Systems Programming and Operating System

(Code - 310251)

Semester VI - Computer Engineering
(Savitribai Phule Pune University)

Strictly as per the New Credit System Syllabus (2015 Course)
Savitribai Phule Pune University w.e.f. academic year 2017-2018

Dilip Kumar Sultania
B.Tech.(hons.) Computer Science and Engineering,
I.I.T. , Kharagpur.

TechKnowledge™ Publications

PE36A Price ₹ 275/-

(Book Code : PE36A)

Systems Programming and Operating System (Code : 310251)
Dilip Kumar Sultania
Semester VI – Computer Engineering (Savitribai Phule Pune University)

First Edition	**:**	January 2011	**(For Pune University)**
First Edition	**:**	January 2018	
Second Revised Edition	**:**	January 2019	
Third Revised Edition	**:**	January 2020	**(TechKnowledge Publications)**

This edition is for sale in India, Bangladesh, Bhutan, Maldives, Nepal, Pakistan, Sri Lanka and designated countries in South-East Asia. Sale and purchase of this book outside of these countries is unauthorized by the publisher.

Printed at : 37/2, Ashtavinayak Industrial Estate,
Near Pari Company,
Narhe, Pune, Maharashtra State, India.
Pune – 411041

ISBN : 978-93-89684-96-4

Published by

TechKnowledge Publications

Head Office : B/5, First floor, Maniratna Complex,
Taware Colony, Aranyeshwar Corner,
Pune - 411 009. Maharashtra State, India
Ph : 91-20-24221234, 91-20-24225678.
Email : info@techknowledgebooks.com,
Website : www.techknowledgebooks.com

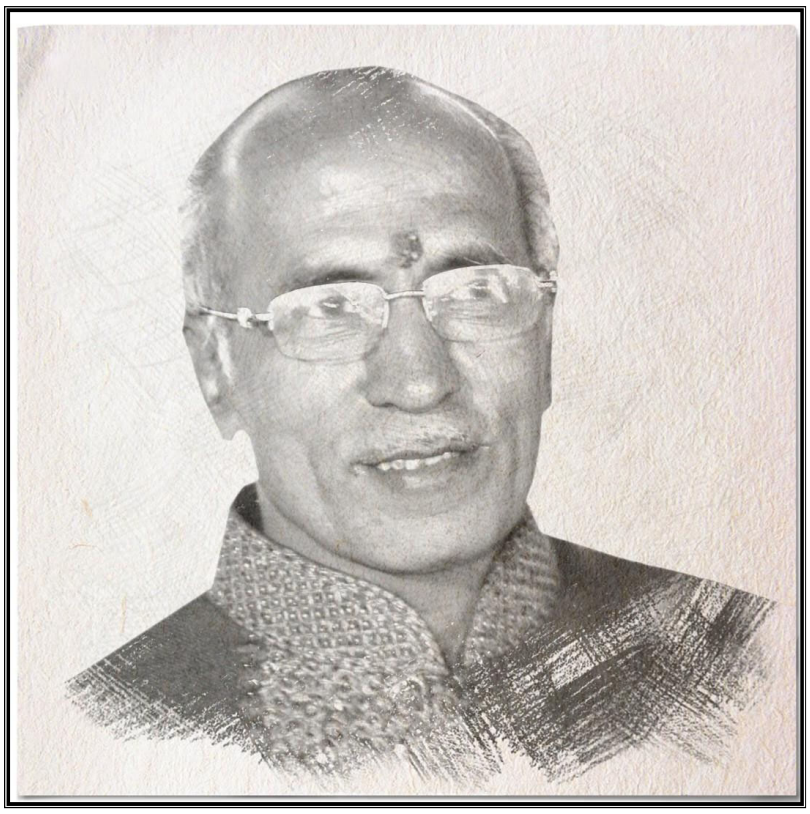
[310251] (FID : PE36) (Book Code : PE36A)

We dedicate this Publication soulfully and wholeheartedly,

in loving memory of our beloved founder director,

Late Shri. Pradeepji Lalchandji Lunawat,

who will always be an inspiration, a positive force and strong support behind us.



"My work is my prayer to God"

- Lt. Shri. Pradeepji L. Lunawat

Soulful Tribute and Gratitude for all Your Sacrifices, Hardwork and 40 years of Strong Vision…

Preface

My Dear Students,

I am extremely happy to come out with this book on **"Systems Programming and Operating System"** for you. The topics within the chapters have been arranged in a proper sequence to ensure smooth flow of the subject.

A large number of solved examples have also been included. Therefore, we are sure that this book will cater all your needs for this subject.

I present this book in the loving memory of **Late Shri. Pradeepji Lunawat,** our source of inspiration and a strong foundation of **"TechKnowledge Publications"**. He will always be remembered in our heart and motivate us to achieve our milestone.

I am thankful to the staff members of TechKnowledge Publications and others for their efforts to make this book as good as it is. I have jointly made every possible efforts to eliminate all the errors in this book. However if you find any, please let me know, because that will help me to improve further.

I am also thankful to my family members and friends for patience and encouragement.

- Dilip Kumar Sultania

❑❑❑

Syllabus

Savitribai Phule Pune University
Third Year of Computer Engineering (2015 Course)
310251 : Systems Programming and Operating System

Teaching Scheme	Credits	Examination Scheme
TH : 4 Hours/Week	04	In-Sem(Paper) : 30 Marks
		End-Sem(Paper) : 70 Marks

Prerequisite Courses: Fundamentals of Programming Languages(110011,110003), Data Structures (210243,210252)

Companion Course : Systems Programming and Operating System Lab (310257)

Course Objectives

- To understand basics of System Programming.
- To learn and understand data structures used in design of system software.
- To learn and understand basics of compilers and tools.
- To understand functions of operating system.
- To learn and understand process, resource and memory management.

Course Outcomes

On completion of the course, student will be able to–

- Analyze and synthesize system software
- Use tools like LEX & YACC.
- Implement operating system functions.

Course Contents

UNIT I : Introduction (09 Hours)

Introduction: Components of System Software: Text editors, Loaders, Assemblers, Macro processors, Compilers, Debuggers. Machine Structure, Machine language and Assembly Language. Assemblers: General design procedure, design of two pass assembler **(Refer Chapter 1)**

UNIT II : Macro Processor, Linker and Loader (09 Hours)

Macro Processor : Macro instructions, Features of macro facility, Design of two-pass, single pass and nested macro processor. Loaders: Loader schemes: Compile and go, General Loader Scheme, Absolute loaders, subroutine linkages, relocating loaders, direct linking loaders, overlay structure. Design of an absolute loader, Design of direct linking loader. Linkers: Relocation and linking concepts, Design of linker, self relocating programs, Static and dynamic link libraries, use of call back functions. Case Study: Loading phases using Java.

(Refer Chapters 2, 3 and 4)

(Book Code : PE36A)

UNIT III : Language Translator **(09 Hours)**

Role of lexical analysis - parsing & Token, patterns and Lexemes & Lexical Errors, regular definitions for the language constructs & strings, sequences, Comments & Transition diagram for recognition of tokens, reserved words and identifiers, examples Introduction to Compilers and Interpreters: General Model of Complier, Program interpretation, Comparison of compiler and Interpreter, Use of Interpreter and components of Interpreter. Case Study: Overview of LEX and YACC specification and features. **(Refer Chapter 5)**

UNIT IV : Operating Systems **(09 Hours)**

Operating Systems: Introduction to different types of operating Real Time Operating Systems, System Components, OS services, System structure- Layered Approach. Process Management: Process Concept- Process states, Process control block, Threads, Process Scheduling: Types of process schedulers, Types of scheduling: Preemptive, Non preemptive. Scheduling algorithms: FCFS, SJF, RR, Priority,

Deadlocks: Methods of handling deadlocks, Deadlock prevention, avoidance and detection, Recovery from deadlocks.

Case Study: Process Management in multi-cores OS. **(Refer Chapter 6)**

UNIT V : Memory Management **(09 Hours)**

Memory management: Review of Programming Model of Intel 80386, Contiguous and non-contiguous, Swapping, Paging, Segmentation, Segmentation with Paging. Virtual Memory: Background, Demand paging, Page replacement scheme- FIFO, LRU, Optimal, Thrashing. Case Study: Memory Management in multi-cores OS.

(Refer Chapter 7)

UNIT VI : I/O Management **(09 Hours)**

I/O Management: I/O Devices, Organization of I/O function, I/O Buffering, Disk Scheduling- Disk Scheduling policies like FIFO, LIFO, STTF, SCAN, C-SCAN. File Management: Concept, Access methods, Directory Structure, Protection, File System implementation, Directory Implementation, Allocation methods, Free Space management. Case Study: I/O and File Management in multi-cores OS

Case Study: Light weight and heavy weight OS: Linux, Tizen. **(Refer Chapter 8)**

❑❑❑

Savitribai Phule Pune University
Third Year of Computer Engineering (2015 Course)
310257 : Systems Programming and Operating System Lab

Teaching Scheme	Credits	Examination Scheme
PR : 4 Hours/Week	02	TW : 25 Marks
		PR : 50 Marks

Companion Course : Systems Programming and Operating System (310251)

Course Objectives

- To implement basic language translator by using various needed data structures
- To implement basic Macroprocessor
- To design and implement Dynamic Link Libraries
- To implement scheduling schemes

Course Outcomes

On completion of the course, student will be able to–

- Understand the internals of language translators
- Handle tools like LEX & YACC.
- Understand the Operating System internals and functionalities with implementation point of view

Guidelines for Instructor's Manual

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

Guidelines for Student Journal

The laboratory assignments are to be submitted by student in the form of journal. Journal consists of prologue, Certificate, table of contents, and **handwritten write-up** of each assignment (Title, Objectives, Problem Statement, Outcomes, software & Hardware requirements, Date of Completion, Assessment grade/marks and assessor's sign, Theory- Concept in brief, algorithm, flowchart, Design, test cases, 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 may be avoided. Use of DVD containing students programs maintained by lab In-charge is highly encouraged. For reference one or two journals may be maintained with program prints at Laboratory.

Guidelines for Assessment

Continuous assessment of laboratory work is done based on overall performance and lab assignments performance of student. Each lab assignment assessment will assign grade/marks based on parameters with appropriate weightage. Suggested parameters for overall assessment as well as each lab assignment assessment include- timely completion, performance, innovation, efficient codes, punctuality and neatness.

Guidelines for Practical Examination

Both internal and external examiners should jointly set problem statements. During practical assessment, the expert evaluator should give the maximum weightage to the satisfactory implementation of the problem statement. The supplementary and relevant questions may be asked at the time of evaluation to test the student's for advanced learning, understanding of the fundamentals, effective and efficient implementation.

(Book Code : PE36A)

So encouraging efforts, transparent evaluation and fair approach of the evaluator 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 the 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. The instructor may set multiple sets of assignments and distribute among batches of students. It is appreciated if the assignments are based on real world problems/applications. Encourage students for appropriate use of Hungarian notation, proper indentation and comments. Use of open source software is to be encouraged. In addition to these, instructor may assign one real life application in the form of a mini-project based on the concepts learned.

Set of suggested assignment list is provided in groups- A, B, C, D (All Compulsory)

Operating System recommended : 64-bit Open source Linux or its derivative

Programming tools recommended : Eclipse IDE

Suggested List of Laboratory Assignments	
Group A	
1.	Design suitable data structures and implement pass-I of a two-pass assembler for pseudo-machine in Java using object oriented feature. Implementation should consist of a few instructions from each category and few assembler directives.
2.	Implement Pass-II of two pass assembler for pseudo-machine in Java using object oriented features. The output of assignment-1 (intermediate file and symbol table) should be input for this assignment.
3.	Design suitable data structures and implement pass-I of a two-pass macro-processor using OOP features in Java.
4.	Write a Java program for pass-II of a two-pass macro-processor. The output of assignment-3 (MNT, MDT and file without any macro definitions) should be input for this assignment.
Group B	
1.	Write a program to create Dynamic Link Library for any mathematical operation and write an application program to test it. (Java Native Interface / Use VB or VC++).
2.	Write a program using Lex specifications to implement lexical analysis phase of compiler to generate tokens of subset of 'Java' program.
3.	Write a program using Lex specifications to implement lexical analysis phase of compiler to count no. of words, lines and characters of given input file.
4.	Write a program using YACC specifications to implement syntax analysis phase of compiler to validate type and syntax of variable declaration in Java.
5.	Write a program using YACC specifications to implement syntax analysis phase of compiler to recognize simple and compound sentences given in input file.
Group C	
1.	Write a Java program (using OOP features) to implement following scheduling algorithms: FCFS , SJF (Preemptive), Priority (Non-Preemptive) and Round Robin (Preemptive)
2.	Write a Java program to implement Banker's Algorithm
3.	Implement UNIX system calls like ps, fork, join, exec family, and wait for process management (use shell script/ Java/ C programming).
4.	Study assignment on process scheduling algorithms in Android and Tizen.
Group D	
1.	Write a Java Program (using OOP features) to implement paging simulation using 1. Least Recently Used (LRU) 2. Optimal algorithm

❑❑❑

UNIT I

UNIT II

UNIT IV

Syllabus : Operating Systems : Introduction to different types of operating system, Real Time Operating Systems, System Components, OS services, System structure- Layered Approach.

Process Management: Process Concept- Process states, Process control block, Threads, Process Scheduling: Types of process schedulers, Types of scheduling: Preemptive, Non preemptive. Scheduling algorithms: FCFS, SJF, RR, Priority,

Deadlocks : Methods of handling deadlocks, Deadlock prevention, avoidance and detection, Recovery from deadlocks.

Case Study : Process Management in multi-cores OS.

UNIT V

Chapter 7 : Memory Management 7-1 to 7-30

Syllabus : Memory management : Review of Programming Model of Intel 80386, Contiguous and non-contiguous, Swapping, Paging, Segmentation, Segmentation with Paging. Virtual Memory: Background, Demand paging, Page replacement scheme- FIFO, LRU, Optimal, Thrashing.

Case Study : Memory Management in multi-cores OS.

UNIT VI

Chapter 8 : Input and Output, File System 8-1 to 8-35

Syllabus : I/O Management : I/O Devices, Organization of I/O function, I/O Buffering, Disk Scheduling- Disk Scheduling policies like FIFO, LIFO, STTF, SCAN, C-SCAN.

File Management: Concept, Access methods, Directory Structure, Protection, File System implementation, Directory Implementation, Allocation methods, Free Space management.

Case Study: I/O and File Management in multi-cores OS

Case Study: Light weight and heavy weight OS: Linux, Tizen.

❑❑❑

List of Practicals

❑❑❑