



Examen de Certificación para Desarrollador Asociado de LabVIEW

Folleto de Prueba

Nota: Durante el examen, esta PROHIBIDO el uso de computadora o cualquier material de referencia.

Instrucciones:

Si no recibe este examen con el sello de “Certificación NI” en el sobre, **NO ACEPTÉ** este examen. Regréselo inmediatamente al proveedor. Se le entregará uno nuevo.

- **Por favor, no retire la grapa en ninguna sección. Si cualquier parte del papel de este examen se encuentra faltante o separado al regresar a National Instruments, se considerará como examen reprobado.**
- Por favor, no haga preguntas al supervisor. Si usted cree que el enunciado de la pregunta no es claro, puede hacer una observación en la pregunta, y escribir sus razones por haber escogido la respuesta que crea que mejor responda a la pregunta.
- Este examen no puede ser retirado del área donde se realiza o ser reproducido en ningún formato. No puede retener ninguna porción del examen después de haberlo completado.

Detalles del examen:

- Tiempo estimado de realización: 1 hora
- Características del examen: Opción múltiple
- Número de preguntas: 40 preguntas
- Calificación aprobatoria: 70%

IMPORTANTE: Al haber completado el examen, colóquelo en el sobre que se le proveyó y **SELLE** el sobre. Entregue el sobre sellado a su proveedor.

Hoja de Respuestas:

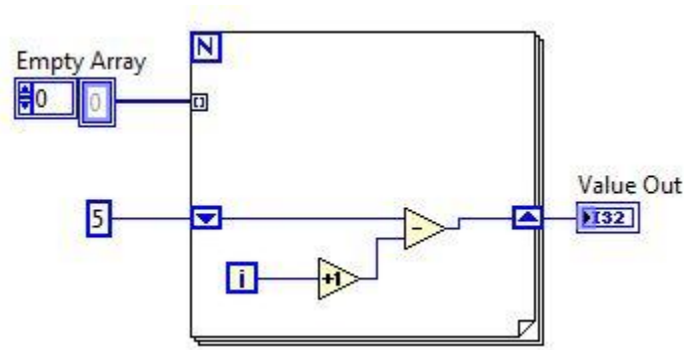
Para verificar sus respuestas rápidamente con las soluciones en la Página de Soluciones, proporcione sus respuestas en la Página de Respuestas. Desprenda esta página y proporcione sus respuestas. Esta página no está incluida en el examen CLAD actual; se incluye aquí con el propósito de practicar únicamente. La Página de Soluciones está al final del Ejemplo de Examen.

1. _____
2. _____
3. _____
4. _____
5. _____
6. _____
7. _____
8. _____
9. _____
10. _____
11. _____
12. _____
13. _____
14. _____
15. _____
16. _____
17. _____
18. _____
19. _____
20. _____
21. _____
22. _____
23. _____
24. _____
25. _____
26. _____
27. _____
28. _____
29. _____
30. _____
31. _____
32. _____
33. _____
34. _____
35. _____
36. _____
37. _____
38. _____
39. _____
40. _____

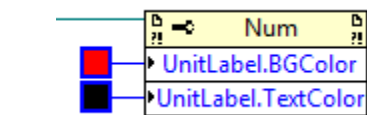
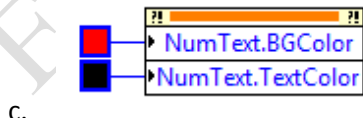
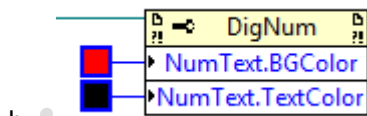
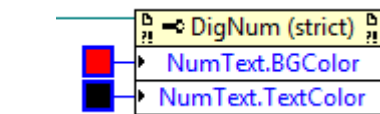
Elementos del Ejemplo de Examen:

1. Para implementar un diagrama de estado que permita la futura escalabilidad de la aplicación, la mejor opción para una estructura base es:
 - a. Estructura Sequence
 - b. Estructura Case
 - c. Formula node
 - d. Estructura orientada a objetos
2. Clickeando en el botón _____ permite eludir un nodo en el Diagrama de Bloques sin ir paso a paso a través del nodo.
 - a. Step Into
 - b. Step Over
 - c. Step Out
 - d. Step Through
3. ¿Cuál de los siguientes enunciados se aplica a los Nodos de Propiedad? **(Puede haber más de una respuesta correcta)**
 - a. Los Nodos de Propiedad permiten que los atributos de los objetos del panel frontal sean manipulados programáticamente.
 - b. Los Nodos de Propiedad pueden ser utilizados para actualizar los valores contenidos en los objetos del panel frontal.
 - c. Más de un Nodo de Propiedad puede ser usado para un solo objeto del panel frontal.
 - d. Los Nodos de Propiedad pueden ser utilizados para generar un evento "Value Change" programáticamente.
4. Tienes un panel frontal de control en un VI de alto nivel que debes controlar desde dentro de un subVI ¿Qué debes pasar al subVI?
 - a. Las propiedades de los controles
 - b. Los métodos de los controles
 - c. Las referencias de los controles
 - d. Los tipos de datos de los controles

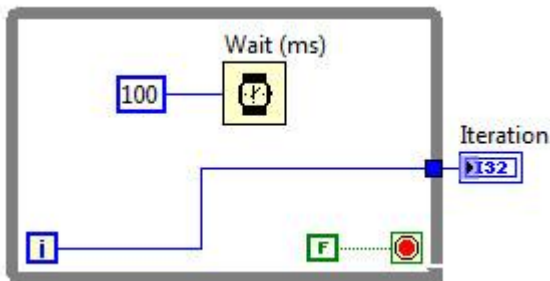
5. ¿Qué valor muestra el indicador **Value Out** después de que se ejecute el VI?



- a. 0
b. 4
c. 5
d. 6
6. Usted estableció un Breakpoint en el código contenido en el caso False de una estructura Case. ¿Se pausará la ejecución del VI si se está ejecutando el código del caso True de la estructura de Case?
- a. Si
b. No
7. ¿Qué nodo de propiedad cambia los atributos de color del texto de cualquier control digital numérico de un subVI?

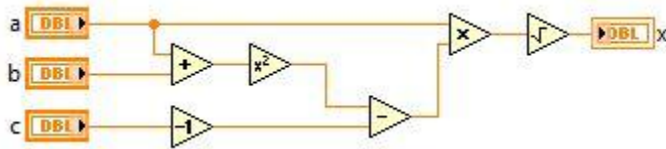


8. ¿Qué modo de actualización de un Waveform Chart se debe usar para mostrar datos en ejecución desplazándose continuamente de izquierda a derecha en el gráfico?
- Strip Chart
 - Scope Chart
 - Sweep Chart
 - Step Chart
9. ¿Cuál de los siguientes elementos no cumple con el paradigma de programación por flujo de datos?
- Registros de Corrimiento
 - Túneles
 - SubVIs
 - Variables locales
10. Si es posible, una estructura Sequence se debe reemplazar con un(a):
- Estructura Event
 - Bucle For
 - Máquina de estados
 - Bucle While
11. ¿Cuál de los siguientes enunciados es **verdadero** sobre la siguiente sección de código?



- El bucle se ejecutará una vez y el indicador Iteration tendrá un valor de uno
- El bucle se ejecutará una vez y el indicador Iteration tendrá un valor de cero
- El ciclo se ejecutará infinitamente, y el programa deberá ser abortado
- El bucle no se ejecutará y el indicador Iteration tendrá un valor de cero

12. ¿Qué ecuación es equivalente al código?



$$x = \sqrt{a \times ((a + b)^2 - (c - 1))}$$

a.

$$x = \sqrt{a \times ((c - 1) - (a + b)^2)}$$

b.

$$x = \sqrt{a \times ((a + b^2) - (c - 1))}$$

c.

$$x = \sqrt{a^2 \times ((a + b)^2 - (c - 1))}$$

d.

13. Un VI tiene habilitada la gestión automática de errores. Este VI llama a un SubVI que tiene deshabilitada dicha opción. El terminal Error Out del SubVI no está cableado en el VI que le llama. El SubVI devuelve un error al VI que le llama. ¿Se mostrará un cuadro de diálogo de error debido la gestión automática de errores?

- a. No, ya que el SubVI tiene el Automatic Error Handling deshabilitado
- b. Sí, ya que el SubVI tiene el Automatic Error Handling deshabilitado
- c. No, ya que el terminal Error Out del SubVI no está cableado en el VI que le llama
- d. Sí, ya que el terminal Error Out del SubVI no está cableado en el VI que le llama.

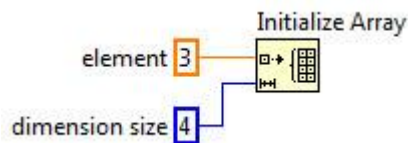
14. ¿Cuál de los siguientes enunciados es **falso**? (Puede haber más de una respuesta correcta)

- a. Un panel de conexión de un SubVI define dónde cablear entradas y salidas
- b. Los colores de terminales del panel de conexión de un SubVI concuerdan con los tipos de datos a las que se deben de conectar
- c. Se debe tener un icono/conector para usar una SubVI
- d. Un icono de SubVI debe ser editado desde la paleta de funciones

15. ¿Cuál de los siguientes no puede usarse para transferir datos?

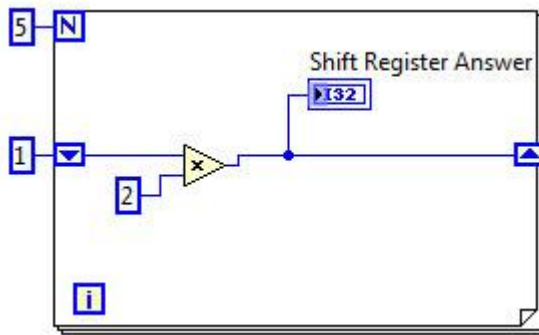
- a. Semáforos
- b. Colas
- c. Notificadores
- d. Variables locales

16. La lista de errores muestra todo lo que sigue **EXCEPTO**:
- a. Items con errores
 - b. Errores y advertencias
 - c. Detalles sobre los errores
 - d. Códigos de error
17. El auto-indexado en el borde de un bucle es una característica que permite que: **(Puede haber más de una respuesta correcta)**
- a. Los arrays sean traspasados automáticamente desde una iteración a la siguiente
 - b. Las funciones de array indexen automáticamente los elementos de un array
 - c. Las funciones que utilizan arrays los indexen automáticamente
 - d. Los arrays sean acumulados automáticamente en los bordes del bucle
18. ¿Cuál es el resultado de la función Inicializar Arreglo después de que el siguiente código se ha ejecutado?



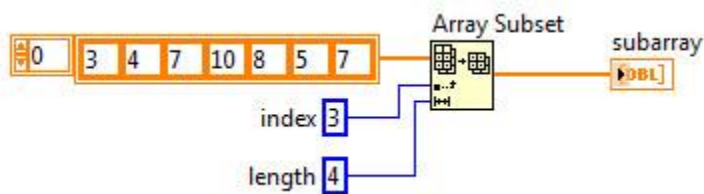
- a. Arreglo de una dimensión {3, 3, 3, 3}
 - b. Arreglo de una dimensión {4, 4, 4}
 - c. Arreglo de una dimensión {3, 4}
 - d. Arreglo de una dimensión {4, 3}
19. ¿Cuál de las siguientes opciones causará un evento sea capturado por la estructura de eventos de LabVIEW?
- a. Cambiar el valor de un control del Panel Frontal haciendo clic en el control
 - b. Cambiar la propiedad Value de un control del Panel Frontal a través de un nodo de propiedad
 - c. Cambiar el valor de un control del Panel Frontal a través de una referencia de control
 - d. Cambiar el valor de un control del Panel Frontal a través de una variable local
20. ¿Qué variable suele usarse comúnmente para eliminar la condición de carrera previniendo el acceso simultáneo a código o datos?
- a. Variable global funcional
 - b. Variable local
 - c. Variable global
 - d. Variable compartida

21. ¿Cuál es el valor en **Shift Register Answer** después del siguiente código haya sido ejecutado?



- a. 16
- b. 24
- c. 32
- d. 10

22. ¿Cuál es el resultado en **subarray** después de que el siguiente código se haya ejecutado?







- a. Arreglo de una dimensión {8, 5, 7}
- b. Arreglo de una dimensión {10, 8, 5}
- c. Arreglo de una dimensión {10, 8, 5, 7}
- d. Arreglo de una dimensión {7, 10, 8, 5}

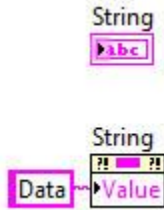
23. ¿En dónde puede ser documentada una VI de tal manera que la descripción aparezca en la ventana de "Show Context Help"?

- a. En la ventana de "VI Properties Documentation" (Propiedades de Documentación de VI)
- b. Escribiendo en la ventana Show Context Help
- c. Creando una etiqueta libre en el panel frontal
- d. Editando los archivos de ayuda de LabVIEW

24. ¿Cuál de las siguientes terminales de control controlan cuantas veces se ejecuta un ciclo For?

- a. 
- b. 
- c. 
- d. 

25. ¿Demuestra este fragmento de código un uso apropiado del Property Node?



- a. Si, porque actualizando el valor via Property Node usa menos memoria que escribiendo en el terminal
- b. Si, porque actualizando el valor via Property Node permite comprobar errores
- c. No, porque deberías siempre cablear los datos directamente a los indicadores cuando sea posible
- d. No, porque los Property Nodes deberían usarse sólo con tipos de datos más complejos

26. La función Wait Until Next ms Multiple:

- a. Comienza la temporización al inicio del programa y se completa en un múltiplo del temporizador del PC
- b. Comienza la temporización después de que el código del bucle se haya completado y expira cuando se cumple un múltiplo del tiempo indicado por el usuario
- c. Comienza la temporización después de que el código del bucle se haya completado y expira cuando se cumple el tiempo indicado por el usuario
- d. Comienza la temporización al inicio del programa y espera hasta que el código del bucle se haya completado

27. ¿Qué mecanismo de sincronización asegura que no haya pérdida de datos cuando una aplicación provee datos más rápidamente de lo que pueden ser procesados?

- a. Notificador
- b. Cola
- c. Semáforo
- d. Variable local

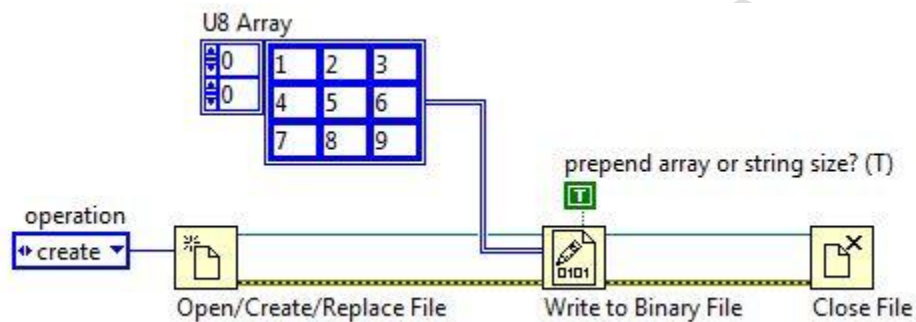
28. ¿Cuál de los siguientes enunciados NO es válido?

- a. Puedes crear un Cluster de Clusters.
- b. Puede crear un Array de Arrays.
- c. Puedes crear un Cluster de Arrays.
- d. Puede crear un Array de Clusters.

29. ¿Qué tipo de dato no es aceptado por el terminal selector de caso en una estructura case?

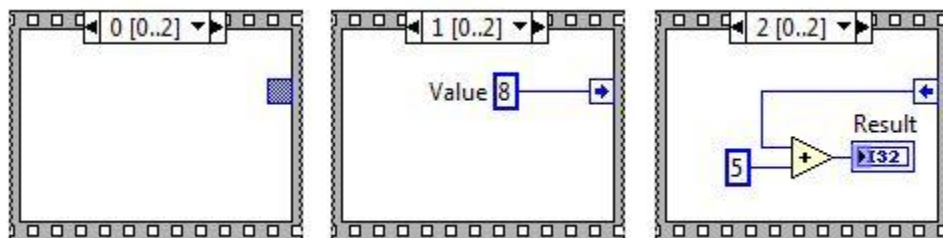
- a. Arrays
- b. Valores de tipo Enumerated
- c. Cadenas de caracteres
- d. Números enteros

30. ¿Cuántos bytes escribe LabVIEW en el archivo cuando se ejecuta el código?



- a. 9 Bytes
- b. 11 Bytes
- c. 13 Bytes
- d. 17 Bytes

31. ¿Por qué el terminal de secuencia local se muestra como "no asignado" en el Frame 0 de una estructura secuencial apilada?



- a. El desarrollador a elegido no cablear el valor a ningún terminal en este marco
- b. El valor sólo está disponible para marcos después del marco 1

- c. El tipo de dato del terminal es incompatible con el tipo de dato de Value
- d. El desarrollador a deshabilitado el terminal

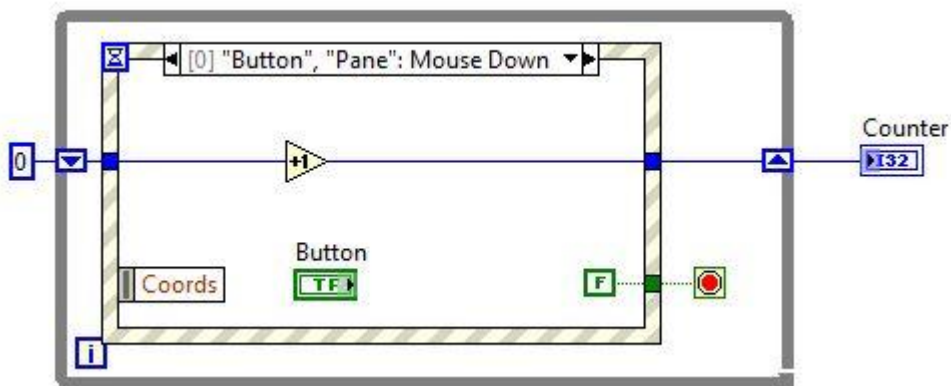
32. La función Wait puede ser añadida a los bucles While:

- a. Para liberar memoria disponible.
- b. Para asignar memoria usada por la CPU.
- c. Para permitir al tiempo de procesador completar otras tareas.
- d. Para reservar el procesador en el que se está ejecutando el código.

33. El tipo de dato de un Ring puede ser cualquiera de los tipos de dato permitidos, mientras que la representación numérica del tipo de dato de un Enum puede ser solamente un tipo de dato sin signo.

- a. Verdadero
- b. Falso

34. Cuando el usuario clickea el control **Botón**, ¿cuántas veces se llama a la función **Incrementar**?

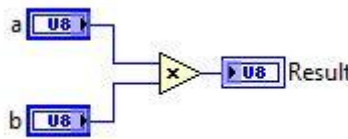


- a. 0
- b. 1
- c. 2
- d. 3

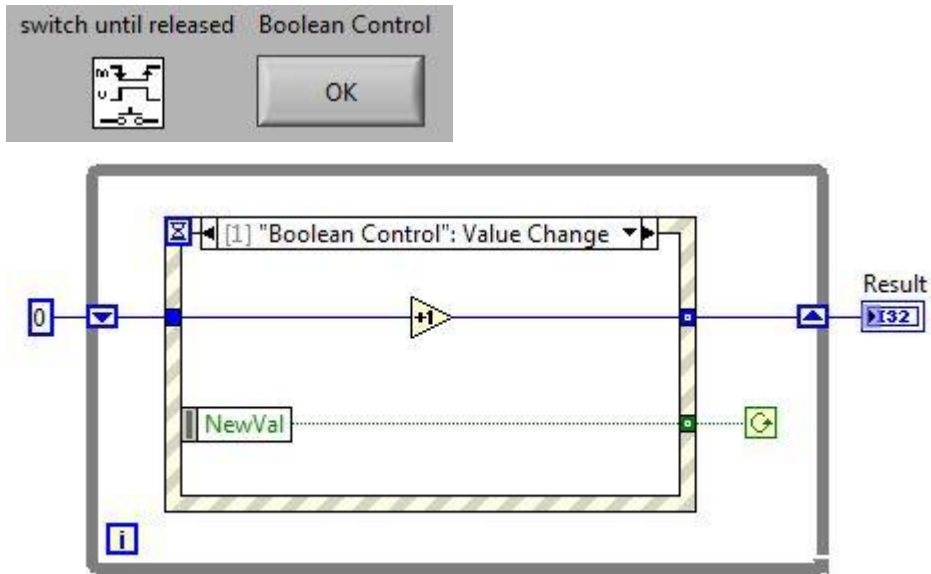
35. ¿Porqué LabVIEW es un lenguaje de programación de flujo de datos?

- a. Se utiliza iconos en lugar de texto para crear aplicaciones
- b. El flujo de datos a través de los nodos en el diagrama de bloques determina la ejecución
- c. No tiene una función Goto para controlar la ejecución
- d. La localización de la memoria en la que se guardan los datos cambia en cada iteración

36. Si el valor de **a** es 2 y **b** es 128, ¿qué valor mostrará el indicador **Resultado** después de que se ejecute el VI?



- a. -128
 - b. 0
 - c. 255
 - d. 256
37. ¿Qué VI se utiliza normalmente para terminar de cablear el Cluster de Error y mostrar cualquier mensaje de error?
- a. Merge Errors
 - b. One Button Dialog/Two Button Dialog
 - c. Generate Front Panel Activity
 - d. Simple Error Handler
38. ¿Cual de los siguientes métodos creará un gráfico XY Graph con dos líneas (plots)?
- a. Crear un cluster único a partir de dos arrays de valores X y dos arrays de valores Y en un formato X,Y,X,Y.
 - b. Agrupando juntos los arrays X e Y por parejas en dos clusters y luego crear un array de estos dos clusters.
 - c. Agrupando juntos los arrays X e Y por parejas en dos clusters y luego crear un cluster de estos dos clusters.
 - d. Creando un solo array de cada uno de los arrays de X e Y en un patrón X,Y,X,Y.
39. Cuando el VI se ejecuta, el usuario presiona y luego suelta el control **Boolean** con la acción mecánica de "Switch Until Released". Asumiendo que el valor inicial del control **Boolean** es FALSE, ¿qué valor mostrará el indicador **Result** después de la ejecución?



- a. 0
- b. 1
- c. 2
- d. 3

40. ¿Cuál de los siguientes es el mejor método para actualizar un indicador en el panel frontal?

- a. Utilice un Valor de nodo de propiedad
- b. Conecte directamente a la terminal del indicador
- c. Utilice una variable local
- d. Utilice una variable global funcional

Página de Resultados:

A continuación se presentan las respuestas y links a recursos adicionales para el Ejemplo de Examen de CLAD. Todas las referencias listadas son artículos de *LabVIEW Help*. Para verificar rápidamente sus respuestas, proporciónelas en la Página de Respuestas, despegue la Página de Respuestas y compárela con la Página de Resultados. Esta Página de Resultados no está incluida en el examen CLAD actual; está incluido aquí con el propósito de practicar únicamente.

1. **Respuestas correctas:** B

Tema: Patrones de diseño

Justificación: En LabVIEW, una máquina de estados simple consiste en un Bucle While que contiene una estructura Case. La estructura Case permite decidir programáticamente cual de los casos ejecutar. También es escalable ya que permite añadir nuevos casos fácilmente a la estructura Case.

Referencias: *Creando VIs desde Plantillas, Estructura Case*

2. **Respuesta correcta:** B

Tema: Depurando

Justificación: Clickando el botón Step Into provoca que LabVIEW abra el nodo o subVI. El botón Step Out se utiliza para volver de un subVI al VI principal durante la ejecución paso a paso (single-stepping). No existe el botón Step Through. Step Over provee la funcionalidad declarada en la pregunta, por lo tanto es la respuesta correcta.

Referencias: *Single-Stepping a través de un VI*

3. **Respuestas correctas:** A, B, C, D

Tema: Nodos de Propiedad

Justificación: Todos los enunciados son correctos para Property Nodes.

Referencias: *Nodos de Propiedad*

4. **Respuestas correctas:** C

Tema: VI Server

Justificación: La única opción de respuesta que se puede pasar llamando desde un VI a un subVI es la referencia de control. La referencia puede ser usada entonces con Property Nodes e Invoke Nodes para llamar a propiedades y métodos, respectivamente. El tipo de dato es una propiedad del control.

Referencias: *Controlando Objetos del Panel Frontal Programáticamente desde un SubVI, referencia de servidor de VI*

5. **Respuestas correctas:** C

Tema: Bucles

Justificación: Un array vacío está cableado al bucle For usando un tunel de auto-indexado. Esto causa que el bucle For itere una vez por cada elemento del array, que, en este caso, es cero. Sin embargo, el valor 5 está escrito en el registro de desplazamiento antes de la ejecución del bucle, y desde que el bucle itera cero veces, el mismo valor de 5 se presenta a la salida del registro de desplazamiento.

Referencias: *Bucle For, Usando Registros de Desplazamiento para Recordar Valores en las Iteraciones*

6. **Respuestas correctas:** B

Tema: Depurando

Justificación: El código que contiene breakpoints se ejecuta normalmente hasta que un dato se pasa por un cable que contenga un breakpoint. Entonces, LabVIEW suspende la ejecución. Para que el breakpoint se active, la sección de código que lo contiene debe ser llamada. Con una estructura Case, solo un caso se ejecutará cada vez que se la llame. Como, en este caso, el caso que contenía el breakpoint no fue llamado, la ejecución no se pausó.

Referencias: *Manejando Breakpoints*

7. **Respuestas correctas:** B

Tema: VI Server

Justificación: Los property nodes estrictos requieren un conocimiento preciso de la representación numérica. Dado que la pregunta se refiere a los medios para cambiar de color cualquier tipo de control, un nodo de propiedad estricta no es suficiente. Un nodo de propiedad implícito (C), sólo puede ser usado localmente. La respuesta D muestra una propiedad que ni siquiera es relevante en esta cuestión. Por lo tanto, la opción B es la mejor respuesta.

Referencias: *Cambio entre Tipos Estrictos y Tipos Débiles de Control Refnums, Nodos de Propiedad*

8. **Respuestas correctas:** A

Tema: Graficos y Tablas

Justificación: Los Strip Charts empiezan el trazado de derecha a izquierda y continúan desplazándose durante el trazado. Los Scope Charts empiezan el trazado de izquierda a derecha y continúan hasta que el Chart está lleno. Entonces el Chart se limpia, y el trazado comienza de nuevo por la izquierda. Los Sweep Charts se comportan como los Scope Charts, excepto que cuando el Chart se rellena, los Sweep Charts comienzan el trazado a la izquierda y progresivamente sobrescriben los datos previos. No hay tal cosa como un Step Chart en LabVIEW.

Referencias: *Personalizando Gráficos y Charts*

9. **Respuestas correctas:** D

Tema: Variables locales

Justificación: Las variables locales no conforman el paradigma de Flujo de Datos porque se comunican por referencia, no por valor. La premisa básica de las variables locales es permitir la transferencia de datos cuando es imposible cablearlos. Esto evita el paradigma de Flujo de Datos.

Referencias: *Flujo de Datos en el Diagrama de Bloques, Variables Locales, Usando Cuidadosamente Variables Locales y Globales*

10. **Respuestas correctas:** C

Tema: Patrones de diseño

Justificación: Una máquina de estados puede ser usada para obtener la misma funcionalidad que una estructura secuencial. Sin embargo, las máquinas de estado permiten al desarrollador para determinar programáticamente la secuencia en tiempo de ejecución. Esto hace que la implementación con una máquina de estados sea más escalable.

Referencias: *Estructuras de Secuencia y Caso*

11. **Respuestas correctas:** C

Tema: Bucles

Justificación: El bucle While tiene un valor Booleano FALSE cableado a su terminal condicional, configurado para pararse si es TRUE. Por lo tanto, no hay una condición que provoque la parada del

bucle While. Por lo tanto, el bucle While podría ejecutarse indefinidamente y el VI deberá ser abortado.

Referencias: *Estructuras de Bucle For y While, Bucle While*

12. **Respuestas correctas:** A

Tema: Principios de programación de LabVIEW

Justificación: Debido a que LabVIEW es un lenguaje de Flujo de Datos, podemos trazar el flujo de datos en el diagrama de bloques para ver que operaciones se ejecutan primero, segundo, y así sucesivamente.

Referencias: *Flujo de Datos en el Diagrama de Bloques*

13. **Respuestas correctas:** D

Tema: Gestión de errores

Justificación: Si se activa la gestión automática de errores en el VI principal, y los terminales de error del subVI no están cableados, LabVIEW automáticamente maneja los errores mostrando un cuadro de diálogo.

Referencias: *Manejo de Errores*

14. **Respuestas correctas:** D

Tema: Entorno de LabVIEW

Justificación: Los iconos no pueden ser editados desde la paleta de funciones. Los iconos de los subVI pueden ser editados clicando con el botón derecho en el icono de la parte superior derecha del VI y seleccionando **Edit Icon**.

Referencias: *Creando un Icono de VI*

15. **Respuestas correctas:** A

Tema: Sincronización y Comunicación

Justificación: Colas, notificadores, y variables locales están todas diseñadas para transferir datos. Los semáforos no pasan datos. En cambio, su único propósito es prevenir que ciertas partes de código se ejecuten cuando otras secciones críticas se están ejecutando.

Referencias: *Sincronización de VIs y Funciones*

16. **Respuestas correctas:** D

Tema: Entorno de LabVIEW

Justificación: Tienes que mirar en el cluster de salida de error o en un diálogo de error para encontrar el código de error.

Referencias: *Ventana Error list*

17. **Respuestas correctas:** A,D

Tema: Arrays y Clusters

Justificación: La Auto-indexación es una característica para bucles que interactúan con arrays. Las funciones de arrays por sí mismas no tienen características iterativas de auto-indexación.

Referencias: *Estructuras de Bucles For y While, Habilitando Auto-Indexación para Bucles*

18. **Respuestas correctas:** A

Tema: Arrays y Clusters

Justificación: La función Initialize Array crea un array con una longitud especificada por dimension size.

El valor de cada elemento está especificado por el elemento input. Para inicializar arrays

multidimensionales, puedes simplemente expandir la función Initialize Array para mostrar más entradas

de dimension size.

Referencias: *Función Initialize Array*

19. **Respuestas correctas:** A

Tema: Estructuras de eventos

Justificación: Cada posible respuesta se refiere a un evento de Cambio de Valor. Los eventos de Cambio de Valor son generados por la interacción del usuario con el panel frontal, o por una llamada a un Value (Signaling) Property Node. Llamar a un Value Property Node no genera un evento.

Referencias: *Eventos disponibles, Usando Eventos en LabVIEW*

20. **Respuestas correctas:** A

Tema: Variables Globales Funcionales

Justificación: Puedes introducir datos críticos o secciones de código en variables globales funcionales. Como las variables globales funcionales son VIs no reentrantes, la posibilidad de condiciones de carrera se elimina.

Referencias: *Sugerencias para el uso de Sistemas de ejecución y Prioridades*

21. **Respuestas correctas:** C

Tema: Bucles

Justificación: El bucle For se ejecuta 5 veces. Empezando con el valor 1, el resultado de la iteración previa se multiplica por 2. Por lo tanto, el valor en el indicador después de 5 iteraciones es equivalente a $1 \times 2 \times 2 \times 2 \times 2$ (2^5), o 32.

Referencias: *Usando Registros de Desplazamiento para recordar valores de iteración*

22. **Respuestas correctas:** C

Tema: Arrays y Clusters

Justificación: La función Array Subset toma un array de entrada y devuelve una sección especificada del array, empezando por el índice dado y continuando un numero de elementos igual a la longitud. Aquí, el valor del índice 3 especifica el cuarto elemento del array, o 10. Como la longitud especificada tiene un valor 4, se devuelve un array de longitud 4 como el siguiente: {10, 8, 5, 7}.

Referencias: *Función Array Subset*

23. **Respuestas correctas:** A

Tema: Documentación

Justificación: La ventana de documentación en las propiedades del VI es el único sitio donde editar la información sobre el VI que aparece en la Ayuda Contextual.

Referencias: *Descripción de Propiedades del VI, Página de Documentación (Caja de Diálogo de Propiedades del VI)*

24. **Respuestas correctas:** D

Tema: Bucles

Justificación: El Bucle For se refiere a la entrada en el terminal Count para determinar cuantas iteraciones ejecuta.

Referencias: *Bucle For*

25. **Respuestas correctas:** C

Tema: Nodos de Propiedad

Justificación: Como el terminal indicador no está aislado de los datos que están escribiendo las

estructuras, no hay razón para no cablear directamente al indicador en lugar de utilizar un Property Node. Es siempre mejor práctica cablear directamente cuando sea posible, porque usando variables o Property Nodes para actualizar valores puede provocar condiciones de carrera si no se usan cuidadosamente.

Referencias: *Flujo de Datos en el Diagrama de Bloques, Usando Cuidadosamente Variables Locales y Globales*

26. **Respuestas correctas:** A

Tema: Temporización

Justificación: Las respuestas B y D son incorrectas porque la función Wait Until Next ms Multiple se ejecuta antes de que termine la ejecución del bucle, no antes. La respuesta C es incorrecta, porque describe la funcionalidad de la función Wait (ms).

Referencias: *Función Wait Until Next ms Multiple*

27. **Respuestas correctas:** B

Tema: Sincronización y Comunicación

Justificación: La respuesta C no es correcta porque los semáforos no transmiten datos. La respuesta A no es correcta porque los notificadores transmiten datos, pero sólo pueden transmitir datos de uno en uno. Los datos se sobrescriben y se pierden si el programa escribe en el notificador dos veces antes de que el dato sea leído. La respuesta D es incorrecta porque las variables locales no tienen mecanismos para determinar cuando la variable ha sido actualizada, así que no hay forma de decir si el dato es una nueva adquisición o no. Las colas soportan múltiples elementos y operan usando un principio FIFO, garantizando que ningún dato se pierda o sobrescriba.

Referencias: *Sincronización de VIs y Funciones*

28. **Respuestas correctas:** B

Tema: Arrays y Clusters

Justificación: No hay necesidad de hacer un array de arrays, puedes simplemente añadir dimensiones a un array existente.

Referencias: *Cambiando las Dimensiones de un Array*

29. **Respuestas correctas:** A

Tema: Estructuras de Casos

Justificación: Los arrays no se aceptan por el terminal de selector de caso porque el terminal de selector de caso requiere un valor escalar.

Referencias: *Creando Estructuras de Caso*

30. **Respuestas correctas:** D

Tema: E/S de ficheros

Justificación: LabVIEW representa los arrays como una lista de bytes que contienen una cabecera y un array de datos. La cabecera contiene un entero de 4 bytes para cada dimensión que especifica la longitud de esa dimensión. Siguiendo a la cabecera se encuentra el actual array de datos. En la pregunta, el array tiene dos dimensiones. Tiene 4 bytes en la cabecera por cada dimensión que da un total de 8 bytes en la cabecera. Como el array está compuesto por 8 bits, o enteros de 1 byte, hay un total de 9 bytes en el array de datos actual. Esto hace un total de 17 bytes escribiéndose al archivo.

Referencias: *Flattened Data, Creando Archivos Binarios, Cómo Guarda LabVIEW los Datos en Memoria*

31. **Respuestas correctas:** B

Tema: Estructuras Secuenciales

Justificación: La secuencia local es escrita por primera vez en el frame 1. Por lo tanto, en el frame 0, no hay datos disponibles para leer porque los datos no han sido escritos todavía.

Referencias: *Añadiendo y Quitando Terminales de Secuencia Locales*

32. **Respuestas correctas:** C

Tema: Temporización

Justificación: La función Wait (ms) no hace nada para liberar o asignar memoria o especificar núcleo del procesador. Todo lo que hace es provocar una pausa de un corto período de tiempo en la ejecución de un VI para dar tiempo al procesador a completar otras tareas.

Referencias: *Función Wait (ms)*

33. **Respuestas correctas:** A

Tema: Tipos de datos

Justificación: Los Rings pueden usar cualquier representación numérica mientras que Enums pueden usar solo enteros sin signo.

Referencias: *Ring Constant, Enum Constant*

34. **Respuestas correctas:** C

Tema: Estructuras de eventos

Justificación: El Event Case que se muestra ocupa dos eventos: El evento Mouse Down en el control **Button**, y el evento Mouse Down en el panel, o panel frontal. Estos dos eventos ocurren simultáneamente, así que la estructura de eventos los pone en cola, y los maneja primero uno, y luego el otro. Así, se producen dos iteraciones, y el valor del registro de desplazamiento se incrementa dos veces.

Referencias: *Usando eventos en LabVIEW, Evento Mouse Down*

35. **Respuestas correctas:** B

Tema: Principios de programación de LabVIEW

Justificación: Las funciones y los VIs se ejecutan en cuanto todas sus entradas tienen datos. Por lo tanto, es el flujo de datos a través del programa el que determina el orden de ejecución.

Referencias: *Flujo de Datos en el Diagrama de Bloques*

36. **Respuestas correctas:** B

Tema: Tipos de datos

Justificación: Los enteros que se muestran son enteros de 8 bits sin signo. El rango de los enteros de 8 bits es 0-255. El resultado de 2 veces 128 es 256, uno más que el máximo valor permitido por un entero de 8 bits sin signo. Por lo tanto, el valor salta otra vez a 0.

Referencias: *Tabla de Tipos de Datos Numéricos, Conversión Numérica*

37. **Respuestas correctas:** D

Tema: Gestión de errores

Justificación: La función Merge Errors no muestra ningún cuadro de diálogo. Las funciones de One y Two Button Dialog son de propósito general y es mejor no usarlas para las aplicaciones de manejo de errores. No hay entradas de error en estas funciones. No existe ningún VI como Generate Front Panel Activity. El Simple Error Handler es la mejor opción porque acepta una entrada de cluster de error y muestra un

cuadro de diálogo de error al usuario en caso de error.

Referencias: *Manejando Errores, VI Simple Error Handler*

38. **Respuestas correctas:** B

Tema: Graficos y Tablas

Justificación: Los gráficos XY aceptan un cluster de dos arrays, un array X y un array Y para generar un single plot. Para generar múltiples gráficos, los gráficos XY aceptan un array de estos clusters.

Referencias: *Gráficos XY*

39. **Respuestas correctas:** C

Tema: Acciones Mecánicas de los Booleanos

Justificación: Si la acción mecánica está configurada como Switch Until Release, se generan dos eventos cuando el usuario clikea y suelta. El primer evento es una transición de FALSE a TRUE, y el segundo una transición de TRUE a FALSE. Las acciones Latch están específicamente diseñadas para resetear el valor de un botón después de que el cambio se haya leído sin generar un segundo evento.

Referencias: *Cambiando la Acción Mecánica de un Objeto Booleano, Evento de Cambio de Valor, Usando eventos con un Control Booleano Latch*

40. **Respuestas correctas:** B

Tema: Principios de programación de LabVIEW

Justificación: Cuando sea posible, siempre es mejor cablear los datos directamente a los indicadores. Esto evita el paradigma de Flujo de Datos.

Referencias: *Flujo de Datos en el Diagrama de Bloques, Usando Cuidadosamente Variables Locales y Globales*