

Certified LabVIEW Associate Developer (CLAD) Certification and Exam Overview

Certification Overview

The National Instruments LabVIEW Certification Program consists of the following three certification levels:

- Certified LabVIEW Associate Developer (CLAD)
- Certified LabVIEW Developer (CLD)
- Certified LabVIEW Architect (CLA)

Each level is a prerequisite for the next level of certification.

A CLAD demonstrates a broad and complete understanding of the core features and functionality available in the LabVIEW Full Development System and possesses the ability to apply that knowledge to develop, debug, and maintain small LabVIEW modules. The typical experience level of a CLAD is approximately 6 to 9 months in the use of the LabVIEW Full Development System.

A CLD demonstrates experience in developing, debugging, and deploying and maintaining medium to large scale LabVIEW applications. A CLD is a professional with an approximate cumulative experience of 12 to 18 months developing medium to large applications in LabVIEW.

A CLA demonstrates mastery in architecting LabVIEW applications for a multi-developer environment. A CLA not only possesses the technical expertise and software development experience to break a project specification into manageable LabVIEW components but has the experience to see the project through by effectively utilizing project and configuration management tools. A CLA is a professional with an approximate cumulative experience of 24 months in developing medium to large applications in LabVIEW.



Note The CLAD certification is a prerequisite to taking the CLD exam.

The CLD certification is a prerequisite to taking the CLA exam.

There are no exceptions to this requirement for each exam.

Certified LabVIEW Associate Developer (CLAD) Certification and Exam Overview

Exam Overview

Product: LabVIEW Full Development System version 2010 for Windows. Refer to [LabVIEW Development Systems](#) comparison for details on the features available in the LabVIEW Full Development System.

Exam Duration: 1 hour

Number of Questions: 40

Style of Questions: Multiple-choice

Passing grade: 70%

The exam validates application knowledge and not the ability to recall menu steps or names of VIs and components.

The use of LabVIEW or any other external resources is prohibited during the exam. For assistance and wherever appropriate, screenshots from the *LabVIEW Help* are provided in the exam.

To maintain the integrity of the exam, you may not copy or reproduce any section of the exam. Failure to comply will result in failure. In areas where the exam is deployed as a paper based exam, detaching the binding staple will result in failure without evaluation.

Exam Logistics

United States and Europe: The CLAD exam can be taken at Pearson Vue test centers. The exam is computer-based and results are available immediately upon completion of the exam. Refer to www.pearsonvue.com/ni for more details and scheduling.

Asia: The exam is paper-based, for which the evaluations and results take about 4 weeks. Please contact your National Instruments local office for details and scheduling.

For general questions or comments, email: certification@ni.com.

**Certified LabVIEW Associate Developer (CLAD)
Certification and Exam Overview**

Exam Topics

The CLAD consists of 40 questions. Each exam consists of a specific number of questions from each category listed in the table below.

Exam Topics		Number of Questions
General	LabVIEW Programming Principles	3
	LabVIEW Environment	2
	Data Types	2
	Arrays and Clusters	4
	Error Handling	2
	Documentation	1
	Debugging	2
Structures	Loops	4
	Case Structures	1
	Sequence Structures	1
	Event Structures	2
Programming Tasks	File I/O	1
	Timing	2
	VI Server	2
	Synchronization and Communication	2
	Design Patterns	2
Front Panel	Charts and Graphs	2
	Mechanical Actions of Booleans	1
	Property Nodes	2
Variables	Local Variables	1
	Functional Global Variables	1
Total		40

**Certified LabVIEW Associate Developer (CLAD)
Certification and Exam Overview**

Exam Topics (Overview):

Topic	Subtopic
1. LabVIEW Programming Principles	a. Data Flow b. Parallelism
2. LabVIEW Environment	a. Virtual Instruments (VIs) b. Front Panel and Block Diagram c. Icon and Connector Pane d. Context Help Window
3. Data Types	a. Numeric, String, Boolean, Path, Enum b. Clusters c. Arrays d. Type Definitions e. Waveforms f. Timestamps g. Dynamic Data Type h. Data Representation i. Coercion j. Data Conversion and Manipulation
4. Arrays and Clusters	a. Array Functions b. Cluster Functions c. Function Polymorphism
5. Error Handling	a. Error Clusters b. Error Handling VIs and Functions c. Custom Error Codes d. Automatic/Manual Error Handling
6. Documentation	a. Importance b. Context Help
7. Debugging	a. Tools b. Techniques
8. Loops	a. Loop Components b. Auto-indexing c. Shift Registers d. Loop Behavior
9. Case Structures	a. Case Selector b. Tunnels c. Applications
10. Sequence Structures	a. Types b. Behavior c. Applications
11. Event Structures	a. Notify and Filter Events b. Applications

**Certified LabVIEW Associate Developer (CLAD)
Certification and Exam Overview**

12. File I/O	a. Functions and VIs b. Applications
13. Timing	a. Timing Functions b. Applications
14. VI Server	a. Class Hierarchy b. Applications
15. Data Synchronization and Communication	a. Notifiers b. Queues c. Semaphores d. Global Variables e. Applications
16. Design Patterns	a. State Machine b. Master/Slave c. Producer/Consumer (Data and Events) d. Applications
17. Charts and Graphs	a. Types b. Plotting Data
18. Mechanical Action of Booleans	See CLAD Topic Details
19. Property Nodes	See CLAD Topic Details
20. Local Variables	a. Behavior b. Applications
21. Functional Global Variables	a. Behavior b. Applications

**Certified LabVIEW Associate Developer (CLAD)
Certification and Exam Overview**

CLAD Topics Details

1. LabVIEW Programming Principles

a. Data Flow

- i. Define data flow
- ii. Identify the importance of data flow in LabVIEW
- iii. Identify programming practices that enforce data flow in the block diagram, VIs, and subVIs
- iv. Identify programming practices that break data flow
- v. Trace execution of code through a VI

b. Parallelism

- i. Define parallel execution
- ii. Identify parallel code structure
- iii. Identify programming caveats of parallelism
- iv. Define race conditions
- v. Identify race conditions in code
- vi. Identify indeterminate execution

2. LabVIEW Environment

a. Virtual Instruments (VIs)

i. Front Panel and Block Diagram

- 1. Identify the relationship between front panel objects and block diagram objects
- 2. Visually inspect and analyze front panels and block diagrams to describe functionality
- 3. Determine front panel results based on given block diagrams
- 4. Identify VI types that do not have block diagrams
- 5. Utilize front panel object properties and options for given applications

ii. Icon and Connector Pane

- 1. Identify the purpose of the icon and connector pane
- 2. Identify and distinguish between different connection types

b. Context Help Window

- i. Identify and define the three connector pane terminal types – Required, Recommended, and Optional
- ii. Determine the functionality of a VI or function, given its Context Help window

**Certified LabVIEW Associate Developer (CLAD)
Certification and Exam Overview**

3. Data Types and Data Structures

a. Numeric, String, Boolean, Path, Enum

- i. Identify the most appropriate data type for front panel and block diagram objects
- ii. Identify and describe functions associated with the following data types
 - 1. Numeric–Numeric, Conversion, Data Manipulation, and Comparison palettes
 - 2. String–String, String/Number Conversion, and String/Array/Path Conversion palettes
 - 3. Boolean–Boolean palette
 - 4. Path–Path functions on File I/O palette

b. Clusters

- i. Identify applications that would benefit from data grouping using clusters
- ii. Select and apply the Bundle, Unbundle, Bundle by Name, and Unbundle by Name functions
- iii. Determine the impact of reordering controls or indicators in a cluster

c. Arrays

- i. Select and apply functions on the Array palette
- ii. Identify techniques that cause memory usage issues
- iii. Identify techniques that minimize memory usage
- iv. Identify and describe applications that would benefit from proper array usage

d. Type Definitions

- i. Identify and describe the applications which would benefit from the use of a type definition or a strict type definition
- ii. Determine if a type definition or a strict type definition is needed to represent a data item

e. Waveforms

- i. Select and apply waveform data type to display data on graphs and charts
- ii. Select and apply the Build Waveform and Get Waveform Components functions for given applications

**Certified LabVIEW Associate Developer (CLAD)
Certification and Exam Overview**

f. Timestamps

- i. Describe the timestamp data type and how it applies to measurement data
- ii. Select and apply timestamp functions located on the Timing palette for given applications

g. Dynamic Data Type

- i. Identify use cases for dynamic data
- ii. Describe the functionality of the Convert from Dynamic Data Express VI
- iii. Identify which types of controls/indicators and inputs/outputs can use dynamic data

h. Data Representation

- i. Describe bit usage for different data representations
- ii. Change numeric representation of controls, indicators, and constants
- iii. Identify data representation range limitations and wrap-around with different types of integers
- iv. Identify native LabVIEW big-endianness

i. Coercion

- i. Select the most appropriate data type to limit coercion
- ii. Identify resulting data type and memory usage in heterogeneous numeric operations
- iii. Correctly select and apply functions from the Conversion palette

j. Data Conversion and Manipulation

- i. Define and apply principles of data conversion, manipulation, and typecasting
- ii. Identify and select functions used for converting between data types and numeric representations

4. Arrays and Clusters

a. Array Functions

- i. Identify functions from the Array palette
- ii. Determine outcome of given block diagrams using array functions
- iii. Select and apply functions to give desired execution behavior
- iv. Compare and select equivalent design alternatives

**Certified LabVIEW Associate Developer (CLAD)
Certification and Exam Overview**

b. Cluster Functions

- i. Identify functions from the Cluster, Class, & Variant palette pertaining to clusters
- ii. Determine outcome of given block diagrams using cluster functions
- iii. Select and apply cluster functions to give desired execution behavior

c. Function Polymorphism

- i. Define polymorphism
- ii. Identify benefits of polymorphism
- iii. Determine the output of data elements in VIs using polymorphic inputs

5. Error Handling

a. Error Clusters

- i. Define and identify the function of the components of the error cluster
- ii. Identify terminals that accept error clusters as inputs
- iii. Differentiate between errors and warnings

b. Error Handling VIs and Functions

- i. Identify VIs from the Dialog & User Interface palette that pertain to error handling
- ii. Identify the most appropriate locations to handle and report errors
- iii. Select a VI or function to complete specified error handling and reporting functionality

c. Custom Error Codes

- i. Identify the reserved range for custom error codes
- ii. Generate custom errors from VIs by manipulating error clusters

d. Automatic/Manual Error Handling

- i. Describe the effects automatic error handling
- ii. Design VIs that manage errors thoroughly and effectively
- iii. Given a block diagram, describe the execution behavior when errors occur

**Certified LabVIEW Associate Developer (CLAD)
Certification and Exam Overview**

6. Documentation

a. Importance

- i. Identify the importance of adding a description to VI Properties
- ii. Identify the importance of adding a tip strip

b. Context Help

- i. Determine which inputs are required for executing a VI
- ii. Describe how to document inputs and outputs of a VI in Context Help

7. Debugging

a. Tools

- i. Identify debugging tools—Highlight Execution, Breakpoints and Single-Stepping, Probes
- ii. Explain the function and proper use case for specific debugging tools

b. Techniques

- i. Given a situation, select the most appropriate debugging tool or strategy
- ii. Determine if an error occurs given a specific block diagram

8. While Loops and For Loops

a. Loop Components

- i. Identify loop components and describe their functions—Tunnels, Count Terminal, Conditional Terminal, Iteration Terminal, Shift Registers
- ii. Describe the behavior of loop components

b. Auto-Indexing

- i. Identify auto-indexing tunnels
- ii. Identify default indexing settings when creating new tunnels
- iii. Describe auto-indexing tunnels and determine the effects of using or not using auto-indexing tunnels

**Certified LabVIEW Associate Developer (CLAD)
Certification and Exam Overview**

c. Shift Registers

- i. Describe the appropriate use and initialization of shift registers as data storage elements
- ii. Determine the data values in shift registers after a set number of iteration or upon loop termination
- iii. Identify the behavior of initialize and uninitialized stacked shift registers
- iv. Identify Feedback Nodes and their use in loops

d. Loop Behavior

- i. Identify specific behavior of For Loops and While Loops
- ii. Select and apply the most suitable looping structure
- iii. Given a block diagram, determine the number of iterations a loop iterates
- iv. Identify use cases for the conditional terminal in For Loops
- v. Determine which loop terminals are required for code execution in various situations

9. Case Structures

a. Case selector

- i. Identify data types that are acceptable as inputs
- ii. Identify different case options for ranges of numeric values
- iii. Given a block diagram, determine which case executes

b. Tunnels

- i. Identify the different options for output tunnels
- ii. Identify pros and cons of each tunnel type

c. Applications

- i. Determine when a case structure should be used instead of other structures
- ii. Identify proper placement of controls and indicators with respect to case structures

**Certified LabVIEW Associate Developer (CLAD)
Certification and Exam Overview**

10. Sequence Structures

a. Types

- i. Flat sequence structures
- ii. Stacked sequence structures

b. Behavior

- i. Identify basic functionality of sequence structures
- ii. Determine results of a given block diagram containing sequence structures
- iii. Describe sequence structure behavior when errors occur
- iv. Describe the behavior of sequence locals in stacked sequence structures

c. Applications

- i. Identify pros and cons of stacked and flat sequence structures
- ii. Determine when a sequence structure is more appropriate than other structures

11. Event Structures

a. Notify and Filter Events

- i. Define filter events and notify events
- ii. Describe the differing behavior of filter and notify events
- iii. Identify filter and notify events on a block diagram
- iv. Apply Value (signaling) property nodes with event structures

b. Applications

- i. Identify the advantages of event-driven programming
- ii. Identify different ways an event may be generated
- iii. Given a block diagram, determine the execution results

12. File I/O

a. Functions and VIs

- i. Identify VIs and functions from the File I/O palette
- ii. Determine the outcome of given block diagrams using these functions
- iii. Identify pros and cons of high-level and low-level File I/O VIs

b. Applications

- i. Predict if an error occurs in a block diagram
- ii. Determine the number of bytes written by certain functions given a block diagram
- iii. Determine the most and least efficient methods for writing data to a file

**Certified LabVIEW Associate Developer (CLAD)
Certification and Exam Overview**

13. Timing

a. Timing Functions

- i. Identify and describe functions on the Timing palette
- ii. Describe the effect of rollover with the Tick Count function

b. Applications

- i. Given a scenario, select the most appropriate function
- ii. Select appropriate functions for decreasing CPU usage in a loop
- iii. Select appropriate functions for timing applications over long periods

14. VI Server

a. Class Hierarchy

- i. Describe method and property inheritance
- ii. Select appropriate references for interacting with controls and subVIs

b. Applications

- i. Identify appropriate use cases for property nodes and invoke nodes
- ii. Select appropriate property nodes and invoke nodes to call properties and methods
- iii. Differentiate between strictly and weakly typed control references
- iv. Describe interaction between calling VIs and subVIs using VI Server

15. Data Synchronization and Communication

a. Notifiers

- i. Identify and describe functions on the Notifier palette
- ii. Given a block diagram using notifiers, determine execution outcome

b. Queues

- i. Identify and describe functions on the Queue palette
- ii. Given a block diagram using queues, determine the execution outcome

c. Semaphores

- i. Describe the functionality of semaphores
- ii. Identify appropriate use cases for semaphores

**Certified LabVIEW Associate Developer (CLAD)
Certification and Exam Overview**

d. Global Variables

- i. Describe the behavior of global variables
- ii. Identify appropriate use cases for global variables

e. Applications

- i. Given design scenarios, select the best data synchronization mechanism
- ii. Describe the differing functionality between notifiers and queues

16. Design Patterns

a. State Machine

- i. Identify principal components of the state machine architecture
- ii. Identify mechanisms used for maintaining state information

b. Master/Slave

- i. Identify principal components of the master/slave architecture
- ii. Identify pros and cons of the master/slave design pattern
- iii. Describe inherent loop timing provided by notifiers

c. Producer/Consumer (Data and Events)

- i. Identify principal components of the producer/consumer design pattern
- ii. Identify pros and cons of the producer/consumer design pattern
- iii. Describe inherent loop timing provided by queues

d. Applications

- i. Given a programming task, select the best design pattern
- ii. Compare design patterns and identify pros and cons of each

17. Charts and Graphs

a. Types

- i. Distinguish between the different types of charts and graphs
- ii. Describe the buffering functionality of waveform charts
- iii. Identify which graphs support uneven X axis scales
- iv. Identify which types of charts and graphs support multiple axes

b. Plotting Data

- i. Identify the data types accepted by charts and graphs
- ii. Given a scenario, select the most appropriate chart or graph type

18. Mechanical Action of Booleans

- a. Describe the six different mechanical actions
- b. Identify appropriate use cases for each action
- c. Given a scenario and a block diagram, determine the execution outcome

**Certified LabVIEW Associate Developer (CLAD)
Certification and Exam Overview**

19. Property Nodes

- a. Define the execution order of Property Nodes
- b. Identify ideal use cases for Property Nodes
- c. Determine what happens if an error occurs during execution of a Property Node

20. Local Variables

a. Behavior

- i. Describe the behavior of local variables
- ii. Given a block diagram using local variables, determine the result
- iii. Identify possible race conditions

b. Applications

- i. Determine when it is appropriate to use local variables for communication
- ii. Debug block diagrams that use local variables inappropriately

21. Functional Global Variables

a. Behavior

- i. Describe the behavior of functional global variables
- ii. Identify the components and the data storage mechanism
- iii. Identify the need for non-reentrancy

b. Applications

- i. Describe the synchronization capability of functional global variables
- ii. Describe information hiding
- iii. Given a scenario, determine if a functional global variable is appropriate

Certified LabVIEW Associate Developer (CLAD) Certification and Exam Overview

CLAD Exam Preparation Resources

Use the following resources for exam preparation:

CLAD Preparation:

- [CLAD Preparation E-Kit](#) (includes preparation guides and sample exams)

CLAD Preparation Webcasts:

- [National Instruments CLAD Preparation Course \(Online\) webcast](#)
- [Most missed concepts on the CLAD exam](#)

Free Online LabVIEW Training & Tutorials

- [Online LabVIEW Graphical Programming Course](#) (hosted by Connexions)
- [LabVIEW Introduction Course – Three Hours](#)
- [LabVIEW Introduction Course – Six Hours](#)

National Instruments Instructor-led or Self-paced training courses:

- [LabVIEW Core 1](#)
- [LabVIEW Core 2](#)
- [LabVIEW Core 3](#)
- [LabVIEW Performance](#)

Other National Instruments Resources:

- [National Instruments Academic Web](#)
- [National Instruments Developer Zone](#)
- [National Instruments LabVIEW Zone](#)
- [National Instruments LabVIEW Support](#)