National Instruments Training and Certification

What Does It Take To Get Certified?

Steven Hoenig
NJ Business Unit Manager
Certified LabVIEW Architect
Certified Professional Instructor
Overview

• About Bloomy Controls
• National Instruments Training
  • Benefits of Training
  • Training Options
  • LabVIEW Training/Certification Track
• Certification – LabVIEW Track
  • Certified LabVIEW Associate Developer (CLAD)
  • Certified LabVIEW Developer (CLD)
  • Certified LabVIEW Architect (CLA)
About Bloomy Controls

Bloomy Controls is a full service integrator providing turnkey solutions, consulting, and training for Test and Measurement systems.

- Founded in 1991
- Windsor, CT; Marlborough, MA; Fort Lee, NJ
- National Instruments **Select** Alliance Partner
- Industry Leader in NI LabVIEW development
- International System Deployment
NI Certified Training Centers

- 2 Certified Training Centers
  - Windsor, CT; Marlborough, MA

- Qualified Instructors
  - 14 Certified Professional Instructors
  - 13 Certified LabVIEW Architects
    - *World Leader for CLA’s in one company*
  - 5 Certified LabVIEW Developers
  - 2 Certified TestStand Developers/Architects
  - 1 Certified CVI Developer

www.bloomy.com
Software Development

• Software Experience:
  • LabVIEW, LabVIEW RT
  • TestStand
  • LabWindows / CVI
  • C, C++, C#, .NET, Ladder Logic
  • Data Management (DIAadem, SQL, Oracle)

• Software Engineering Practices:
  • Mature development practices based on *The LabVIEW Style Book*
  • Source Code Control
  • Re-usable Code
  • Bug Tracking
TRAINING
NI Training Saves Time and Money

• In a worldwide survey, NI training customers reported:

  - 66% faster learning
  - 50% faster development
  - 43% less maintenance

...as a result of NI training

• Lexmark reported 170% ROI from LabVIEW Basics training

*Source: NI WW training customer survey, 2009, n = 640, from US, Canada, Germany, Spain, Poland, Mexico, Brazil, South America, Japan, Taiwan, ASEAN
How Do These Savings Affect Project Costs?

- Sample project:
  - 6 month project with a 3 year life
  - 1 engineer, 60% of time spent on project
  - Annual cost of an engineer: $100,000

<table>
<thead>
<tr>
<th>Total Phase Costs</th>
<th>Self-Training</th>
<th>NI Training</th>
</tr>
</thead>
<tbody>
<tr>
<td>Expected total labor cost of project - learning phase</td>
<td>$12,000</td>
<td>$7,378</td>
</tr>
<tr>
<td>Expected total labor cost of project - development phase</td>
<td>$28,800</td>
<td>$14,400</td>
</tr>
<tr>
<td>Expected total labor cost of project - maintenance phase</td>
<td>$4,800</td>
<td>$2,736</td>
</tr>
<tr>
<td>TOTAL PROJECT LABOR COST</td>
<td>$45,600</td>
<td>$24,514</td>
</tr>
<tr>
<td>TOTAL PROJECT LABOR COST SAVINGS with NI training</td>
<td></td>
<td>$21,086</td>
</tr>
<tr>
<td>TRAINING ROI</td>
<td></td>
<td>286%</td>
</tr>
</tbody>
</table>

Details at [ni.com/training](http://ni.com/training)
NI Training & Certification Program

Worldwide Reach

• >20,000 users trained annually worldwide

• 6,000 active certified users worldwide (and growing)

• Courses taught in >40 countries

• Online training available in Americas, Europe, Japan
# Standard NI Training Offerings

<table>
<thead>
<tr>
<th>Over 25 Courses</th>
<th>Four Training Formats</th>
<th>Three Purchasing Methods</th>
</tr>
</thead>
<tbody>
<tr>
<td>LabVIEW</td>
<td>Instructor-led</td>
<td>Individual courses</td>
</tr>
<tr>
<td>Data Acquisition</td>
<td>In a regional classroom</td>
<td>Training memberships</td>
</tr>
<tr>
<td>RF</td>
<td>Onsite at your facility</td>
<td>Training credits</td>
</tr>
<tr>
<td>Real-Time</td>
<td>Online</td>
<td></td>
</tr>
<tr>
<td>FPGA</td>
<td>Self-paced</td>
<td></td>
</tr>
<tr>
<td>Instrument Control</td>
<td></td>
<td></td>
</tr>
<tr>
<td>CompactRIO</td>
<td></td>
<td></td>
</tr>
<tr>
<td>And more...</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
NI Training Courses

• Software
  • LabVIEW
    • Core 1, 2, 3
    • Connectivity
    • Advanced Architectures
    • Managing Software Engineering
    • Real-Time
    • FPGA
    • Performance
    • Object-Oriented Programming
  • LabWindows/CVI
    • Basics 1 and 2
  • TestStand
    • 1 and 2
  • DIAdem
    • Basics and Advanced
  • Multisim & Ultiboard

• Hardware
  • RF
    • Fundamentals
    • Application Development
  • CompactRIO
  • Modular Instruments:
    • DMMs
    • Switches
    • High-Speed Digital I/O
  • Data Acquisition and Signal Conditioning
  • Instrument Control
  • Machine Vision
What Makes Up NI Courses?

- Lecture
- Demonstrations
- Quizzes
- Hands-On Programming Exercises
  - Individual
  - Team-based (some courses)
### Standard Purchasing Methods

<table>
<thead>
<tr>
<th>Individual course purchase</th>
<th>Training and Certification Membership</th>
<th>Training Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Specific date and location</td>
<td>• Unlimited courses and certification exams for an individual</td>
<td>• Similar to a gift card</td>
</tr>
<tr>
<td>• Specific individuals</td>
<td>• 6 month, 1 year, 2 year options available</td>
<td>• Flexibility to redeem for courses, certification exams, self-paced materials at a later date</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• 1 training credit ≈ ½ day of basic training</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• Expire in 1 year</td>
</tr>
</tbody>
</table>

www.bloomy.com
Custom Training Options

We can customize the training experience in three ways:

**Purchasing Mechanism**
(ex: Corporate Membership)

**Your Training Plan**

**Delivery**
(ex: online onsite)

**Content**
(ex: LabVIEW Core 1, 2, 3 in 1 week)
Training Needs Assessment

Benefits:
- Understand skill level of your employees
- Identify technical lead candidates
- Uncover knowledge or process gaps
- Reflect your unique project needs
- Inform you of relevant training options and investment necessary to reach goals
NI Certifications Align with Training

Courses

Begin Here

New User

LabVIEW Core 1
LabVIEW Core 2

Certifications

Certified LabVIEW Associate Developer Exam

Experienced User

LabVIEW Intermediate
LabVIEW Core 3
LabVIEW Intermediate II

Certified LabVIEW Developer Exam

Advanced User

Advanced Architectures for LabVIEW
Managing Software Engineering in LabVIEW

Certified LabVIEW Architect Exam

"Certification is an absolute must for anyone serious about calling himself a LabVIEW expert... At our organization, we require that every LabVIEW developer be on a professional path to become a Certified LabVIEW Architect."

James Kring, James Kring, Inc.
Certification Program Overview

Certified Architect
- LabVIEW
- TestStand

Certified Developer
- LabVIEW
- LabWindows CVI
- TestStand

Certified Associate Developer
- LabVIEW

Mastery
Experience
Foundation
LabVIEW Training & Certification Path

**LabVIEW Core 1 & 2**
- LabVIEW fundamentals
- Basic application creation using LabVIEW
- Basics of data acquisition & instrument control

**LabVIEW Core 3 & Connectivity**
- Modular application development
- Structured design and development practices
- Inter-application communication and connectivity techniques

**Advanced Architectures for LabVIEW**
- Large application design
- Advanced development techniques
- Implementing multi-developer projects

**LabVIEW Certified Associate Developer**
- Demonstrates knowledge of core features and functions
- Develops, debugs and maintains small LabVIEW modules

**LabVIEW Certified Developer**
- Demonstrates experience in developing scalable, readable and maintainable applications
- Develops, debugs, and deploys medium to large scale applications

**LabVIEW Certified Architect**
- Demonstrates mastery in architecting applications for multi-developer teams
- Develops requirements architectures, tools, components and best practices for team

www.bloomy.com
CERTIFICATION
CLAD Exam: Format & Style

- Multiple choice style – 40 questions
  - 37 questions have one answer
  - 3 questions have multiple answers
    - These questions will be clearly identified
- Time allocated: 1 hour
- Computer Based Test
  - Mark questions for review – can go back to review items after completion
  - Skip questions – can go back to skipped items
CLAD Exam Question Format & Style

• Question format
  ▪ Multiple choice, application based
  ▪ Graphical - Questions / answers may contain screen shots of VIs / functions

• Question style
  ▪ Given a scenario, determine the best function / method for solving the problem
  ▪ Given a block diagram, predict the outcome
  ▪ Given a block diagram and missing function(s), determine the best function(s) to complete the functionality
1. Review the exam topics in the CLAD Exam Preparation Guide
2. Identify any knowledge gap
3. Attend NI Training courses or utilize resources (at the end of the CLAD exam preparation guide) to close the knowledge gaps
4. Assess yourself - Take the Sample exams
   (http://zone.ni.com/devzone/cda/epd/p/id/5225)
5. Webcast - Understand the Most Missed Concepts within the CLAD Exam
CLAD Exam Topics

1. LabVIEW programming concepts
2. LabVIEW environment
3. Software constructs in LabVIEW
4. Programming VIs and functions
5. Data communication and synchronization VIs and functions
6. VI Server VIs and functions
7. Error handling VIs and functions
8. Design patterns
9. SubVI design
10. Debugging tools and techniques
11. VI design and documentation (style) practices
12. Memory, performance, and determinism
CLAD Exam Topic: Programming Concepts

- LabVIEW programming concepts:
  - Data flow
    - Identify programming practices that enforce data flow on the block diagram, in VIs, and SubVIs
    - Identify programming practices that break data flow
You develop a SubVI that only outputs a value and need to use this SubVI in a (calling) VI. Which of the following is the best way to enforce dataflow to control the execution of the SubVI?

a. Use the SubVI in a Sequence structure
b. Modify the SubVI to have dummy inputs that can be used from the calling VI
c. Modify the SubVI to have Error clusters that can be used from the calling VI
d. Modify the SubVI to have a global variable and use it from the calling VI
Which of the following does not conform to data flow programming paradigm?

a. Shift Registers  
b. Tunnels  
c. SubVIs  
d. Local Variables
CLAD Exam Topic: Software Constructs

- Software constructs in LabVIEW
  - Program control structures and data storage
    - Determine the data values in the shift registers after a set number of iterations occur or upon loop termination
What is the value in Shift Register Answer after the following code has executed?

a. 16
b. 24
c. 32
d. 10
Programming VIs and functions

- Determine the output or intermediate values of data elements in an application that utilizes VIs and functions
- Determine the most appropriate VI(s) or function(s) to complete a specified functionality
Which timing function (VI) is the best choice for timing control logic in applications that run for extended periods of time?

a. Tick Count (ms)

b. Wait (ms)

c. Get Date/Time In Seconds

d. Format Date/Time String
CLAD Exam Topic: Design Patterns

- Design patterns
  - Identify a design pattern, explain its pros and cons, and compare it with other design patterns
The following block diagram represents which common type of VI architecture?

a. Multiple Case Structure VI  
b. General VI  
c. State Machine VI  
d. Parallel Loop VI
CLAD Sample Question: Design Patterns

What is one disadvantage of using the State Machine VI architecture?

a. A State Machine can only traverse states in order
b. If two state changes occur at the same time, only the first state change will be handled and the second will be lost
c. The diagram becomes significantly larger when changing from a general architecture to a State Machine
d. State Machines cannot acquire data or use DAQ functions
• Debugging tools and techniques
  - Given an error situation, select the most appropriate method to debug the error
CLAD Sample Question: Debugging Techniques

In what instance would you use the Probe tool rather than Highlight Execution?

a. To see the flow of data
b. To see the value of a wire in real-time
c. To look into a SubVI, as the process is running
d. To slowdown the VI and show data values in wires
CLAD Exam: What It Takes?

- Understand the Environment
  - Review Training Material – not just slides
  - Review NI Developer Zone review material
  - Understand concepts

- Know What Questions to Expect
  - Review Online Tests
  - Review Training Quiz material

- Be sure to answer all questions
- Take practice exams – time yourself
- Program, Program, Program...
Next Step: CLD Exam

• CLD Exam Logistics
  • Process
  • Grading

• CLD Exam Objective and Theme

• CLD Exam Design
  • Design Patterns
  • Timing Methods
  • Development Style
  • Documentation
  • Error Handling
CLD Exam: Process

- A PC with LabVIEW is provided
- Application development specification includes the following:
  - Screen shot of Front Panel
  - Description of the controls and indicators
  - General requirements
  - Functional specifications of the application
- LabVIEW documentation, Express VIs and examples may be used
- No external resources permitted
- Time allowed for exam: 4 hours
## CLD Exam: Grading Criteria

<table>
<thead>
<tr>
<th>Grading Criteria</th>
<th>Points on Exam</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>Functionality</td>
<td>15</td>
<td>37.5%</td>
</tr>
<tr>
<td>Style</td>
<td>15</td>
<td>37.5%</td>
</tr>
<tr>
<td>Documentation</td>
<td>10</td>
<td>25%</td>
</tr>
<tr>
<td><strong>Total Points</strong></td>
<td><strong>40</strong></td>
<td><strong>100%</strong></td>
</tr>
</tbody>
</table>

Passing grade is 75% or higher
CLD Exam: Demonstrates

- Problem solving skills
- LabVIEW competency
- Modular, scalable, and maintainable application design experience
- Consistent documentation
- Moderate development (wiring) speed
- Debugging and testing
Program design should be scalable to adapt to additional steps
Program design should be scalable to adapt to additional steps.
• Exams are based on a sequencer type application
• Step order is configurable via an input file or user interface
• Timing is an essential aspect of the application
• Output or log file may be required
• Application must respond to user interface activity within 100 msec.
CLD Exam: Design
This template is for the Standard State Machine design pattern.

Each frame of this case structure executes code for its state and computes what state to transition to next.

States are represented as values of an enumeration. These enumerations are instances of a type definition so that states can be quickly added. To edit the type definition, right-click on an enumeration and select Open Type Def.
## CLD Exam Design Patterns

<table>
<thead>
<tr>
<th>Design Pattern</th>
<th>Advantage</th>
<th>Disadvantage</th>
</tr>
</thead>
<tbody>
<tr>
<td>State machine</td>
<td>• Handles sequence control</td>
<td>• Cannot handle storing of sequences</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• May not be responsive enough to user interface events</td>
</tr>
</tbody>
</table>
Design Pattern: Queue Based
UI Event Handler
<table>
<thead>
<tr>
<th>Design Pattern</th>
<th>Advantage</th>
<th>Disadvantage</th>
</tr>
</thead>
<tbody>
<tr>
<td>State machine</td>
<td>• Handles sequence control</td>
<td>• Cannot handle storing of sequences</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• May not be responsive enough to user interface events</td>
</tr>
<tr>
<td>Queue based user interface Event Handler</td>
<td>• Extends state machine to store sequences</td>
<td>• Does not allow intensive event or sequence processing</td>
</tr>
<tr>
<td></td>
<td>• Handles user interface events</td>
<td>• User interface events need to be controlled by limiting access to front panel controls</td>
</tr>
</tbody>
</table>
Design Pattern: Producer Consumer

This template is for the Producer/Consumer design pattern.

This loop is the producer loop.

Data (can be any type)

Queue Event

No Error

This loop is the consumer loop.

Releasing the queue stops the consumer loop(s).

New Item

status

Error?
# CLD Exam Design Patterns

<table>
<thead>
<tr>
<th>Design Pattern</th>
<th>Advantage</th>
<th>Disadvantage</th>
</tr>
</thead>
<tbody>
<tr>
<td>State machine</td>
<td>• Handles sequence control</td>
<td>• Cannot handle storing of sequences</td>
</tr>
<tr>
<td></td>
<td>• May not be responsive enough to user interface events</td>
<td></td>
</tr>
<tr>
<td>Queue based UI Event Handler</td>
<td>• Extends state machine to store sequences</td>
<td>• Does not allow intensive event or sequence processing</td>
</tr>
<tr>
<td></td>
<td>• Handles user interface events</td>
<td>• User interface events need to be controlled by limiting access to FP controls</td>
</tr>
<tr>
<td>Producer / Consumer with Events</td>
<td>• Responsive to user interface events</td>
<td>• Does not integrate non-user interface events well</td>
</tr>
<tr>
<td></td>
<td>• Allows intensive event and sequence processing</td>
<td></td>
</tr>
</tbody>
</table>
### CLD Exam: Timing Methods

<table>
<thead>
<tr>
<th>Timing Type</th>
<th>Timing Method</th>
<th>Applications</th>
</tr>
</thead>
<tbody>
<tr>
<td>Execution timing control</td>
<td>Wait (ms)</td>
<td>Regulate execution of state machine loop to allow other programs to run on the computer</td>
</tr>
<tr>
<td></td>
<td>Wait Until Next (ms) multiple</td>
<td></td>
</tr>
</tbody>
</table>

- Execution timing control allows us to regulate how quickly a loop executes on the processor.
Execution Timing Control

This template is for the Standard State Machine design pattern.
Each frame of this case structure executes code for its state and
computes what state to transition to next.
States are represented as values of an enumeration. These
enumerations are instances
of a type definition so that states can be quickly added. To edit the
type definition, right click on an enumeration and select Open
Type Def.
## CLD Exam: Timing Methods

<table>
<thead>
<tr>
<th>Timing Type</th>
<th>Timing Method</th>
<th>Applications</th>
</tr>
</thead>
<tbody>
<tr>
<td>Software timing control</td>
<td>Get Date/Time in sec.</td>
<td>Good general purpose timing method for timing a sequence operation.</td>
</tr>
<tr>
<td></td>
<td>Tick Count</td>
<td>Encapsulate in functional global or SubVI</td>
</tr>
<tr>
<td></td>
<td>Express Elapsed time VI</td>
<td>Measures Elapsed time with stop / reset control functions. Ready made solution!!</td>
</tr>
</tbody>
</table>

- Software timing control allows us to time a real-world operation to perform for a set time period.
Software Timing Control

Functional Global Using Get/Date Time In secs function
Software Timing Control – Using an Express VI

This is a subVI from the Car Wash sample exam

Elapsed Time
Express VI
• Avoid the use of local variables when you can use a wire
• Use property nodes to modify control attributes and control values but not indicator values
• Typedefine reused enums and data structures
• Close references if opened explicitly
• Avoid data coercion and default tunnels
• Avoid copies of code - Develop SubVI if code is reused
• Create readable block diagram
  • Avoid unnecessary bends, overlapping objects, wires
  • code should flow left to right, top to bottom
### CLD Exam: Documentation

<table>
<thead>
<tr>
<th>Documentation</th>
<th>Example</th>
</tr>
</thead>
<tbody>
<tr>
<td>Label wires to identify their use</td>
<td><img src="example.png" alt="Indicators" /></td>
</tr>
<tr>
<td>Label constants</td>
<td><img src="example.png" alt="Seconds" /></td>
</tr>
<tr>
<td>Description and tool tips for UI controls</td>
<td><img src="example.png" alt="Configure Timing" /></td>
</tr>
<tr>
<td>Block diagram comments</td>
<td><img src="example.png" alt="Block Diagram" /></td>
</tr>
<tr>
<td>VI / SubVI Properties » Documentation</td>
<td><img src="example.png" alt="Context Help" /></td>
</tr>
</tbody>
</table>

*Tip: Brevity!!*
CLD Exam: Error Handling

Wire error terminals

Stop VI on error when specified

Use Simple error handler

Check the Sensor value is > 20 psi

Successful Run

Run Interlock
Run Sequence
Pressure Sensor

20.00

Lockout

User Controls
CLD Exam: Preparation Tips

• Test yourself:
  – Time yourself through the sample exams
  – Evaluate your solutions using the Exam Preparation Guide
    and the evaluation criteria given at the end of the guide

• Do better:
  – Incorporate best practices in all your LabVIEW projects
  – Use the generic sequencer model to develop new scenarios
  – Improve your coding speed
  – Find optimal ways to solve problems in LabVIEW
CLD Exam: Exam Tips

• Take a few minutes to plan your application
  – Decide on the best main VI architecture for the application
  – Decide which modules / SubVIs you will need to create
  – Decide on the timing method
• Focus on getting most of the core functionality working
• Document the VI and SubVIs
• Clean up block diagram towards the end
• Pace yourself... Time passes quickly when you are having fun!!
CLD Exam: What It Takes?

• Understand the Requirements
  • Review NI Developer Zone review material
  • Sample exams are either fully representative or actually representative of what you will see on the exam
  • Functional Coding is only 37% of the exam!

• Know What to Expect
  • You will have to use a design pattern
  • You will have to use error handling
  • You will need to document – don’t wait for the end
  • You will only have 4 hours – do not overcomplicate

• Allocate time to design your application
• Take practice exams – time yourself
• Program, Program, Program...
Next Step: CLA Exam

- CLA Exam Logistics
  - Process
  - Grading
- CLA Exam Objective and Theme
- CLA Exam Architecture
  - Architectural Design
  - Architectural Coding
  - Documentation
  - Requirements
CLA Exam: Process

• A PC with LabVIEW 2009 is provided
• Detailed application specification includes the following:
  • List of project deliverables
  • Sketch of Front Panel User Interface
  • Description of the controls and indicators
  • Functional requirements of the application – by component
• No external resources permitted - LabVIEW help and design patterns are allowed
• Time allowed for exam: 4 hours
# CLA Exam: Grading Criteria

<table>
<thead>
<tr>
<th>Grading Criteria</th>
<th>Points on Exam</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>User interface and block diagram style</td>
<td>10</td>
<td>10%</td>
</tr>
<tr>
<td>Documentation</td>
<td>20</td>
<td>20%</td>
</tr>
<tr>
<td>Requirements coverage</td>
<td>30</td>
<td>30%</td>
</tr>
<tr>
<td>Architecture development</td>
<td>40</td>
<td>40%</td>
</tr>
<tr>
<td><strong>Total Points</strong></td>
<td><strong>100</strong></td>
<td><strong>100%</strong></td>
</tr>
</tbody>
</table>

Passing grade is 70 % or higher
CLA Exam: Demonstrates

- Mastery of architecting LabVIEW applications for a multi-developer environment
  - LabVIEW competency – VI development and design pattern usage
  - Software development expertise
  - Ability to break down a project specification into manageable component
- Thorough and intuitive documentation abilities
- Ability to effectively utilize project management tools to see a project through
CLA Exam: Beyond Coding

• Only 10% of the exam involves actual functional LabVIEW code development
• You will be presented with a project similar to that of the CLD Exam
• Rather than coding a solution to the project, you will now be architecting a solution so that other developers can code the solution for you
  • Review and understand a set of technical requirements
  • Interpret requirements into a maintainable LabVIEW architecture
  • Set up key building blocks for future development
  • Plan and track project progress of development team
CLA Exam: Develop an Architecture (40%)

- You will be required to answer essay type questions defining critical architectural decisions
  - Primary framework of project
  - Timing architectures
  - Design patterns, software architecture, user interfaces
  - Design decisions, guidelines
  - Error handling strategy
  - Project management decisions, tools
  - Team development strategies
  - How will you test your software

- The architectural decisions made in this phase will be implemented in the next
CLA Exam: Implement Architecture (10%)

- Develop a project hierarchy
- Develop a core “main” VI - no functional code
  - User interface
  - Primary architecture / design pattern
  - Primary data structures
  - Error handling structure
  - Event and timing structures
  - Operational with fully connected modules
- Develop shell modules/subVIs - no functional code
  - Inputs / Outputs and Icons
  - Primary API, architecture, and data structure
  - Error handling and communication
  - Templates / reuse / classes – standardize
CLA Exam: Developer Instructions (20%)

- Provide detailed instruction for “main” VI and functional modules
  - Your developers now need to know exactly how to complete the functionality of the module
  - Instructions should be specific and concise
- Follow standard LabVIEW development guidelines for documentation
  - All VI Descriptions must be completed
  - Use control documentation on Front Panel objects
  - Free label comments as appropriate on block diagrams
  - Data Structures must be labeled
CLA Exam: Track Requirements (30%)

• All technical requirements must be addressed in the code
• Unique requirement ID must be used
• Use standard parts of architecture documentation to cover requirements
  • VI Description
  • Control Documentation Property
  • Project or Library Documentation Property
  • Front Panel or Block Diagram comments (free labels)
• NI Requirements Gateway will be used to verify – be sure to use correct syntax and cover ALL items
CLA Exam: What It Takes?

- Understand the Requirements
  - Review NI Developer Zone review material
  - Sample exams are either fully representative or actually representative of what you will see on the exam
  - Functional Coding is only 10% of the exam!

- Know What to Expect
  - 4 hours go by very quickly – do not overcomplicate (mind the 80/20 rule)
  - Project deliverables don’t change – the application does
  - The ability to direct and manage are critical

- Read the whole exam material before diving in
- Prepare in advance how to handle each section
- Plan, Plan, Plan...
Why Train and Certify in LabVIEW

- LabVIEW is a software development platform
  - Offers a rich set of functions and syntax
  - Offers advanced development tools
  - Requires standard software development processes to enable successful designs
- All of the above are promoted through the training and certification process
- Invest in acquiring the necessary skills that will pay off in the long run
- Join a continually growing group of professional developers embracing the LabVIEW environment
- Prepare your company or group to take on the most challenging application development tasks
Steven Hoenig
NJ Business Unit Manager
Certified LabVIEW Architect | Certified Professional Instructor
Office: 201.944.9890 | Mobile: 201.240.8749
steven.hoenig@bloomy.com

More information and downloads:
www.bloomy.com

Increase Productivity. Improve Quality. Reduce Cost.