



Piensa. Conecta. Construye.

Cuaderno de inicio: cinco ramas, cinco prácticas

Nuestra IA. Una construcción de todos.



Propuestas de sesiones de 60 minutos con datos ficticios, elección del estudiante y comprobación del resultado. No constituyen programas acreditados ni prácticas validadas en instalaciones reales.

Las imágenes son conceptuales y generadas con IA. Los ejercicios son originales de esta propuesta; los libros citados conservan su autoría y sus licencias. No se atribuyen estos ejercicios a sus autores.

01 / Inteligencia artificial

Lógica de Riego con Datos Simulados

Objetivo observable

Comprender la toma de decisiones básica mediante reglas condicionales aplicadas a datos de humedad ficticios.

Materiales y datos

- Hoja de papel para diagramas
- Calculadora simple
- Lista de valores de sensores: [18, 32, 47, 61]

Elige tu enfoque y tu ritmo

El estudiante elige si simular un día seco (valores bajos) o húmedo (valores altos) para ajustar su regla.

Práctica guiada

1. Revisar los valores de humedad proporcionados por los sensores ficticios.
2. Definir la regla simple: Si humedad < 30 , entonces sugerir riego.
3. Aplicar la regla a cada valor de la lista [18, 32, 47, 61].
4. Registrar en papel qué sensores activan la alerta de riego.
5. Discutir por qué el valor 32 no activa la alerta aunque esté cerca del límite.
6. Concluir que esta es una lógica básica y no sustituye a un agrónomo real.

Tu evidencia y autocorrección

Crear un diagrama de flujo en papel que represente la decisión de riego para los cuatro sensores.

Bitácora del estudiante

Mi elección: _____

Datos y procedimiento: _____

Resultado observado: _____

Comprobación esperada

Identificar correctamente que solo el sensor con valor 18 requiere riego según la regla establecida.

Antes de consultar la respuesta, compara tu procedimiento con los datos y explica cualquier diferencia.

Pistas progresivas de SmartChat

1. Recuerda que la condición es estrictamente menor a 30.
2. Dibuja un rombo para la decisión y dos caminos (Sí/No).
3. Esta simulación no garantiza resultados en campo real.

Reflexión

¿Cómo cambiaría la regla si el cultivo fuera diferente? ¿Por qué es importante definir umbrales claros?

Criterio de logro: resultado correcto, procedimiento reproducible, autocorrección explicada y una limitación identificada.

Referencia de apoyo; no fuente de este ejercicio original:
<https://d2l.ai/>

02 / Redes y ciberseguridad

Diagnóstico de Subred y Pasarela Incorrecta

Objetivo observable

Visualizar la estructura de una red local y diagnosticar un error de configuración de pasarela sin conexión externa.

Materiales y datos

- Papel y lápiz para diagramas de red
- Definición de subred: 192.168.10.0/24
- Nodos definidos: .10, .20, .30

Elige tu enfoque y tu ritmo

El estudiante decide qué nodo (10, 20 o 30) intentará conectarse a internet primero para encontrar el error.

Práctica guiada

1. Dibujar la red local con la subred 192.168.10.0/24 y los tres nodos.
2. Asignar la pasarela correcta: 192.168.10.1.
3. Simular un escenario donde el nodo intenta usar la pasarela incorrecta: 192.168.20.1.
4. Analizar por qué el tráfico no sale de la red local con esa configuración.
5. Identificar que 192.168.20.1 pertenece a una subred diferente.
6. Corregir el diagrama apuntando al gateway .1.

Tu evidencia y autocorrección

Dibujar un diagrama de papel mostrando el tráfico bloqueado hacia la pasarela incorrecta y corregirlo.

Bitácora del estudiante

Mi elección: _____

Datos y procedimiento: _____

Resultado observado: _____

Comprobación esperada

Reconocer que 192.168.20.1 está fuera del rango de la subred 192.168.10.0/24.

Antes de consultar la respuesta, compara tu procedimiento con los datos y explica cualquier diferencia.

Pistas progresivas de SmartChat

1. La máscara /24 cubre del .1 al .254 en el tercer octeto 10.
2. El tráfico hacia otra subred requiere un router correcto.
3. Este ejercicio es puramente teórico y no usa hardware real.

Reflexión

¿Qué sucede en una red real cuando se configura mal la puerta de enlace? ¿Cómo afecta esto a la comunicación?

Criterio de logro: resultado correcto, procedimiento reproducible, autocorrección explicada y una limitación identificada.

Referencia de apoyo; no fuente de este ejercicio original:
<https://www.computer-networking.info/>

03 / Nube y centros de datos

Comparativa de Disponibilidad: Local vs Remoto

Objetivo observable

Calcular el tiempo de inactividad permitido basado en un porcentaje de disponibilidad y comparar modelos.

Materiales y datos

- Calculadora
- Concepto de disponibilidad del 99.9%
- Periodo de análisis: 30 días

Elige tu enfoque y tu ritmo

El estudiante elige si calcular el tiempo en minutos o en horas para visualizar mejor la pérdida.

Práctica guiada

1. Definir el total de minutos en 30 días ($30 * 24 * 60$).
2. Calcular el 0.1% del tiempo total como tiempo máximo de inactividad.
3. Verificar que el resultado sea aproximadamente 43.2 minutos.
4. Comparar este límite con una aplicación local sin redundancia.
5. Discutir que el modelo en la nube no promete cero fallos, solo alta disponibilidad.
6. Concluir las diferencias de responsabilidad entre mantener un servidor local y usar un servicio remoto.

Tu evidencia y autocorrección

Realizar el cálculo matemático para determinar los 43.2 minutos de inactividad permitidos.

Bitácora del estudiante

Mi elección: _____

Datos y procedimiento: _____

Resultado observado: _____

Comprobación esperada

Obtener el valor correcto de 43.2 minutos y entender que es un límite, no una garantía absoluta.

Antes de consultar la respuesta, compara tu procedimiento con los datos y explica cualquier diferencia.

Pistas progresivas de SmartChat

1. Total minutos = 43200.
2. 0.1% de 43200 es 43.2.
3. La nube reduce el riesgo pero no lo elimina.

Reflexión

¿Es aceptable tener 43 minutos de caída al mes para tu negocio? ¿Qué implica esto para la planificación?

Criterio de logro: resultado correcto, procedimiento reproducible, autocorrección explicada y una limitación identificada.

Referencia de apoyo; no fuente de este ejercicio original:
<https://kubernetes.io/docs/tutorials/kubernetes-basics/>

04 / Ingeniería y robótica

Misión de Dron Estacionario en Cuadrícula Teórica

Objetivo observable

Planificar una misión de cobertura utilizando lógica de cuadrícula sin hardware físico ni vuelo real.

Materiales y datos

- Cuadrícula impresa o dibujada de 20x10 metros
- Celdas definidas de 5x5 metros
- Marcador para trazar rutas

Elige tu enfoque y tu ritmo

El estudiante elige la dirección inicial del barrido (horizontal o vertical) para cubrir la cuadrícula.

Práctica guiada

1. Dibujar un área rectangular de 20m x 10m.
2. Dividir el área en celdas cuadradas de 5m x 5m.
3. Calcular el número total de celdas necesarias (8 celdas).
4. Planificar una ruta lógica para que un dron estacionario 'escanee' cada celda.
5. Simular la misión en papel sin considerar baterías ni viento.
6. Reflexionar sobre cómo se adaptaría esto a un entorno con obstáculos reales.

Tu evidencia y autocorrección

Trazar en la cuadrícula de papel la ruta óptima para cubrir las 8 celdas sin repetir pasos innecesarios.

Bitácora del estudiante

Mi elección: _____

Datos y procedimiento: _____

Resultado observado: _____

Comprobación esperada

Identificar correctamente que se necesitan 8 celdas y trazar una ruta coherente.

Antes de consultar la respuesta, compara tu procedimiento con los datos y explica cualquier diferencia.

Pistas progresivas de SmartChat

1. 20m dividido por 5m da 4 columnas.
2. 10m dividido por 5m da 2 filas.
3. Total de celdas es $4 * 2 = 8$.

Reflexión

¿Cómo afecta el tamaño de la celda a la precisión del mapeo? ¿Qué limitaciones tiene un dron estacionario?

Criterio de logro: resultado correcto, procedimiento reproducible, autocorrección explicada y una limitación identificada.

Referencia de apoyo; no fuente de este ejercicio original:
<https://modernrobotics.northwestern.edu/nu-gm-book-resource/>

05 / Desarrollo de software

Validación de Entradas y Control de Errores

Objetivo observable

Diseñar un validador lógico que acepte valores dentro de un rango específico y rechace los inválidos.

Materiales y datos

- Lista de valores de entrada: [0, 50, 100, -1, 101]
- Regla de validación: Valores válidos entre 0 y 100 inclusive
- Papel para registrar resultados

Elige tu enfoque y tu ritmo

El estudiante decide si el sistema debe mostrar un mensaje de error específico o simplemente ignorar la entrada.

Práctica guiada

1. Establecer las condiciones: válido si valor ≥ 0 y valor ≤ 100 .
2. Procesar el valor 0 (Válido).
3. Procesar el valor 50 (Válido).
4. Procesar el valor 100 (Válido).
5. Procesar el valor -1 (Inválido) y el valor 101 (Inválido).
6. Contabilizar: 3 entradas válidas esperadas, 2 inválidas.

Tu evidencia y autocorrección

Crear una tabla de verdad en papel que clasifique cada uno de los 5 valores como Válido o Inválido.

Bitácora del estudiante

Mi elección: _____

Datos y procedimiento: _____

Resultado observado: _____

Comprobación esperada

Clasificar correctamente 0, 50 y 100 como válidos; -1 y 101 como inválidos.

Antes de consultar la respuesta, compara tu procedimiento con los datos y explica cualquier diferencia.

Pistas progresivas de SmartChat

1. El rango incluye los extremos 0 y 100.
2. Cualquier número negativo o mayor a 100 falla la prueba.
3. Esta lógica es la base de la seguridad en software.

Reflexión

¿Por qué es crucial validar las entradas antes de procesarlas en un software real? ¿Qué pasa si no se valida?

Criterio de logro: resultado correcto, procedimiento reproducible, autocorrección explicada y una limitación identificada.

Referencia de apoyo; no fuente de este ejercicio original:
<https://openstax.org/books/introduction-python-programming/pages/preface>

Biblioteca y recursos de apoyo

Selecciona lecturas según tu nivel. Los enlaces apuntan a las editoriales o proyectos originales. Los materiales de terceros no se renombran como libros propios de dIAlectica.

IA / Dive into Deep Learning

<https://d2l.ai/>

Libro y PDF del proyecto. Texto CC BY-SA 4.0; conservar atribución y condiciones de adaptación.

Redes / Computer Networking: Principles, Protocols and Practice

<https://www.computer-networking.info/>

Libro abierto con ejercicios y enlaces oficiales a PDF. Consultar la licencia de la edición antes de redistribuir.

Nube / Kubernetes Basics

<https://kubernetes.io/docs/tutorials/kubernetes-basics/>

Tutorial oficial. Apoyo para una práctica posterior de despliegue; no demuestra que exista nube operativa propia.

Robótica / Modern Robotics

<https://modernrobotics.northwestern.edu/nu-gm-book-resource/>

Material de autores y universidad. Enlace de consulta; no se presupone permiso para reimprimir el libro.

Software / Introduction to Python Programming, OpenStax

<https://openstax.org/books/introduction-python-programming/pages/preface>

Lectura y PDF oficial. CC BY-NC-SA 4.0: preservar atribución y condiciones no comerciales y de compartir igual.

Nota editorial

El PDF de Dive into Deep Learning descargado se conserva sin modificar en books/. El PDF de OpenStax no pudo descargarse en esta sesión; está enlazado a su fuente oficial. Los recursos de nube y robótica se enlazan sin copiar ni atribuir permisos no verificados.

Los umbrales agrícolas y la cuadrícula del dron son simulaciones en papel. No son instrucciones de riego ni de vuelo. El acceso al aula público actual es para adultos; cualquier futura actividad con menores requiere su flujo específico.