Teaching Guidelines for

Concepts of Programming & Operating System

PG-DAC September 2021

Duration: 40 class room hours + 24 lab hours (**64 hours**)

Prerequisites: Knowledge of computer fundamentals

Evaluation: 100 marks (Concepts of Programming – 40 marks + Operating Systems – 60 marks)
Weightage: Theory exam – 40%, Lab exam – 40%, Internals – 20%

Concepts of Programming

Duration: 12 theory hours + 12 lab hours (**24 hours**)

Objective: To introduce the fundamental programming concepts in Java.

Evaluation: 40 marks (Theory exam: 12 + Lab exam: 20 + Internals: 8 marks)

Text Book:

- Core and Advanced Java Black Book / Dreamtech Press

References:

- Java The Complete Reference by Herbert Schildt / McGraw Hill
- Core Java : Fundamentals - Volume 1 Gary Cornell, Cay S. Horstmann/ Pearson
- Programming in Java by Sachin Malhotra, Saurabh Choudhary / Oxford University Press

---

(Note: Each Session is of 2 hours)

Session 1: Getting Started
Lecture:

- Setup development environment (JRE, JDK, eclipse)
- Writing your first Java program
- About main () method

Lab:
Write Java programs to:

- Print Hello World

Session 2: Variables & Methods
Lecture:

- Java Data Types, Primitives and Binary Literals
- Data type compatibility and casting of primitive data types
- Static variables and methods
- Accessing static variables and methods of different class
- Final variables

Lab:
Write Java programs to:

- Add two numbers/binary numbers/characters

- Calculate compound interest
- Calculate power of a number
- Swap two numbers
- Calculate area of rectangle
- Calculate area and circumference of circle using multiple classes
- Java program to find ASCII value of a character
- Print default values of primitive data type variables

Session 3: Operators
Lecture:
- Arithmetic Operator
- Relational Operator
- Logical Operator
- Unary Operator
- Ternary Operator
- Assignment Operator

Session 4: Conditional and Looping Statements
Lecture:
- If, else if, switch
- break & continue keyword
- for loop
- while loop
- do while loop
- Recursion

Lab: (4 hours)
Write Java programs to:
- Display prime numbers between 1 and 100 or 1 and n
- Swap two variables without using the third variable
- Find the factorial of a number
- Check if a number is palindrome or not
- Print Fibonacci series till n
- Add two integer variables in 5 different ways using functions and control statement
- Find square root of a number without sqrt method
- Check Armstrong number
- Calculate grades of students using their marks
- Use switch case, recursion, print patterns, etc.

Session 5: Objects
Lecture:
- Reference variables and methods
- Constructors (Default constructor, parameterised constructor)
- Static method v/s instance method
- Reference variable as instance member of the class
- String class

Lab:
- Build a class Employee which contains details about the employee and compile and run its instance.
- Build a class which has references to other classes. Instantiate these reference variables and invoke instance methods.

Session 6: Arrays
Lecture:

- Initializing an Array in Java
- Two dimensional array in java
- Java Variable Arguments explained
- Add, update, read array elements
- Sorting and searching in array
- Java String Array to String
- How to copy arrays in Java

Lab:
Write Java programs to:

- Calculate average of numbers using Array
- Reverse an array
- Sort an array in ascending order
- Convert char Array to String
- Add two Matrix using Multi-dimensional Arrays
- Sort strings in alphabetical order
- Find out the highest and second highest numbers in an array
- Concatenate two arrays

Concepts of Operating System

Duration: 28 class room hours + 12 lab hours (**40 hours**)

Objective: To introduce Operating System concepts with Linux environment, and to learn Shell Programming.

Evaluation: 60 marks (Theory exam: 28 + Lab exam: 20 + Internals: 12 marks)

Text Books:

- Operating Systems Principles by Abraham Silberschatz, Peter Galvin & Greg Gagne / Wiley
- Unix Concepts and Applications by Sumitabha Das / McGraw Hill

References:

- Modern operating Systems by Andrew Tanenbaum & Herbert Bos/ Pearson
- Principles of Operating Systems by Naresh Chauhan / Oxford University Press
- Beginning Linux Programming by Neil Matthew & Richard Stones / Wrox
- Operating System : A Design-Oriented Approach by Charles Crowley / McGraw Hill

---

Session 1:
Lecture:
Introduction to OS

- What is OS; How is it different from other application software; Why is it hardware dependent
- Different components of OS
- Basic computer organization required for OS
- Examples of well known OS including mobile OS, embedded system OS, Real Time OS, desktop OS server machine OS etc. ; How are these different from each other and why
- Functions of OS

- User and Kernel space and mode; Interrupts and system calls

(No Lab)

Session 2:
Lecture:
Introduction to Linux

- Working basics of file system;
- Commands associated with files/directories & other basic commands. Operators like redirection, pipe.
- What are file permissions and how to set them.
- Permissions (chmod, chown, etc); access control list; network commands (telenet, ftp, ssh, sftp, finger)
- System variables like – PS1, PS2 etc. How to set them

Shell Programming

- What is shell; What are different shells in Linux?
- Shell variables; Wildcard symbols
- Shell meta characters; Command line arguments; Read, Echo

Lab:

- Working with various OS commands
- Shell programs related to Session 2

Session 3:
Lecture:
Shell Programming

- Decision loops (if else, test, nested if else, case controls, while…until, for)
- Regular expressions; Arithmetic expressions
- More examples in Shell Programming

Lab:

- Shell Programs related to Session 3

Sessions 4, 5 & 6:
Lecture:
Processes

- What is process; preemptive and non-preemptive processes
- Process management; Process life cycle
- What are schedulers – Short term, Medium term and Long term.
- Process scheduling algorithms – FCFS, Shortest Job First, Priority, RR, Queue. Belady's Anomaly
- Examples associated with scheduling algorithms to find turnaround time to find the better performing scheduler.
- Process creation using fork; waitpid and exec system calls; Examples on process creation; Parent and child processes
- Orphan and zombie processes

Lab: (2 hours)

- Creating processes - parent and child processes
- Handling orphan and zombie processes.

Session 7:
Lecture:
Signals

- What are signals

- Generating and handling signals

Threads

- What are threads; user and kernel threads; how threads are different from processes
- Thread programming using pthread.

Lab:

- Assignment on signals
- Assignment on threads – Thread creation, thread synchronization

Sessions 8 & 9:

Lecture:

Memory management

- What are different types of memories; What is the need of Memory management
- Continuous and Dynamic allocation
- First Fit, Best Fit, worst Fit
- Compaction
- Internal and external fragmentation
- Segmentation – What is segmentation; Hardware requirement for segmentation; segmentation table and its interpretation
- Paging – What is paging; hardware required for paging; paging table; Translation look aside buffer
- Concept of dirty bit
- Shared pages and reentrant code
- Throttling

(No Lab)

Session 10:

Lecture:

Virtual Memory

- What is virtual memory
- Demand paging
- Page faults
- Page replacement algorithms

(No Lab)

Session 11:

Lecture:

Deadlock

- Necessary conditions of deadlock
- Deadlock prevention and avoidance
- Semaphore
- Mutex
- Producer consumer problem
- Dead-lock vs Starvation

Lab:

- Semaphore, Mutex

Sessions 12 & 13:

Lecture:

Inter process communication

- Message queues,
- Shared memory

- Pipes
- FIFO

Lab: (2 hours)

- IPC using shared memory
- IPC using Pipes
- IPC using FIFO

Session 14:

Lecture:

- File management
- Working with files and directories
- Mobile OS – Android

(No Lab)