EROTECH SOLUTIONS

Software Developer

Topic	C & DS under Linux(Ubuntu)
Number of Hours	120
Assignments	250
Classroom programs	150
Class hours	2 hours per day
Lab	Personal

M.Babu Krishna – Trainer

(21+ Years Experienced)

Module1: Linux Basics Duration:10 Hours

Introduction to Unix/Linux

- Unix and its history
- Introduction to Linux
- Login session
- Working with the Unix file system (Linux Directories)
- Linux Basic Commands (ls, pwd, touch, mkdir, rmdir, cp, mv, cat, rm)
- Handling files and directories (with met characters).
- Working with vi (visual editor)
- Linux documentation

File utilities

- Standard I/O,
- redirection and pipes
- Changing file access rights (users and permissions)
- Soft links and hard links
- Checking file integrity.

Linux Utilities

- Disk utilities
- Process utilities
- Text processing utilities Miscellaneous commands
- Compressing and archiving (backup and restore) utilities
- User management
- Time management and shutdown.

Module 2: C Language under Linux (Ubuntu) Duration:50 Hours

Programming Language overview and essentials

- Types of Languages
- IDE Environment (Editor, Compiler, Execution, Debugger)
- compilation steps (Compilation process tool chain (Preprocessor, Compiler, Assembler and Linker)
- Debugging with GDB (GNU Debugger) Sample C Programs
- Basic Syntax, Basic Elements (Character set, Tokens, Semicolons, comments, whitespaces, Keywords, Identifiers)
- input and output,
- Constants (Numeric, Character and String)
- Variables, Expression and Statement
- Data types (char, int, float and double)
- Sign and Size Qualifiers
- Numeric conversions (Binary, Octal, Decimal and Hexa Decimal)

Operators

- Arithmetic operators
- Assignment operators
- Relational operators
- Logical or boolean,
- Increment/Decrement,
- Conditional, Comma, size of () operators
- Storage classes (viz., auto, register, static and extern) - Scope and Life Time, Recursion, Enum
- Operations on Bits (Bit Set, Reset, Toggle and Checking Status)

Arrays

- Basic operations on Array
- Two dimensional arrays
- Multi-dimensional arrays
- Passing arrays as functional arguments
- Strings

Pointers

- Why do we need pointers?
- What are pointers?
- Using pointers
- Dereferencing pointer variables
- NULL pointers, Void pointers
- Call by value and Call by reference Pointer arithmetic
- Precedence of dereferencing and increment/decrement operators
- Pointer to pointer
- Array of pointers
- Pointer to an Array
- Function pointers
- Array of function pointers
- Passing Pointers to functions
- Return array/pointer from

Structures and Unions

- Defining a structure
- Accessing structure members
- Typedef
- Structure padding.
- Array of structures
- Arrays within structures
- Nested structures
- Pointer to structure,
- Pointer within structure
- Structures and functions
- Defining a union
- Accessing union members
- Bit fields

File I/O

- What is File I/O?
- Types of file handling - Low Level and High Level
- Types of files
- File operations (Opening files, Closing a file, Writing and Reading files)
- File I/O types (Character I/O, String I/O, formatted I/O and block I/O) Traversing within the file
- Error handling.

Preprocessors

- Header files
- Macro Substitution
- Nested Macros

- Bitwise and Implicit/Explicit Operators
- Operators Precedence and associativity.

Functions

- Defining a function
- Function declarations
- Calling a function
- Function arguments along with input and output types
- Returning from function

function

- Passing pointers as parameters
- Command line arguments

Memory Management

- Dynamic memory allocation
- Resizing and releasing memory.

- Issues with Macros
- Parameterized Macros
- Macros vs functions
- Overloading functions in "C" way
- String zing and token pasting operators
- Conditional compilation
- Debugging purposes
- Predefined macros.

Module3: Data Structures Duration:20 Hours

Introduction to Data Structures

- Why data structures?
- Efficient memory utilization and faster access
- Searching Techniques (Linear and Binary Search)
- Sorting Techniques (Bubble Sort)

Stacks and Queues

- Stacks using Arrays
- Queues using Arrays, Circular Queue
- Stacks using Linked Lists
- Queues using Linked Lists
- Infix to postfix conversion of expression
- Solving Arithmetic expression using stacks.

Linked Lists

- Operations on Linked Lists
- Creation, insertion, deletion, search, display
- count number of nodes reversing list etc
- Single Linked List
- Doubly Linked List
- Circular Linked List

Trees

- Introduction to trees
- Tree traversals (In-Order, Pre-order and Post-Order)
- Binary Search Tree (Creation, Display and Deletion).