Savitribai Phule Pune University **Third Year of Computer Engineering (2019 Course)** **310248: Laboratory Practice I**	Home
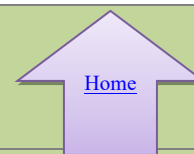
Teaching Scheme **Practical: 04 Hours/Week**	**Credit:02**	**Examination Scheme and Marks** **Term work: 25 Marks** **Practical: 25 Marks**

Companion Course: Systems Programming and Operating System (310243), Elective I(310245)

Course Objectives:

- To learn system programming tools
- To learn modern operating system
- To learn various techniques, tools, applications in IoT and Embedded Systems /Human Computer Interface/Distributed Systems/ Software Project Management

Course Outcomes:

On completion of the course, learners will be able to

- **Systems Programming and Operating System**
  - **CO1:** Implement language translators
  - **CO2**: Use tools like LEX and YACC
  - **CO3**: Implement internals and functionalities of Operating System
- **Internet of Things and Embedded Systems**
  - **CO4:** Design IoT and Embedded Systems based application
  - **CO5:** Develop smart applications using IoT
  - **CO6:** Develop IoT applications based on cloud environment

  OR

- **Human Computer Interface**
  - **CO4:** Implement the interactive designs for feasible data search and retrieval
  - **CO5:** Analyze the scope of HCI in various paradigms like ubiquitous computing, virtual Reality and ,multi-media, World wide web related environments
  - **CO6:** Analyze and identify user models, user support, socio-organizational issues, and stakeholder requirements of HCI systems

  OR

- **Distributed Systems**
  - **CO4:** Demonstrate knowledge of the core concepts and techniques in Distributed Systems
  - **CO5**: Apply the principles of state-of-the-Art Distributed Systems in real time applications
  - **CO6**: Design, build and test application programs on Distributed Systems

  OR

- **Software Project Management**
  - **CO4:** Apply Software Project Management tools
  - **CO5:** Implement software project planning and scheduling
  - **CO6:** Analyse staffing in software project

Guidelines for Instructor's Manual

The instructor's manual is to be developed as a reference and hands-on resource. It should include prologue (about University/program/ institute/ department/foreword/ preface), curriculum of the course, conduction and Assessment guidelines, topics under consideration, concept, objectives, outcomes, set of typical applications/assignments/ guidelines, and references.

Guidelines for Student's Laboratory Journal

The laboratory assignments are to be submitted by student in the form of journal. Journal consists of Certificate, table of contents, and handwritten write-up of each assignment (Title, Date of Completion, Objectives, Problem Statement, Software and Hardware requirements, Assessment grade/marks and assessor's sign, Theory- Concept in brief, algorithm, flowchart, test cases, Test Data Set(if applicable), mathematical model (if applicable), conclusion/analysis. Program codes with sample output of all performed assignments are to be submitted as softcopy. As a conscious effort and little contribution towards Green IT and environment awareness, attaching printed papers as part of write-ups and program listing to journal must be avoided. Use of DVD containing students programs maintained by Laboratory In-charge is highly encouraged. For reference one or two journals may be maintained with program prints in the Laboratory.

Guidelines for Laboratory /Term Work Assessment

Continuous assessment of laboratory work should be based on overall performance of Laboratory assignments by a student. Each Laboratory assignment assessment will assign grade/marks based on parameters, such as timely completion, performance, innovation, efficient codes, punctuality and

Guidelines for Practical Examination

Problem statements must be decided jointly by the internal examiner and external examiner. During practical assessment, maximum weightage should be given to satisfactory implementation of the problem statement. Relevant questions may be asked at the time of evaluation to test the student's understanding of the fundamentals, effective and efficient implementation. This will encourage, transparent evaluation and fair approach, and hence will not create any uncertainty or doubt in the minds of the students. So, adhering to these principles will consummate our team efforts to the promising start of student's academics.

Guidelines for Laboratory Conduction

The instructor is expected to frame the assignments by understanding the prerequisites, technological aspects, utility and recent trends related to the topic. The assignment framing policy need to address the average students and inclusive of an element to attract and promote the intelligent students. Use of open source software is encouraged. Based on the concepts learned. Instructor may also set one assignment or mini-project that is suitable to respective branch beyond the scope of syllabus. For the elective subjects students should form group of 3-4 students. The faculty coordinator will take care that all the assignment should be assigned to class and minimum two assignments are compulsory for each group.

Programming tools recommended: -

Human computer Interface-GUI in python

Internet of Things and Embedded System- Raspberry Pi/Arduino Programming; Arduino IDE/Python Interfacing. Other IoT devices

Software project management-MS project/Gantt Project/Primavera

Virtual Laboratory:

- http://cse18- iiith.vlabs.ac.in/Introduction.html?domain=Computer%20Scie nce
- http://vlabs.iitb.ac.in/vlabs-dev/labs/cglab/index.php

Suggested List of Laboratory Experiments/Assignments
Assignments from all Groups (A, B, C) are compulsory

Part I: Systems Programming and Operating System	
Sr. No.	**Group A (Any Two Assignments from Sr. No. 1 to 3)**
1.	Design suitable Data structures and implement Pass-I and Pass-II of a two-pass assembler for pseudo-machine. Implementation should consist of a few instructions from each category and few assembler directives. The output of Pass-I (intermediate code file and symbol table) should be input for Pass-II.

2.	Design suitable data structures and implement Pass-I and Pass-II of a two-pass macro-processor. The output of Pass-I (MNT, MDT and intermediate code file without any macro definitions) should be input for Pass-II.
3.	Write a program to create a Dynamic Link Library for any mathematical operation and writean application program to test it. (Java Native Interface / Use VB or VC++)
Group B(Any Two Assignments from Sr. No. 4 to 7)	
4.	Write a program to solve Classical Problems of Synchronization using Mutexand Semaphore.
5.	Write a program to simulate CPU Scheduling Algorithms: FCFS, SJF (Preemptive), Priority (Non-Preemptive) and Round Robin (Preemptive).
6.	Write a program to simulate Memory placement strategies – best fit, first fit, next fit and worst fit.
7.	Write a program to simulate Page replacement algorithm.

Part II : Elective I

Suggested List of Laboratory Experiments/Assignments
(Any Two assignments from each elective subject are compulsory andInstructor will take care that all the assignments should be covered among different batch students)

Internet of Things and Embedded Systems	
1.	Understanding the connectivity of Raspberry-Pi / Adriano with IR sensor. Write an application to detect obstacle and notify user using LEDs.
2.	Understanding the connectivity of Raspberry-Pi /Beagle board circuit with temperature sensor. Write an application to read the environment temperature. If temperature crosses a threshold value, generate alerts using LEDs.
3.	Understanding and connectivity of Raspberry-Pi /Beagle board with camera. Write an application to capture and store the image.
4.	Create a small dashboard application to be deployed on cloud. Different publisher devices can publish their information and interested application can subscribe.
Human Computer Interface	
1.	Design a paper prototype for selected Graphical User Interface.
2.	Implement GOMS (Goals, Operators, Methods and Selection rules) modeling technique to model user's behavior in given scenario.
3.	Design a User Interface in Python.
4.	To redesign existing Graphical User Interface with screen complexity.
Distributed System	
1.	Implementation of Inter-process communication using socket programming: implementing multithreaded echo server.
2.	Implementation of RPC Mechanism.
3.	Simulation of election algorithms (Ring and Bully).
4.	Implementation of Clock Synchronization: a) NTP b) Lamports clock.
Software Project Management	
1.	**Create Project Plan** ▪ Specify project name and start (or finish) date. ▪ Identify and define project tasks. ▪ Define duration for each project task. ▪ Define milestones in the plan ▪ Define dependency between tasks ▪ Define project calendar. ▪ Define project resources and specify resource type ▪ Assign resources against each task and baseline the project plan

2.	**Execute and Monitor Project Plan**
	▪ Update % Complete with current task status.
	▪ Review the status of each task.
	▪ Compare Planned vs Actual Status
	▪ Review the status of Critical Path
	▪ Review resources assignation status
3.	**Generate Dashboard and Reports**
	• **Dashboard**
	o Project Overview
	o Cost Overview
	o Upcoming Tasks
	• **Resource Reports**
	o Over-allocated Resources
	o Resource Overview
	• **Cost Reports**
	o Earned Value Report
	o Resource Cost Overview
	o Task Cost Overview
	• **Progress Reports**
	o Critical Tasks
	o Milestone Report
	o Slipping Tasks

@The CO-PO Mapping Matrix

PO/CO	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12
CO1	1	2	2	2	3	-	-	-	-	-	-	1
CO2	1	2	2	2	2	-	-	-	-	-	-	1
CO3	1	2	2	2	2	-	-	-	-	-	-	1
CO4	1	2	3	2	-	2	-	-	2	1	2	-
CO5	1	2	2	1	-	2	-	-	3	2	1	-
CO6	2	2	2	1	-	2	-	-	2	-	2	1