in 8051, I/O programming in 8051, Logic operations in 8051, Data conversion programs in 8051, Accessing ROM and Data serialization using 8051.

Unit 3: Timer programming in C

8051 timers, programming timers in C, programming counters in C

Unit 4: Serial port programming in C

Basics of serial communication, 8051 connection to RS232, Serial port programming in C

Unit 5: Interrupts programming in C

8051 interrupts, Timer interrupts, external hardware interrupts, serial communication interrupts and all interrupt programming in C

Text book:

The 8051 Microcontroller and Embedded Systems – Using Assembly and C, Second Edition, Prentice Hall. Authors: *Muhammad Ali Mazidi, Janice Gillispie Mazidi & Rolin D. McKinlay.*

Chapters: 1, 2, 7,9,10 & 11. (Excluding Assembly language programming)

SELF - LEARNING COURSE -2
INTRODUCTION TO PYTHON PROGRAMMING

Credit: 5   **Subject Code: U1CSSL52**

Total marks 100(EXTERNAL)

Learning Objectives:

1. To develop logic for Problem Solving.

2. To develop problem solving skills and their implementation through **Python (version: 2.7)**
3. To understand Open Source Software.
Prerequisite: Knowledge on Java and C++ Programming.

Unit I:
Getting Started: Introduction to Python- an interpreted high level language, interactive mode and script mode. **Variables, Expressions and Statements:** Values, Variables and keywords; Operators and Operands in Python: (Arithmetic, relational and logical operators), operator precedence, Expressions and Statements (Assignment statement); Taking input (using raw_input() and input()) and displaying output(print statement);
Putting Comments.

Unit II:
Functions: Importing Modules (entire module or selected objects), invoking built in functions, functions from math module (for e.g. ceil, floor, fabs, exp, log, log10, pow, sqrt, cos, sin, tan, degrees, radians), using random() and randint() functions of random module to generate random numbers, composition. **Defining functions,** invoking functions, passing parameters *(default parameter values, keyword arguments)*, scope of variables, void functions and functions returning values, flow of execution.

Unit III:
Conditional constructs and looping: if else statement While, For (range function), break, continue, else, pass, Nested loops, use of compound expression in conditional constructs and looping

Unit IV:
PROGRAMMING WITH PYTHON: Strings: Creating, initialising and accessing the elements; String operators: +, *, in, not in, range slice [n:m]; Comparing strings using relational operators; String functions & methods: len, capitalize, find, isalnum, isalpha, isdigit, lower, islower, isupper, upper, lstrip, rstrip, isspace, istitile, partition, replace, join, split, count, decode, encode, swapcase, Pattern Matching

Unit V:
Lists: Concept of mutable lists, creating, initializing and accessing the elements, traversing, appending, updating and deleting elements; List operations (joining, list slices); List functions & methods: len, insert, append, extend, sort, remove, reverse, pop **Dictionaries:** Concept of key-value pair, creating, initializing and accessing the elements in a dictionary, traversing, appending, updating and deleting elements. Dictionary functions & Methods: cmp, len, clear(), get(), has_key(), items(), keys(), update(), values(). **Tuples:** Immutable concept, creating, intialising and accessing the elements in a tuple; Tuple functions: cmp(), len(), max(), min(), tuple()

TEXTBOOKS:
1. Open Source: **Python (version: 2.7)**: https://www.python.org/about/gettingstarted/
2**.** A Byte of Python – Swaroop CH (A free book on Programming with Python).

SELF LEARNING COURSE - 3
SOFTWARE TESTING

Credit: 5   **Total Mark: 100**

Subject Code: U1CSSL53

UNIT 1:

Principles of Testing, Software development life cycle models: phases of software project, quality, quality assurance, quality control - testing, verification, validation - process model to represent different phases - Life cycle models (waterfall, prototyping and rapid application development, spiral and iterative, The V, modified V models).

UNIT 2:

White box testing: static testing, structural testing, challenges in white box testing. Black box testing: what is black box testing?, why?, when?, how to do black box testing?

UNIT 3:

Integration Testing: Integration testing as a type of testing – Integration testing as a phase of testing – scenario testing – Defect Bash. System and acceptance testing: why? – functional Vs non functional – functional system testing – non functional testing – acceptance testing – summary of testing.

UNIT 4:

Performance testing: factors governing – methodology – tools – process for performance testing. Regression testing: what is? - Types – when to do? – How to do ? – Best practices.

UNIT 5:

Test Metrics and Measurements: what are metrics and measurements? – why mwrtics in testing? – types of metrics – project metrics – progress metrics – productivity metrics – release metrics.

TEXT BOOK :

Software Testing Principles and practices , Srinivasan Desikan, Gopalaswamy ramesh, Pearson Publication, 2016.

SELF - LEARNING COURSE - 4
INTERNET OF THINGS

Credit : 5   **Total Marks : 100**

Subject Code: U1CSSL54

UNIT I: IoT Introduction – Introduction, Features, Advantages & Disadvantages, Embedded devices and system, IoT Eco system and IoT framework

UNIT II: Architecture and Domains

Architecture models, IoT Energy domain, Iot Biometric domain, Security Camera and Door Lock system, Smart home and city, Smart agriculture, Healthcare and transportation, transform Business

UNIT III: IoT Devices

Smart Objects, IoT devi es and IoT boards in market

UNIT IV: IoT platforms

IoT palatform, ThinkWorx platform

UNIT V: Communication Protocols

Data link, Network and Session layer

References :

1. https://www.javatpoint.com/iot-tutorial
2. Internet of Things - Architecture and Design Principles First Edition (English, Paperback, Raj Kamal) Mc Graw Hill publishers
3. Internet of Things with Arduino and Bolt Paperback by Ashwin Pajankar (Author)

VIRUDHUNAGAR HINDU NADARS' SENTHIKUMARA NADAR COLLEGE
(An Autonomous Institution Affiliated to Madurai Kamaraj University)
[Re-accredited with 'A' Grade by NAAC]
Virudhunagar – 626 001.