Programming for Engineers: Fundamentals of Python, MATLAB, and C++ for Engineering Applications

Course general description:

In today's data-driven and technology-focused world, programming has become an essential skill for engineers. Whether it's automating repetitive tasks, analyzing large datasets, simulating engineering systems, or developing algorithms, coding is a powerful tool that enhances problem-solving capabilities. This course introduces engineers to the fundamentals of programming using three widely-used languages: Python, MATLAB, and C++. Participants will learn how to apply these languages to solve real-world engineering problems, from system modeling and simulation to product design and optimization.

Audience:

This course is designed for:

1. Engineers (mechanical, electrical, civil, chemical, etc.) seeking to enhance their programming skills.
2. Researchers and academics working on computational models and simulations.
3. Students in engineering disciplines looking to build a strong foundation in programming.
4. Professionals involved in data analysis, automation, and software integration in engineering workflows.

Course objectives:

By end of the course participants will gain:

1. Understand the fundamentals of programming concepts such as variables, loops, conditionals, functions, and data structures.
2. Gain proficiency in Python, MATLAB, and C++ for engineering applications.
3. Learn to write scripts and programs for numerical computation, data analysis, and system simulation.
4. Develop skills in debugging, testing, and optimizing code for efficiency.
5. Apply programming techniques to solve practical engineering problems through case studies and hands-on exercises.
6. Explore advanced topics such as algorithm development, automation, and integration with engineering tools.

Course duration:

5 days

Course location:

Dubai

Course contents:

Day 1: Introduction to Programming and Python Basics

- Fundamentals of Programming – Overview of programming languages, key concepts (syntax, variables, data types), and engineering applications.
- Python Basics – Installation, IDEs (Jupyter Notebook, PyCharm), basic scripting, and arithmetic operations.
- Engineering Libraries – Introduction to NumPy, Matplotlib, and SciPy for scientific computing.
- Challenges in Learning Programming – Group discussion on difficulties engineers face in coding.
- Case Study & Pretest – Write a Python script to calculate stress-strain relationships and assess baseline knowledge.

Day 2: MATLAB for Numerical Computation and Data Analysis

- MATLAB Basics – Workspace, scripts, functions, matrix operations, and visualization.
- Engineering Applications – Signal processing, Fourier transforms, and system simulation with Simulink.
- Data Analysis & Exporting – Analyzing experimental data and integrating results with other tools.
- Hands-On Exercises – Simulating a mechanical system and plotting results.
- Quiz & Discussion – Assess understanding of MATLAB fundamentals and applications.

Day 3: C++ for System Development and Performance Optimization

- C++ Basics – Variables, loops, conditionals, and object-oriented programming concepts.
- Performance Optimization – Memory management, pointers, and writing efficient high-performance code.
- Engineering Applications – Algorithm development for control systems and integration with embedded hardware.
- Case Study & Discussion – Optimize a heat exchanger design using C++ and compare programming languages.
- Quiz – Evaluate knowledge of C++ fundamentals and applications**.**

Day 4: Advanced Topics and Problem Solving

- Automation & Scripting – Automating tasks using Python, batch processing, and CAD/CAE tool integration.
- Algorithm Development & Debugging – Writing efficient algorithms, debugging techniques, and code optimization.
- Data Handling & Integration – Extracting and analyzing engineering data from multiple sources.
- Hands-On Simulation – Automate data analysis with Python and debug MATLAB/C++ programs.
- Group Activity – Collaborative debugging and code optimization session**.**

Day 5: Final Assessment and Future Trends

- Emerging Trends – AI, machine learning, cloud computing, and parallel processing in engineering.
- Career Pathways – Opportunities for engineers with programming skills and resources for continuous learning.
- Final Assessment – Comprehensive written test on course topics.
- Progress Measurement – Compare post-test results with pretest for knowledge gain.
- Feedback & Discussion – Key takeaways, participant questions, and reflections on course content**.**

Methodology:

- 50% lectures & concepts
- 10% Videos
- 15% Case studies
- 15% Exercises
- 10% Discussions

Course code: (TGRL005)