



Cuestionario del primer ejercicio

Especialidad: A6 D2. Programación y computación científica.

Por favor, lea detenidamente antes de comenzar:

- **NO** abra el **CUESTIONARIO** ni empiece el examen hasta que se le indique.
- Para realizar este primer ejercicio se hace entrega de dos documentos:
 1. Cuadernillo con el **cuestionario de preguntas tipo test**, con cuatro respuestas alternativas sobre las materias del programa de esta convocatoria.
 2. **Hoja de respuestas** donde se consignará la respuesta correcta a cada pregunta.
- Al finalizar la prueba se hará entrega de la hoja de respuestas. El cuadernillo con el cuestionario se podrá utilizar como borrador y se podrá llevar por el opositor al finalizar el tiempo marcado para el ejercicio.
- Sólo se calificará las respuestas marcadas en la **HOJA DE RESPUESTAS**
- Una vez abierto el cuestionario, compruebe que consta de todas las páginas y preguntas y que sea legible. En caso contrario solicite uno nuevo al personal del aula.
- Verifique que el número de la solapa donde se recogen sus **datos personales coincide con el número de la hoja** de examen donde se consignan las respuestas.
- El examen se realizará con bolígrafo azul o negro. Si no dispone de uno, solicítelo al Tribunal.
- El cuestionario consta de **100 (CIEN) preguntas**, cada una de ellas con **cuatro respuestas alternativas**, de las cuales **sólo una de ellas es correcta, más 5 (CINCO) preguntas adicionales de reserva**, que serán valoradas en el caso de que se anule algunas de las 100 anteriores. **Marque con una equis la respuesta elegida** en la celda correspondiente a la pregunta, de forma clara (ver fig. 1).
- **Las respuestas múltiples, poco claras o dudosas, se considerarán como no contestadas.** Si desea corregir una respuesta, **rodee la opción incorrecta** con un círculo (ver fig. 2) y marque con una equis la nueva opción que elige.

1	A	B	C	D
	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Fig 1

1	A	B	C	D
	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>

Fig 2.

- Todas las preguntas tendrán el mismo valor y **las contestaciones erróneas se penalizarán** con 1/3 del valor de una contestación correcta.
- **NO Separe** ninguna de las copias de la **HOJA DE RESPUESTAS**. Una vez finalizado, el personal del aula le indicará los pasos a seguir.
- **Dispone de 90 minutos**, máximo, para realizar este ejercicio.



Proceso selectivo por el sistema de acceso libre para ingreso en la Escala de Tecnólogos de los Organismos Públicos de Investigación, convocado por resolución de 22 de diciembre de 2025 (BOE N°314 30 de diciembre) – OEP 2023-2024-2025
Primer Ejercicio

Fecha:
10/05/2026
Página: 2 de 27



1. ***¿Cuál es el principal riesgo técnico de la entrega continua en un modelo de desarrollo incremental?***
 - A) La disminución drástica de la documentación técnica y de usuario.
 - B) El incremento exponencial del coste de gestión de requisitos.
 - C) La degradación de la estructura del software o corrupción del sistema.
 - D) El exceso de validación por parte del cliente en fases tempranas.

2. ***Según el Principio de Responsabilidad Única (SRP), ¿cómo se define la cohesión óptima de un módulo?***
 - A) El módulo debe realizar una única función lógica y tener una sola razón para cambiar.
 - B) El módulo debe centralizar todas las fórmulas del proyecto para minimizar dependencias.
 - C) El código debe estar diseñado para ejecutarse exclusivamente en un único hilo de ejecución.
 - D) La clase debe integrar la lógica de cálculo junto con la gestión de persistencia en disco.

3. ***¿Cuál es una característica técnica esencial para definir un requisito de software de calidad?***
 - A) La ambigüedad controlada para permitir flexibilidad en la implementación científica.
 - B) La capacidad de ser verificado mediante una prueba técnica u objetiva.
 - C) La extensión descriptiva mínima de una página para asegurar el detalle algorítmico.
 - D) La identificación nominal obligatoria del desarrollador que realizará la implementación.

4. ***¿Cuál es el objetivo principal de la fase de "Verificación" dentro del modelo de ciclo de vida en cascada según el estándar IEEE?***
 - A) Validar que el sistema final satisface las necesidades del usuario en su entorno real.
 - B) Transformar los requisitos funcionales en una arquitectura lógica detallada del sistema.
 - C) Comprobar que el producto de una fase determinada cumple con las especificaciones de la fase anterior.
 - D) Corregir errores latentes detectados durante el uso operativo de la aplicación científica.



5. *¿Qué responsabilidad define al Product Owner en Scrum científico?*

- A) Supervisar la arquitectura paralela en el clúster.
- B) Facilitar las ceremonias diarias del equipo.
- C) Priorizar el Backlog para asegurar el máximo valor científico.
- D) Validar individualmente cada commit del repositorio.

6. *¿Cuál es la función del "Lead Time" en un flujo Kanban?*

- A) Medir líneas de código por investigador.
- B) Calcular el tiempo total desde la petición hasta la entrega.
- C) Determinar el número máximo de tareas en ejecución.
- D) Evaluar la complejidad algorítmica previa.

7. *¿Qué ventaja técnica ofrecen las "Pull Requests" (PR) en GitHub?*


- A) Aumentar la velocidad de los scripts de Python.
- B) Facilitar la revisión por pares antes de integrar cambios.
- C) Encriptar resultados para garantizar confidencialidad.
- D) Sustituir la necesidad de realizar pruebas unitarias.

8. *¿Cómo debe resolverse un conflicto de fusión (merge conflict) en Git?*

- A) Resolviendo manualmente las discrepancias en el código.
- B) Borrando el repositorio local y descargando el remoto.
- C) Notificando al administrador para reiniciar el clúster.
- D) Forzando el envío de cambios con el comando --force.

9. *¿Qué diferencia a un IDE de un editor de texto plano?*

- A) Poder guardar archivos en formato .py o .r.
- B) La imposibilidad de conectarse a servidores remotos.
- C) La integración de depurador e inspección de memoria.
- D) El uso obligatorio de interfaz gráfica para comandos.

	Proceso selectivo por el sistema de acceso libre para ingreso en la Escala de Tecnólogos de los Organismos Públicos de Investigación, convocado por resolución de 22 de diciembre de 2025 (BOE N°314 30 de diciembre) – OEP 2023-2024-2025 Primer Ejercicio	Fecha: 10/05/2026 Página: 5 de 27
---	--	---

10. *¿Cuál es la utilidad primordial de integrar un Linter en el IDE?*

- A) Detectar errores sintácticos y problemas de estilo en tiempo real.
- B) Compilar código para su ejecución directa en GPU.
- C) Optimizar el ancho de banda en la transferencia de datos.
- D) Generar documentación técnica según la norma ISO 1016.

11. *En el contexto del análisis de una "propagación de error", donde el fallo observado es consecuencia de una cadena de llamadas a funciones previa, ¿qué técnica proporciona una "instantánea" del camino de ejecución recorrido?*

- A) Puntos de interrupción condicionales.
- B) La pila de llamadas (call stack).
- C) El perfilador de red de las herramientas de desarrollo.
- D) El uso de comentarios para aislar segmentos de código.

12. *¿Cuál es la diferencia técnica fundamental entre un punto de interrupción de línea y uno condicional?*

- A) El de línea solo funciona en IDEs gráficos, mientras que el condicional es exclusivo de depuradores de consola como GDB.
- B) El de línea detiene la ejecución siempre, mientras que el condicional requiere que una excepción sea lanzada previamente.
- C) El de línea detiene la ejecución al alcanzar una instrucción específica, mientras que el condicional solo lo hace si se satisface un criterio lógico definido.
- D) El condicional es una técnica básica de depuración, mientras que el de línea se considera una técnica avanzada.

13. *Un algoritmo divide el problema en dos subproblemas de tamaño $n/2$ y combina en tiempo $O(n)$. Su complejidad es:*

- A) $O(n \log n)$
- B) $O(n^2)$
- C) $O(\log n)$
- D) $O(n)$



14. *Al realizar un análisis espacial a priori de un algoritmo, ¿qué elementos deben sumarse para obtener una medida del espacio necesario?*
- A) Únicamente el tamaño de las variables globales definidas al inicio.
 - B) El tiempo de ejecución multiplicado por el número de instrucciones.
 - C) La suma de las celdas de memoria estáticas (variables globales) y las celdas dinámicas (como el espacio de la pila en recursividad).
 - D) Solo la memoria de entrada y salida requerida por los periféricos
15. *Tienes un sistema que recibe mensajes continuamente y deben procesarse según su prioridad y no solo por orden de llegada. ¿Qué estructura de datos usarías para gestionar esta lista de forma eficiente?*
- A) Pila (Stack): porque el último mensaje más prioritario en llegar debe ser el primero en salir.
 - B) Cola (Queue): para garantizar que se respete estrictamente el orden de llegada.
 - C) Cola con prioridad implementada con lista enlazada (inserción $O(n)$ y extracción $O(1)$): porque permite procesar antes los mensajes que tiene asociados mayor prioridad.
 - D) Cola con prioridad implementada con binary heap (inserción y extracción con $O(\log n)$): porque permite procesar antes los mensajes que tiene asociados mayor prioridad y es más eficiente.
16. *Analice la siguiente estructura de control:*
- ```
cont = 1;
while (cont <= n) {
 x = x + a[cont];
 cont = cont + 1;
}
```
- ¿Cuál es su función de complejidad temporal  $f(n)$  exacta en términos de instrucciones básicas (asignaciones, comparaciones y saltos)?*
- A)  $f(n)=3n+2$
  - B)  $f(n)=2n+3$
  - C)  $f(n)=4n+3$
  - D)  $f(n)=4n+1$



17. *Se desarrolla una aplicación de mapas para representar una red de carreteras que conecta diversas ciudades, donde algunas rutas tienen un solo sentido y otras son de doble sentido. ¿Qué estructura de datos permite representar estas conexiones y sus distancias de forma más natural?*
- A) Árbol general
  - B) Grafo
  - C) Cola de prioridad
  - D) Pila
18. *¿Qué diferencia fundamental de comportamiento y aplicación existe entre una Pila (Stack) y una Cola (Queue) en el desarrollo de software?*
- A) La pila es una lista FIFO utilizada para procesos por lotes, mientras que la cola es LIFO para subrutinas.
  - B) La pila solo permite inserciones al final, mientras que la cola permite inserciones en cualquier extremo.
  - C) La pila se utiliza para almacenar direcciones de retorno de subrutinas, mientras que la cola gestiona trabajos según su orden de llegada.
  - D) En la pila el acceso se realiza por el primer elemento de la lista y en la cola por el último elemento de la lista.
19. *¿Cuál de los pilares de la Programación Orientada a Objetos se encarga de ocultar los detalles internos de un objeto y proteger su estado de modificaciones no autorizadas?*
- A) Herencia
  - B) Abstracción
  - C) Polimorfismo
  - D) Encapsulamiento
20. *¿Cuál de los pilares de la Programación Orientada a Objetos se encarga de ocultar los detalles internos de un objeto y proteger su estado de modificaciones no autorizadas?*
- A) Herencia
  - B) Abstracción
  - C) Polimorfismo
  - D) Encapsulamiento



**21. En Java, tenemos lo siguiente:**

```
String saludo = new String ("¡Hola OPIs!");
saludo.replaceAll("Hola", "Buenas");
```

**Se puede afirmar que:**

- A) la cadena “saludo” ahora contiene “¡Buenas OPIs!”
- B) el método ha devuelto una nueva cadena que contiene “Buenas OPIs!”
- C) al ejecutar “saludo.replaceAll(“Hola”, “Buenas”);” se produce un error
- D) la clase String no es inmutable

**22. Considera el siguiente código en JavaScript. ¿Cuál de las siguientes afirmaciones es correcta respecto a los principios de programación funcional utilizados en el código?**

```
function incrementar(x) {
 return x + 1;
}
function suma (lista) {
 if (lista.length == 0) return 0;
 return lista[0] + suma (lista.slice(1));
}
// Ejecución
const numeros = [1, 2, 3];
const resultado = suma (numeros);
const nuevoValor = incrementar(resultado);

console.log(numeros);
console.log(nuevoValor);
```

- A) Suma demuestra inmutabilidad porque usa lista.slice(1) para no modificar la lista original.
- B) Incrementar no es una función pura porque puede modificar el parámetro que recibe.
- C) El código no utiliza recursividad porque suma solo llama a funciones puras.
- D) La constante números viola la inmutabilidad porque se reasigna dentro de suma.

**23. En programación funcional, una función pura se caracteriza por**

- A) Uso de variables globales
- B) Dependencia del estado externo
- C) Ausencia de efectos secundarios
- D) Ejecución paralela obligatoria





24. ***¿Qué ocurre con el Global Interpreter Lock (GIL) cuando un hilo de Python realiza una operación de I/O?***
- A) El hilo mantiene el GIL bloqueado hasta que la operación finaliza.
  - B) El GIL se libera automáticamente para permitir que otros hilos ejecuten bytecode.
  - C) El sistema operativo termina el hilo para evitar el bloqueo del intérprete.
  - D) El GIL se duplica para gestionar la espera de forma asíncrona.
25. ***¿Cómo garantiza Scala la inmutabilidad en la combinación de paradigmas funcional y orientado a objetos?***
- A) Prohibiendo el uso de variables mutables (var) dentro de cualquier clase.
  - B) Mediante el uso de "case classes" y objetos inmutables por defecto en sus colecciones.
  - C) Forzando que el compilador transforme todo efecto secundario en una función pura.
  - D) Delegando al Garbage Collector la eliminación de objetos que sufran cambios de estado.
26. ***¿Cómo optimiza Python la memoria al usar funciones como map o filter?***
- A) Generando una copia profunda de todos los objetos en el stack de llamadas.
  - B) Mediante evaluación estricta que materializa la colección completa de inmediato.
  - C) Utilizando evaluación perezosa (lazy evaluation) que devuelve iteradores bajo demanda.
  - D) Sincronizando el recolector de basura tras cada iteración del bucle principal.
27. ***¿Por qué los arrays de NumPy proporcionan un almacenamiento y manipulación de datos numéricos mucho más eficiente que las listas estándar de Python?***
- A) Porque las listas de Python compilan automáticamente el código C en segundo plano
  - B) Porque NumPy almacena un único puntero a un bloque contiguo de datos de tipo fijo, evitando la sobrecarga de información de tipo para cada elemento.
  - C) Porque NumPy implementa la estructura de datos orientada a objetos array.array que usa punteros múltiples.
  - D) Porque utilizan evaluación perezosa (lazy evaluation) para ignorar valores redundantes.



28. *En el ecosistema de R para ciencia de datos, cuando se interactúa con bases de datos, ¿qué función principal se utiliza mediante el paquete DBI para establecer la conexión con el servidor (DBMS)?*
- A) `odbc::driverConnect()`
  - B) `dbListTables()`
  - C) `DBI::dbConnect()`
  - D) `dbReadTable()`
29. *Al visualizar grandes cantidades de puntos bidimensionales en Matplotlib, ¿qué función permite representar la densidad del conjunto agrupando los datos en una cuadrícula de hexágonos regulares?*
- A) `plt.hist2d`
  - B) `plt.scatter`
  - C) `plt.hexbin`
  - D) `np.histogramdd`
30. *Además de los histogramas y los agrupamientos hexagonales, ¿qué método común de estimación se utiliza en visualización para "difuminar" puntos en el espacio y obtener una función de densidad suave?*
- A) PCA (Análisis de Componentes Principales).
  - B) Estimación de Densidad del Kernel (KDE).
  - C) Segmentación K-means.
  - D) Mapeo Isométrico (Isomap).
31. *¿Qué valor usa Pandas para representar datos faltantes en arrays de tipo float?*
- A) None
  - B) NULL
  - C) NaN
  - D) NA
32. *Al transformar o resumir datos, ¿qué concepto de Pandas permite la evaluación perezosa de conjuntos subyacentes mediante el patrón "separar, aplicar, combinar" (split-apply-combine)?*
- A) `apply()`
  - B) `groupby()`
  - C) `concat()`
  - D) `transform()`



33. *En el flujo de trabajo científico, ¿cuál es la principal función de los Jupyter Notebooks?*
- A) Es un compilador de C++ para redes neuronales.
  - B) Es una base de datos NoSQL que almacena documentos colaborativos en tiempo real.
  - C) Es un entorno basado en navegador que permite la ejecución interactiva de código y el intercambio de análisis de datos.
  - D) Es un sistema de colas de mensajes diseñado para procesamiento por lotes (batch).
34. *¿Qué tipo de computación prima el volumen de trabajo completado frente al rendimiento pico?*
- A) GPU
  - B) HPC
  - C) HTC
  - D) Vectorial
35. *En MPI (Message Passing Interface) con problema fijo, el cómputo baja como 1/p pero la comunicación crece con p. ¿Qué escalado se degrada?*
- A) weak scaling
  - B) strong scaling
  - C) throughput scaling
  - D) latency scaling
36. *En un nodo NUMA, inicializar un array grande desde un único hilo y usarlo después desde todos los sockets puede degradar el rendimiento por:*
- A) Falsa compartición
  - B) Accesos remotos a memoria
  - C) Divergencia de warps
  - D) Inyección SQL
37. *En análisis de rendimiento paralelo, eficiencia se define como*
- A)  $\text{Speedup} \times \text{número de procesadores}$
  - B)  $\text{Speedup} / \text{número de procesadores}$
  - C)  $\text{Número de procesadores} / \text{speedup}$
  - D)  $\text{Tiempo secuencial} \times \text{tiempo paralelo}$



38. ***En MPI, ¿Qué afirmación es correcta sobre MPI\_Bcast?***
- A) Solo puede usarse si todos los ranks están en el mismo nodo
  - B) Todos los procesos del comunicador deben llamarla con parámetros compatibles, o el programa puede colgarse
  - C) Envía un mensaje punto a punto del root a un único rank
  - D) Entraña implícitamente una barrera global idéntica a MPI\_Barrier
39. ***MPI: un MPI\_Send bloqueante puede no bloquear si:***
- A) Siempre bloquea por definición
  - B) Hay buffering interno o protocolo eager aplicable
  - C) Solo si el mensaje es enorme
  - D) Nunca existe buffering en MPI
40. ***Un algoritmo de reducción en GPU usa operaciones atómicas sobre memoria global. ¿Qué problema puede aparecer?***
- A) contención entre threads
  - B) divergencia de warps
  - C) error de coherencia
  - D) overflow de registros
41. ***En CUDA:  $int\ i = blockIdx.x * blockDim.x + threadIdx.x$ ; Esta expresión calcula:***
- A) el número de bloques
  - B) el índice global del hilo
  - C) la memoria compartida
  - D) el warp ID
42. ***Dado el siguiente código en cupy:***
- ```
import cupy as cp
x = cp.arange(10**7)
y = cp.sum(x)
```
- ¿Qué ocurre al hacer: float(y)***
- A) nada
 - B) copia implícita GPU → CPU
 - C) copia CPU → GPU
 - D) elimina memoria



43. ***Dado el siguiente código en C,***

```
for (int i = 1; i < n; i++)  
    a[i] = a[i] + a[i-1];
```

¿Por qué este bucle no se vectoriza fácilmente?

- A) La suma impide SIMD
- B) Dependencia entre iteraciones
- C) El array puede no estar alineado
- D) Falta “restrict”

44. ***Dado el siguiente código,***

```
double sum = 0;  
#pragma omp parallel for  
for(int i = 0; i < n; i++)  
    sum += a[i];
```

¿Cuál es el principal problema?

- A) Uso incorrecto de OpenMP
- B) Condición de carrera
- C) Overflow
- D) Falsa compartición

45. ***Para recorrer una matriz $A[n][m]$ en C (row-major), el acceso más cache-friendly es:***

- A) for j: for i: $A[i][j]$
- B) for i: for j: $A[i][j]$
- C) Da igual
- D) Solo importa L3

46. ***En una reducción paralela, cambiar el orden de suma puede alterar ligeramente el resultado por:***

- A) Condición de carrera obligatoria
- B) No asociatividad de coma flotante
- C) Falta de compilación
- D) Error de paging



47. *Un código tiene una fracción no paralelizable del 5 %. ¿Cuál es el speedup máximo teórico según Amdahl?*
- A) 20
 - B) 50
 - C) 95
 - D) 200
48. *Si el tiempo se va en I/O, optimizar vectorización NumPy dará:*
- A) Gran mejora siempre
 - B) Poca mejora
 - C) Mejora infinita
 - D) Rompe reproducibilidad
49. *Si el profiling muestra que el tiempo se va en I/O, ¿qué aproximación suele ser más adecuada para mejorar el rendimiento?*
- A) Optimizar vectorización con NumPy
 - B) Aumentar el número de hilos para ejecutar más operaciones en paralelo
 - C) Utilizar buffering o agregación
 - D) Se deben minimizar los identificadores únicos y las claves externas (foreign keys).
50. *En el modelado de bases de datos relacionales, ¿cuál es la regla de oro que justifica aplicar la normalización de datos?*
- A) Todo debe agruparse en un documento JSON complejo.
 - B) Se deben evitar estructuras jerárquicas rígidas.
 - C) Si se duplican valores que podrían almacenarse en un solo lugar, el esquema no está normalizado.
 - D) Se deben minimizar los identificadores únicos y las claves externas (foreign keys).
51. *Según la definición histórica de Edgar Codd, ¿cómo está organizada la información fundamentalmente en un modelo relacional?*
- A) En documentos anidados.
 - B) En relaciones (tablas) formadas por colecciones desordenadas de tuplas (filas).
 - C) En grafos dirigidos de nodos y aristas (edges).
 - D) En almacenes de clave-valor.



52. ***Índice B-tree vs hash: ¿qué afirmación es correcta en motores SQL comunes?***
- A) Hash soporta rangos eficientemente
 - B) B-tree soporta igualdad y rangos
 - C) Hash es siempre mejor
 - D) B-tree no sirve para ORDER BY
53. ***En SQL, COUNT(columna) difiere de COUNT(*) porque:***
- A) COUNT(columna) cuenta también NULL
 - B) COUNT(*) ignora filas duplicadas
 - C) COUNT(columna) ignora NULL
 - D) COUNT(*) solo cuenta claves primarias
54. ***Una ventaja comúnmente citada de las bases de datos de documentos frente a las bases relacionales es el esquema flexible. ¿Con qué término técnico se denomina a este modelo en MongoDB u otros sistemas NoSQL?***
- A) Schema-on-write
 - B) Document-shredding
 - C) Esquema estático
 - D) Schema-on-read
55. ***¿Cuál es la ventaja primordial de usar SQL como un "lenguaje de consulta declarativo" frente a las APIs de consulta imperativa tradicionales?***
- A) Facilita iterar manualmente por los registros con punteros.
 - B) Exige que el programador elija y gestione un índice obligatoriamente en memoria.
 - C) Especifica el patrón que deben cumplir los datos, dejando al optimizador de la base de datos la decisión del método algorítmico o de ejecución más eficiente.
 - D) Define variables condicionales fijas y estructuras en bucles for paralelos
56. ***En las arquitecturas modernas diseñadas para la analítica (Data Warehousing/OLAP), ¿qué disposición en disco resulta significativamente más eficiente al escanear enormes sumas de registros buscando valores sobre un solo atributo?***
- A) Almacenamiento orientado a filas (Row-oriented).
 - B) Almacenamiento distribuido NoSQL de clave-valor.
 - C) Almacenamiento orientado a columnas (Column-oriented).
 - D) Fragmentación de documentos (Document-partitioned indices).



57. *El marco MapReduce distribuye el cálculo masivo utilizando dos fases (map y reduce). ¿Qué estricta restricción se impone a estas funciones para garantizar la tolerancia a fallos y reejecución en la máquina?*
- A) Deben acceder repetidamente a bases de datos relacionales subyacentes en red.
 - B) Deben ser funciones puras, que no causen efectos secundarios y operen estrictamente con los datos de entrada pasados.
 - C) Deben invocar hilos (threads) condicionales en C++.
 - D) Deben escribir sus métricas en una memoria compartida por los clústeres.
58. *El marco MapReduce distribuye el cálculo masivo utilizando dos fases (map y reduce). ¿Qué estricta restricción se impone a estas funciones para garantizar la tolerancia a fallos y reejecución en la máquina?*
- A) Deben acceder repetidamente a bases de datos relacionales subyacentes en red.
 - B) Deben ser funciones puras, que no causen efectos secundarios y operen estrictamente con los datos de entrada pasados.
 - C) Deben invocar hilos (threads) condicionales en C++.
 - D) Deben escribir sus métricas en una memoria compartida por los clústeres.
59. *En las arquitecturas modernas de tratamiento de flujos y lotes (Dataflow), ¿cómo se conoce al patrón que integra en paralelo un sistema batch como Hadoop para cálculos exactos y un sistema de procesamiento stream para resultados aproximados e inmediatos?*
- A) Arquitectura de Microservicios.
 - B) Patrón Mediator.
 - C) Arquitectura Actuador-Sensor.
 - D) Arquitectura Lambda.
60. *¿Cuál es la característica principal del Aprendizaje Supervisado (Supervised Learning)?*
- A) El sistema aprende solo mediante el ensayo y error en un entorno dinámico.
 - B) El algoritmo busca estructuras ocultas en datos que no han sido etiquetados previamente.
 - C) El sistema se entrena con datos etiquetados.
 - D) El sistema aprende de una colección de datos etiquetados y no etiquetados.



61. ***Desde un punto de vista técnico, ¿qué es la "Función de Pérdida" (Loss Function) en un modelo de Machine Learning?***
- A) Una medida de cuánta energía eléctrica consume el hardware durante el proceso.
 - B) El proceso de eliminar neuronas innecesarias para que el modelo ocupe menos espacio.
 - C) Una medida que cuantifica la discrepancia entre la salida que predice el modelo y las etiquetas observadas.
 - D) Un algoritmo que traduce el lenguaje natural a código de programación.
62. ***¿Cuál de las siguientes afirmaciones describe mejor la relación entre memoria compartida y paso de mensajes?***
- A) Los programas de memoria compartida requieren hardware específico y no pueden implementarse sobre paso de mensajes.
 - B) Los programas basados en paso de mensajes no pueden ejecutarse en sistemas de memoria compartida.
 - C) Ambos modelos son equivalentes en poder expresivo, ya que uno puede implementarse sobre el otro con mecanismos adecuados.
 - D) El modelo de paso de mensajes es más potente porque permite comunicación explícita entre procesos.
63. ***En muchos sistemas multiprocesador con memoria compartida, ¿por qué un algoritmo basado en memoria compartida con locks frecuentes puede rendir peor que uno basado en paso de mensajes?***
- A) Porque el acceso a memoria compartida siempre es más lento que la red.
 - B) Porque la contención y coherencia de caché pueden introducir costes de sincronización elevados.
 - C) Porque los algoritmos de paso de mensajes no necesitan sincronización.
 - D) Porque la comunicación por red es siempre más rápida que la memoria.
64. ***CAP: En presencia de partición de red, un sistema CP suele:***
- A) Mantener disponibilidad siempre
 - B) Sacrificar disponibilidad para mantener consistencia
 - C) Sacrificar tolerancia a partición
 - D) Garantizar todo



65. *¿Qué afirmación describe correctamente la diferencia fundamental entre contenedores y máquinas virtuales?*
- A) Los contenedores virtualizan hardware completo
 - B) Las máquinas virtuales comparten el kernel del host
 - C) Los contenedores comparten el kernel del host
 - D) Los contenedores requieren hipervisor
66. *¿Qué mecanismo Kubernetes permite desplegar una nueva versión de un contenedor sin detener el servicio?*
- A) Restart policy
 - B) PersistentVolume
 - C) Rolling update
 - D) Node selector
67. *¿Qué flujo de trabajo Git favorece 'Continuous Deployment'?*
- A) GitFlow
 - B) GitLab flow orientado a versiones de software (releases)
 - C) Trunk-Based development
 - D) GitHub Flow con ramas de larga duración
68. *¿Qué componente suele ejecutar las tareas definidas en un pipeline CI/CD?*
- A) GitHub Action
 - B) Pipeline Artifact
 - C) Repositorio de control de versiones
 - D) Runner o agente
69. *¿Qué modelo de control de accesos representa mejor esta política?*
- if user.department == resource.department and request.hour < 18: allow()*
- A) RBAC, porque department actúa como rol
 - B) ABAC, porque usa atributos de usuario, recurso y contexto
 - C) ACL, porque representa una lista de control de accesos.
 - D) DAC, porque el usuario pertenece a un departamento



70. En un sistema distribuido se registra:

serviceA: login success
serviceB: access granted

¿Qué falta para poder reconstruir la acción completa?

- A) timestamps
- B) correlation ID
- C) logs más grandes
- D) nonce

71. En Snakemake, el DAG (Directed Acyclic Graph) representa:

- A) Versionado Git
- B) Topología de red
- C) Dependencias
- D) Orden de commits

72. Un laboratorio de genómica procesa sus datos en la nube, ¿Qué debe usar para proteger los datos específicamente cuando se están procesando en la CPU?

- A) Cifrado de datos en reposo (AES-256)
- B) Computación confidencial
- C) Seguridad de la capa de transporte (TLS)
- D) Tokenización de datos de salud.

73. ¿Qué función del cliente AWS S3 reemplaza una versión anterior de un fichero en la nube?

- A) `s3_client.upload_file`
- B) `s3_client.update_file`
- C) `s3_client.refresh_object`
- D) `s3_client.patch_object`

74. En análisis de grafos, la complejidad de Dijkstra con heap binario es:

- A) $O(V^2)$
- B) $O(E \log V)$
- C) $O(V \log V)$
- D) $O(E^2)$



75. ***Cuando representamos la información mediante un "Grafo de Propiedades" (Property Graph) como en la herramienta Cypher / Neo4j, ¿qué elementos básicos constituyen una "arista" (edge)?***
- A) Una tupla anidada sin relaciones jerárquicas predecibles.
 - B) Un identificador, un vértice inicial (tail), un vértice final (head), una etiqueta relacional y propiedades clave-valor.
 - C) Un sujeto, predicado y objeto (triple-store).
 - D) Colecciones de documentos JSON interconectados implícitamente por referencias temporales.
76. ***El mecanismo de optimizar problemas matemáticos en un grafo almacenando los resultados intermedios para subproblemas más pequeños (evitando recálculos) es una característica definitoria de:***
- A) Encadenamiento hacia adelante simple.
 - B) Búsqueda en Profundidad Recursiva.
 - C) Algoritmo Monte Carlo.
 - D) Programación Dinámica (Dynamic Programming).
77. ***¿Qué representa técnicamente el "Épsilon de la máquina" ϵ en aritmética de coma flotante?***
- A) El valor absoluto de la diferencia entre el número real más grande y el más pequeño.
 - B) El error acumulado máximo permitido en una suma vectorial de precisión simple.
 - C) El valor más pequeño tal que la operación $1.0 + \epsilon > 1.0$ sea verdadera.
 - D) El límite inferior de underflow donde el hardware redondea cualquier valor a cero.
78. ***¿Por qué el redondeo al "par más cercano" (Ties-to-Even) es el estándar en IEEE 754?***
- A) Porque es el método que consume menos ciclos de reloj en la ALU.
 - B) Para reducir el sesgo estadístico acumulado en operaciones repetitivas.
 - C) Porque permite representar números infinitos sin pérdida de precisión.
 - D) Para asegurar que la resta de dos números iguales sea siempre cero.
79. ***¿Cuál es la causa técnica de la "Cancelación Catastrófica" en cálculo numérico?***
- A) El desbordamiento del exponente durante la multiplicación de matrices densas.
 - B) La división de un número real por una constante épsilon cercana a cero.
 - C) La resta de dos números casi idénticos con errores de redondeo previos.
 - D) Ocurre cuando un número pequeño se redondea automáticamente a cero por underflow.



80. *¿Cuál es la complejidad algorítmica de la FFT para una señal de tamaño N ?*

- A) $O(N \log N)$
- B) $O(N^2)$
- C) $O(N)$
- D) $O(e^N)$

81. *¿Qué fenómeno describe el "Aliasing" en el contexto de la FFT?*

- A) La pérdida de amplitud de la señal debido al uso de funciones de ventana.
- B) El solapamiento de frecuencias cuando la tasa de muestreo es inferior al doble de la frecuencia máxima.
- C) La reordenación incorrecta de los bits en la etapa de inversión de bits.
- D) La aparición de armónicos falsos debido al redondeo de punto flotante.

82. *¿Qué efecto tiene multiplicar punto a punto en frecuencia mediante FFT?*

- A) Se obtiene una convolución lineal perfecta sin necesidad de técnicas de padding.
- B) Se genera una convolución circular que requiere zero-padding para ser lineal.
- C) Se produce una amplificación del error de redondeo proporcional al cuadrado de N .
- D) Se reduce el rango dinámico de la señal original por el escalado de coeficientes.

83. *¿Cuál es la tasa de convergencia del error en un método de Monte Carlo estándar?*

- A) A) $1/N^2$
- B) B) $1/N$
- C) C) $1/\log N$
- D) D) $1/\sqrt{N}$

84. *¿Cuál es la función técnica de un "Muestreo Estratificado" en Monte Carlo?*

- A) Dividir el espacio de búsqueda en subregiones para asegurar una cobertura más uniforme.
- B) Utilizar únicamente números primos para generar las semillas aleatorias.
- C) Eliminar las muestras que se alejan de la media mediante un filtro de outlier.
- D) Aumentar el número de muestras en las zonas donde la función es cero.



85. *¿Qué problema presentan los PRNG lineales congruenciales en dimensiones altas?*
- A) Un consumo excesivo de memoria caché L1 que impide el paralelismo masivo.
 - B) La tendencia de los puntos a alinearse en hiperplanos (Teorema de Marsaglia).
 - C) Una latencia de acceso superior a la de los generadores tipo LFSR.
 - D) La imposibilidad de ser inicializados con una semilla distinta de cero.
86. *¿Qué técnica permite estimar la incertidumbre del modelo comparando resultados numéricos con datos experimentales?*
- A) Análisis de sensibilidad.
 - B) Verificación del código.
 - C) Validación del modelo.
 - D) Perfilado de memoria.
87. *¿Cuál es la función del "Back-to-back testing" en algoritmos científicos complejos?*
- A) Comparar dos versiones del mismo código con los mismos inputs.
 - B) Ejecutar el código en orden inverso para detectar ciclos.
 - C) Validar el software usando únicamente personal no experto.
 - D) Duplicar el hardware para evitar fallos de hardware aleatorios.
88. *En Matplotlib, si deseamos analizar y graficar distribuciones cruzadas y su densidad utilizando la técnica gráfica hexagonal binning como una alternativa a las teselas cuadradas (histogramas 2D), ¿qué método invocamos?*
- A) plt.hist2d
 - B) plt.scatter
 - C) plt.hexbin
 - D) np.histogramdd
89. *¿A qué se refiere la técnica conocida como "Estimación de Densidad del Kernel" (KDE) usada comúnmente en la visualización científica de distribuciones?*
- A) Un clasificador discriminativo de aprendizaje automático para imágenes.
 - B) Un algoritmo para dividir el gráfico 3D en mallas poligonales.
 - C) Un algoritmo estadístico diseñado para "difuminar" (smear out) puntos y sumar sus aportes para estimar una curva o función continua suave.
 - D) Un método que agrupa valores categóricos explícitamente en una jerarquía 2D.



90. *En los flujos de Python científico, ¿qué biblioteca estadística proporciona utilidades de alto nivel como pairplot, que genera instantáneamente una matriz de gráficos de dispersión cruzando todos los atributos de un dataset?*
- A) Scikit-Learn
 - B) Matplotlib
 - C) Pandas
 - D) Seaborn
91. *Cuando se realizan análisis de datos altamente multidimensionales en Pandas y requerimos presentar variables de agrupación superpuestas en celdas de una cuadrícula tabulada para observar interacciones (como un mapa de calor textual), ¿qué herramienta de la API abstrae esta operación eficientemente?*
- A) apply()
 - B) pivot_table
 - C) np.histogram2d
 - D) eval()
92. *Para evaluar las expresiones algebraicas compuestas muy pesadas (vectores enormes), ¿qué módulo de procesamiento se aconseja para maximizar la localidad del caché y no sobrecargar la RAM con variables temporales redundantes?*
- A) scipy.special
 - B) numexpr
 - C) sklearn.manifold
 - D) plt.contourf
93. *¿Qué representación estándar se usa en gráficos por computadora para modelar objetos 3D y calcular visibilidad y oclusiones?*
- A) B-trees de precisión infinita.
 - B) Arrays Numpy densos (Dense ndarrays).
 - C) Mallas poligonales (Polygonal meshes).
 - D) Modelos jerárquicos estáticos.



94. *¿Por qué se desaconseja crear iteraciones ("loops") explícitas en Python sobre los píxeles de una imagen, y se sugiere el uso matricial estricto (por ejemplo, con NumPy)*
- A) Porque los objetos de Python pueden fallar si la imagen contiene ruido.
 - B) Porque las operaciones implícitas en matrices (NumPy array-based arithmetic) son significativamente más rápidas y mapean directamente a bibliotecas C++ o motores de Tensores Profundos (Deep Learning) subyacentes.
 - C) Porque las API imperativas no pueden acceder a canales de color (RGB).
 - D) Porque una iteración explícita no puede guardar en disco posteriormente.
95. *En problemas de vecinos más cercanos o en el reconocimiento de atributos y similitud entre imágenes, ¿qué estrategia adopta el mecanismo Best Bin First (BBF) en su árbol de búsqueda?*
- A) Busca secuencias por coincidencias booleanas estáticas exactas en todos los contenedores a la vez.
 - B) Examina los contenedores (bins) en orden según su proximidad espacial geométrica frente al punto de la consulta.
 - C) Proyecta las distancias en una sola dimensión continua.
 - D) Utiliza análisis léxico para asignar variables.
96. *En tecnologías de mapeo e indexado de algoritmos descriptores en alta dimensionalidad (como los utilizados en segmentación), ¿qué enfoque se encarga de convertir y comparar datos en función de códigos binarios calculando Distancias de Hamming?*
- A) Hashing paramétrico / hashing sensible a localidad (Parameter-sensitive hashing / LSH).
 - B) Transformada de Fourier Discreta 1D.
 - C) Árboles B generativos.
 - D) Grafos Acíclicos Dirigidos Cíclicos.
97. *¿Qué biblioteca aprovecha estrictamente lazy-evaluation, compitiendo con el diseño original de Pandas en informes visuales?*
- A) Vaex
 - B) SQLite
 - C) Flask
 - D) SQLAlchemy



98. *En el desarrollo front-end, ¿cuál es la principal ventaja de utilizar hojas de estilo (CSS) según las fuentes?*
- A) Permiten estructurar semánticamente el contenido del documento.
 - B) Facilitan la separación entre los contenidos de la página (HTML) y su aspecto o formato.
 - C) Son necesarias para que el navegador pueda interpretar las etiquetas de hipertexto.
 - D) Permiten que el servidor web procese los datos de los formularios más rápido.
99. *En la declaración de un formulario, ¿qué diferencia existe entre los métodos de envío GET y POST?*
- A) GET no tiene restricciones de longitud, mientras que POST está limitado a 100 caracteres.
 - B) En el método POST, los valores de las variables son visibles en la URL del navegador.
 - C) POST permite el envío de archivos y carece de restricciones de longitud, a diferencia de GET.
 - D) GET es el único método que permite combinar formularios con JavaScript.
100. *Al usar la extensión Flask-SQLAlchemy para desarrollar el backend de una aplicación científica en Python, ¿cuál es un requisito forzoso al definir todas las clases de un Modelo (Model)?*
- A) Instanciar un servidor proxy en el método constructor `__init__`.
 - B) Definir una columna designada explícitamente como "Clave Primaria" (Primary Key), comúnmente llamada id.
 - C) Escribir en bruto sentencias SELECT de SQL.
 - D) Incluir una referencia json en los métodos estáticos.



PREGUNTAS DE RESERVA

101. *Al diseñar el desarrollo "front-end" de una plataforma, ¿qué patrón de comportamiento se utiliza habitualmente en interfaces de usuario y Model-View-Controller (MVC) para que la vista reciba actualizaciones en tiempo real si cambian los datos subyacentes?*
- A) Patrón Intérprete.
 - B) Patrón Cadena de Responsabilidad.
 - C) Patrón Observador (Observer Pattern).
 - D) Patrón Plantilla.
102. *Para implementar capacidades como historiales de interacción y acciones de "Deshacer/Rehacer" (Undo) dictadas por el usuario en una interfaz interactiva de aplicación, ¿qué patrón estructural es el adecuado?*
- A) Patrón Constructor (Builder).
 - B) Patrón Orden / Comando (Command).
 - C) Patrón Peso Ligero (Flyweight).
 - D) Patrón Mediador (Mediator).
103. *Al momento de conectarse mediante scripts analíticos (ej. R / dbplyr) a motores externos de Bases de Datos, ¿cuál de las siguientes opciones describe el paquete de interfaz genérica (DBI)?*
- A) Es un traductor de Pandas diseñado para compilar librerías locales.
 - B) Es una interfaz unificada y de bajo nivel que provee comandos genéricos para conectar con bases y gestionar consultas SQL sin importar el software DBMS externo.
 - C) Es un driver de base de datos específico para SQLite instalado en memoria de solo lectura.
 - D) Es una plataforma basada en la nube de Amazon orientada al procesamiento NoSQL.



104. *¿En qué consiste conceptualmente la transferencia en servicios REST (Representational State Transfer)?*

- A) En mantener un socket de estado continuo encriptado durante sesiones ininterrumpidas.
- B) En orquestar mensajes asincrónicos entre bases de datos compartiendo la memoria.
- C) En intercambiar de datos a través de servicios utilizando solicitudes HTTP sin estado (stateless) para representar el acceso o cambio en los recursos.
- D) En empaquetar protocolos XML obligatorios dentro de bases documentales para compilar la infraestructura del dominio de aplicaciones.

105. *Para estructurar respuestas e intercambio de datos ligeros en la Web y en APIs, ¿qué formato textual ha reemplazado predominante al XML?*

- A) MapReduce
- B) Protocol Buffers
- C) JSON
- D) Thrift

*******FIN DEL CUESTIONARIO*******