

## Course Outcomes

### Knowledge-Based outcome

#### 1. Understanding LabVIEW Environment

Students will be able to identify and describe the key elements of the LabVIEW development environment, including the front panel, block diagram, toolbar, and palettes.

#### 2. Data Flow Programming

Students will be able to explain the concept of data flow programming in LabVIEW and illustrate how data dependencies control the execution order of functions in a VI

#### 3. Control Structures

Students will be able to describe the purpose and functionality of control structures in LabVIEW, such as For Loops, While Loops, and Case Structures, and predict the outcomes of their use in a given scenario.

#### 4. Data Types and Structures

Students will be able to identify and differentiate between various LabVIEW data types (numeric, Boolean, string, etc.) and complex data structures (arrays, clusters) and discuss their appropriate applications.

#### 5. Debugging and Error Handling

Students will be able to explain the importance of debugging tools in LabVIEW, such as probes, breakpoints, and execution highlighting, and demonstrate how to utilize these tools to identify and resolve errors in a VI.

#### 6. File I/O Operations

Students will be able to describe the process of file I/O operations in LabVIEW, including reading from and writing to files, and compare the different file formats supported by LabVIEW.

#### 7. Creating and Managing SubVIs

Students will be able to explain the purpose of SubVIs in LabVIEW, describe the process of creating and managing SubVIs, and evaluate their use in modular programming."

### 8. Customizing User Interfaces

Students will be able to describe the techniques for customizing the user interface in LabVIEW, including the use of controls and indicators, and discuss how to design an effective and user-friendly front panel.

### 9. Advanced Data Handling

Students will be able to explain advanced data handling techniques in LabVIEW, such as the use of queues, notifiers, and shift registers, and illustrate how they can be applied to manage data flow and synchronization in complex applications.

### 10. LabVIEW Project Management

Students will be able to describe the key aspects of project management within LabVIEW, including the organization of VIs, libraries, and dependencies, and discuss strategies for maintaining large-scale projects.

### 11. Understanding LabVIEW Libraries and Toolkits

Students will be able to identify and describe the purpose of various LabVIEW libraries and toolkits, such as the Vision Development Module or the Control Design and Simulation Module, and discuss how these can be utilized in specialized applications.

## Skills-Based Outcome

### 1. Basic LabVIEW Skills

Students will be able to design and implement basic LabVIEW VIs that utilize loops, case structures, and data flow principles to solve engineering problems. They will be able to configure and control data acquisition hardware using LabVIEW, including setting sampling rates, triggering, and data logging.

### 2. Image Acquisition and Preprocessing

Students will be able to acquire and display live images from a camera using LabVIEW Vision Development Module, and perform basic image preprocessing techniques such as filtering, thresholding, and noise reduction. They will be able to implement image calibration techniques in LabVIEW to correct for distortions and ensure accurate measurements from vision systems.

### 3. Image Analysis and Processing

Students will be able to develop and configure LabVIEW VIs to perform edge detection, shape recognition, and object identification on images, optimizing parameters to enhance detection accuracy.

### 4. Advanced Vision Techniques

Students will be able to implement and fine-tune advanced vision techniques in LabVIEW, such as pattern matching, particle analysis, and color analysis, for complex image processing tasks. Students will be able to configure and optimize LabVIEW VIs to use machine learning algorithms for image classification and feature extraction, applying these techniques to real-world datasets.

### 5. Performance and Optimization

Students will be able to evaluate and optimize the performance of LabVIEW-based vision applications, ensuring real-time processing capabilities and minimizing latency. Students will be able to troubleshoot and refine LabVIEW vision systems, identifying and resolving issues related to image quality, processing speed, and system integration.

### 6. Project Management and Documentation

Students will be able to manage and document LabVIEW vision projects, including organizing code, maintaining version control, and creating comprehensive documentation for end users.

## Application-Based Outcomes

#### 1. Industrial Automation and Control

Students will be able to develop a LabVIEW-based application for monitoring and controlling an industrial process, utilizing vision systems to ensure product consistency and compliance with quality standards. Students will be able to design and implement a LabVIEW vision application that automates the inspection of circuit boards, identifying defects such as missing components, soldering issues, or alignment errors.

#### 2. Security and Surveillance Systems

Students will be able to create a LabVIEW application that integrates vision technology for security surveillance, capable of detecting and alerting based on motion detection and facial recognition.

Students will be able to develop a LabVIEW-based system that processes video feeds to detect and track moving objects in real-time, applicable in security or traffic monitoring scenarios.

### **3. Environmental Monitoring**

Students will be able to implement a vision system in LabVIEW that monitors environmental conditions, such as detecting the presence of specific contaminants or measuring the growth of organisms in a laboratory setting. They will be able to develop a LabVIEW application that utilizes vision technology for precision agriculture, analysing crop health and detecting pests or diseases through image processing techniques.

## Critical Thinking/Problem-Solving

### **1. Diagnosing and Troubleshooting**

Students will be able to diagnose and troubleshoot issues in complex LabVIEW VIs, identifying the root causes of errors or inefficiencies, and implementing solutions that optimize performance.

### **2. Innovative System Design**

Students will be able to design a novel LabVIEW solution for a complex engineering problem, integrating multiple subsystems and applying critical thinking to anticipate potential challenges and ensure system robustness. Students will be able to synthesize knowledge from different areas (e.g., signal processing, data visualization, hardware interfacing) to develop an innovative LabVIEW-based application that addresses a real-world challenge.

### **3. Complex Problem Analysis**

Students will be able to critically analyze a complex control system implemented in LabVIEW, identifying potential weaknesses or failure points, and proposing enhancements to improve system reliability. Students will be able to evaluate a LabVIEW-based vision system's performance in varying conditions, such as changes in lighting or object movement, and suggest improvements that enhance the system's robustness.

### 4. Decision-Making Under Constraints

Students will be able to analyze and prioritize competing design constraints (e.g., cost, performance, scalability) in a LabVIEW project, making informed decisions that balance these factors to meet project goals.

### 5. Evaluating and Justifying Solutions

Students will be able to construct well-supported arguments for the adoption of a particular LabVIEW-based approach in a team setting, using critical evaluation of alternative methods and clear articulation of their reasoning.

## Values/Attitudes Outcomes

### 1. Ethical Use of Technology

Students will demonstrate a commitment to ethical practices in the development and implementation of LabVIEW applications, ensuring that their work upholds the highest standards of integrity and respects user privacy and data security. They will value the importance of ethical considerations in automation and control systems, reflecting on the societal impact of their LabVIEW projects and striving to minimize any negative consequences.

### 2. Commitment to Quality

Students will exhibit a strong commitment to producing high-quality LabVIEW applications, rigorously testing their code, and ensuring the reliability and accuracy of their solutions. Students will embrace the importance of attention to detail in their LabVIEW work, recognizing that precision and thoroughness are crucial for the success of engineering projects.

### 3. Lifelong Learning and Adaptability

Students will commit to lifelong learning by staying current with the latest LabVIEW developments, actively seeking out new knowledge and skills, and adapting to changes in technology. Students will exhibit an open-minded attitude toward continuous improvement in their LabVIEW practice, welcoming feedback and using it to refine their skills and enhance the quality of their work.

#### **4. Responsibility and Accountability**

Students will demonstrate a sense of responsibility in managing LabVIEW projects, taking ownership of their work, and being accountable for meeting deadlines and project requirements. Students will value the importance of ethical responsibility in LabVIEW projects, ensuring that their solutions are not only technically sound but also socially and environmentally responsible.

#### **5. Assessment and Reflection**

Incorporate reflective assessments where students can evaluate their own values and attitudes, such as self-assessments, peer reviews, or reflective essays. Encourage them to think about how these attitudes influence their work and professional development.

#### **6. Integration with Course Content**

Align these objectives with the course content, ensuring that discussions, assignments, and group activities reinforce the importance of these values and attitudes in the context of LabVIEW and related fields.

