

Comenzado el	domingo, 29 de octubre de 2023, 19:05
Estado	Finalizado
Finalizado en	domingo, 29 de octubre de 2023, 19:13
Tiempo empleado	8 minutos 8 segundos
Puntos	20/20
Calificación	10 de 10 (100%)

Pregunta **1**

Correcta

Se puntúa 1 sobre 1

Para cada uno de los algoritmos o procesos listados en la columna de la izquierda, seleccione la expresión de orden que mejor describe su tiempo de ejecución en el peor caso.

Dado un arreglo ya ordenado v con n componentes, buscar un valor x aplicando búsqueda binaria.

$O(\log(n))$



Dado un arreglo v con n componentes, acceder y cambiar el valor del casillero $v[k]$ (con $0 \leq k \leq n-1$).

$O(1)$



Dado un arreglo v con n componentes, buscar un valor x aplicando búsqueda secuencial.

$O(n)$



Dadas dos matrices de orden $n \times n$, obtener la matriz producto aplicando el método tradicional.

$O(n^3)$ (léase: "orden n al cubo")



Dado un arreglo v con n componentes, ordenarlo de menor a mayor mediante el algoritmo de selección directa.

$O(n^2)$ (léase: "orden n al cuadrado")



Pregunta **2**

Correcta

Se puntúa 1 sobre 1

¿Cuántas comparaciones en el **peor caso** obliga a hacer una búsqueda secuencial en una *lista ordenada* (o en un *arreglo ordenado*) que contenga n valores?

Seleccione una:

- ☐ a. Peor caso: $O(n^2)$ comparaciones.
- ☐ b. Peor caso: $O(1)$ comparaciones.
- ☒ c. Peor caso: $O(n)$ comparaciones.
- ☐ d. Peor caso: $O(\log(n))$ comparaciones.

Pregunta **3**

Correcta

Se puntúa 1 sobre 1

¿Cuál es la diferencia entre el *peor caso* y el *caso promedio* en el análisis de algoritmos?

Seleccione una:

- ☐ a. El *peor caso* es la configuración de datos de entrada más favorable para el algoritmo, mientras que el *caso promedio* describe una configuración de datos pensada para desfavorecer al algoritmo.
- ☐ b. El *peor caso* es la configuración de datos de entrada más desfavorable para el algoritmo, mientras que el *caso promedio* describe una configuración de datos pensada para favorecer al algoritmo.
- ☐ c. El *peor caso* es la configuración de datos de entrada más favorable para el algoritmo, mientras que el *caso promedio* describe una configuración aleatoria de datos (no pensada ni para favorecer ni para desfavorecer al algoritmo)
- ☒ d. El *peor caso* es la configuración de datos de entrada más desfavorable para el algoritmo, mientras que el *caso promedio* describe una configuración aleatoria de datos (no pensada ni para favorecer ni para desfavorecer al algoritmo). ✓

Pregunta **4**

Correcta

Se puntúa 1 sobre 1

¿Cuáles de los siguientes son *factores de eficiencia comunes* a considerar en el análisis de algoritmos? (Más de una respuesta puede ser válida... marque *todas* las que considere correctas).

Seleccione una o más de una:

- ☐ a. La calidad aparente de la interfaz de usuario.
- ☒ b. El tiempo de ejecución. ✓
- ☒ c. La complejidad aparente del código fuente. ✓
- ☒ d. El consumo de memoria. ✓

Pregunta **5**

Correcta

Se puntúa 1 sobre 1

¿Qué significa decir que un algoritmo dado tiene un tiempo de ejecución $O(1)$?

Seleccione una:

- ☐ a. El tiempo de ejecución es lineal: si aumenta el número de datos, aumenta el tiempo en la misma proporción.
- ☐ b. El tiempo de ejecución es logarítmico: a medida que aumenta el número de datos, aumenta el tiempo pero en forma muy suave.
- ☒ c. El tiempo de ejecución es constante, sin importar la cantidad de datos. ✓
- ☐ d. El tiempo de ejecución siempre es de un segundo, sin importar la cantidad de datos.


Pregunta **6**

Correcta

Se puntúa 1 sobre 1

¿Qué significa decir que un algoritmo dado tiene un tiempo de ejecución $O(n^2)$?

Seleccione una:

- ☐ a. El tiempo de ejecución es lineal: si aumenta el número de datos, aumenta el tiempo en la misma proporción.
- ☒ b. El proceso normalmente consiste en dos ciclos (uno dentro del otro) de aproximadamente n iteraciones cada uno, de forma  que las operaciones críticas se aplican un número cuadrático de veces.
- ☐ c. A medida que aumenta el número de datos, aumenta el tiempo pero en forma muy suave: el conjunto de datos se divide en dos. se procesa una de las mitades, se desecha la otra y se repite el proceso hasta que no pueda volver a dividirse la mitad que haya quedado.
- ☐ d. El tiempo de ejecución es constante, sin importar la cantidad de datos.


Pregunta **7**

Correcta

Se puntúa 1 sobre 1

¿Qué significa decir que un algoritmo dado tiene un tiempo de ejecución $O(n)$?

Seleccione una:

- ☐ a. El tiempo de ejecución es constante, sin importar la cantidad de datos.
- ☐ b. A medida que aumenta el número de datos, aumenta el tiempo pero en forma muy suave: el conjunto de datos se divide en dos. se procesa una de las mitades, se desecha la otra y se repite el proceso hasta que no pueda volver a dividirse la mitad que haya quedado.
- ☒ c. El tiempo de ejecución es lineal: si aumenta el número de datos, aumenta el tiempo en la misma proporción. 
- ☐ d. El proceso normalmente consiste en dos ciclos (uno dentro del otro) de aproximadamente n iteraciones cada uno, de forma que las operaciones críticas se aplican un número cuadrático de veces.


Pregunta **8**

Correcta

Se puntúa 1 sobre 1

¿Qué significa decir que un algoritmo dado tiene un tiempo de ejecución $O(\log(n))$?

Seleccione una:

- ☒ a. A medida que aumenta el número de datos, aumenta el tiempo pero en forma muy suave: el conjunto de datos se divide en  dos. se procesa una de las mitades, se desecha la otra y se repite el proceso hasta que no pueda volver a dividirse la mitad que haya quedado.
- ☐ b. El tiempo de ejecución es constante, sin importar la cantidad de datos.
- ☐ c. El proceso normalmente consiste en dos ciclos (uno dentro del otro) de aproximadamente n iteraciones cada uno, de forma que las operaciones críticas se aplican un número cuadrático de veces.
- ☐ d. El tiempo de ejecución es lineal: si aumenta el número de datos, aumenta el tiempo en la misma proporción.

Pregunta **9**

Correcta

Se puntúa 1 sobre 1

Suponga que dispone de cuatro algoritmos diferentes para resolver el mismo problema, y que se sabe que los tiempos de ejecución (en el peor caso) son, respectivamente: **$O(n \cdot \log(n))$** , **$O(n^2)$** , **$O(n^3)$** y **$O(n)$** .

¿Cuál de esos tres algoritmos debería elegir, suponiendo que todos hacen el mismo consumo razonable de memoria?

Seleccione una:

- ☒ a. El algoritmo cuyo tiempo de ejecución es $O(n)$ ✓
- ☐ b. El algoritmo cuyo tiempo de ejecución es $O(n \cdot \log(n))$
- ☐ c. El algoritmo cuyo tiempo de ejecución es $O(n^3)$
- ☐ d. El algoritmo cuyo tiempo de ejecución es $O(n^2)$

Pregunta **10**

Correcta

Se puntúa 1 sobre 1

¿Con qué nombre general se conoce en la *Teoría de la Complejidad* a un problema para el cual sólo se conocen algoritmos cuyo tiempo de ejecución es exponencial (o sea, problemas para los que todas las soluciones conocidas son algoritmos con tiempo $O(2^n)$)?

Seleccione una:

- ☐ a. Problemas Irresolubles
- ☒ b. Problemas Intratables ✓
- ☐ c. Problemas Inmanejables
- ☐ d. Problemas Imperdonables

Pregunta **11**

Correcta

Se puntúa 1 sobre 1

En la columna de la izquierda se muestra el código fuente en Python de diversos procesos sencillos. Para cada uno de ellos, seleccione de la lista de la derecha la expresión en notación *Big O* que mejor describa el tiempo de ejecución de cada proceso para el peor caso. (Aclaración: una expresión como n^2 debe entenderse como "n al cuadrado" o n^2).

```
n = int(input('N: '))
ac = 0
for i in range(n):
    for j in range(i+1, n):
        ac += i*j

for i in range(n):
    for j in range(n):
        for k in range(n):
            ac += (i+j+k)
print(ac)
```

 $O(n^3)$ ✓

```
n = int(input('N: '))
ac = 0
for i in range(n):
    for j in range(i+1, n):
        ac += i*j
print(ac)
```

 $O(n^2)$ ✓

```
n = int(input('N: '))
ac = 0
for i in range(n):
    ac += i
print(ac)
```

 $O(n)$ ✓

```
ac = 0
for i in range(10):
    ac += i
print(ac)
```

 $O(1)$ ✓Pregunta **12**

Correcta

Se puntúa 1 sobre 1

Analice el siguiente esquema de una función en Python:

```
def procesar(n, m):
    for i in range(n+1):
        for j in range(m+1):
            # ... acciones sencillas a realizar...
            # ... suponga que no hay otro ciclo aquí...
            # ... y que sólo aparecen operaciones de tiempo constante...
```

¿Cuál de las siguientes expresiones de orden describe mejor el tiempo de ejecución de esta función en el peor caso?

Seleccione una:

- ☐ a. $O(n*n)$
- ☒ b. $O(n*m)$ ✓
- ☐ c. $O(n)$
- ☐ d. $O(m)$

Pregunta **13**

Correcta

Se puntúa 1 sobre 1

Si se realiza un análisis preciso del ordenamiento por *Selección Directa* para un arreglo de n componentes, se llega a la conclusión que ese algoritmo hará $n-1$ pasadas, con $n-1$ comparaciones en la primera, $n-2$ en la segunda, y así sucesivamente reduciendo de a 1 la cantidad de comparaciones hasta hacer sólo una comparación en la última pasada. Por lo tanto, el algoritmo hará *invariablemente* una cantidad total de $\frac{1}{2}(n^2 - n)$ comparaciones. Sabiendo esto, ¿cuáles de las siguientes expresiones son correctas para describir la cantidad de comparaciones que hará el algoritmo, usando distintos tipos de notaciones? (Más de una respuesta puede ser correcta. Marque TODAS las que considere correctas)

Seleccione una o más de una:

- ☐ a. Cantidad de comparaciones: $o(n^2)$
- ☒ b. Cantidad de comparaciones: $\Omega(n^2)$ ✓
- ☒ c. Cantidad de comparaciones: $O(n^2)$ ✓
- ☒ d. Cantidad de comparaciones: $\Theta(n^2)$ ✓

Pregunta **14**

Correcta

Se puntúa 1 sobre 1

¿Cuál es la **principal** característica de todos los métodos de ordenamiento conocidos como **métodos simples** o **directos**?

Seleccione una:

- ☐ a. Son muy fáciles de programar.
- ☐ b. Son muy veloces para cualquier tamaño del arreglo a ordenar.
- ☒ c. Tienen un tiempo de ejecución de orden cuadrático en el peor caso. ✓
- ☐ d. Tienen un tiempo de ejecución de orden lineal en el peor caso.

Pregunta **15**

Correcta

Se puntúa 1 sobre 1

¿Cuáles de las siguientes son características **correctas** del algoritmo *Shellsort*? (Más de una puede ser cierta... marque TODAS las que considere válidas)

Seleccione una o más de una:

- ☐ a. El algoritmo *Shellsort* consiste en una mejora del algoritmo de *Selección Directa*, consistente en buscar iterativamente el menor (o el mayor) entre los elementos que quedan en el vector, para llevarlo a su posición correcta, pero de forma que la búsqueda del menor en cada vuelta se haga en tiempo logarítmico.
- ☒ b. El algoritmo *Shellsort* es complejo de analizar para determinar su rendimientos en forma matemática. Se sabe que para la serie de incrementos decrecientes usada en la implementación vista en las clases de la asignatura, tiene un tiempo de ejecución para el peor caso de $O(n^{1.5})$. ✓
- ☒ c. El algoritmo *Shellsort* consiste en una mejora del algoritmo de *Inserción Directa* (o *Inserción Simple*), consistente en armar suconjuntos ordenados con elementos a distancia $h > 1$ en las primeras fases, y terminar con $h = 1$ en la última. ✓
- ☐ d. En el caso promedio, el algoritmo *Shellsort* es tan eficiente como el *Heapsort* o el *Quicksort*, con tiempo de ejecución $O(n \log(n))$.

Pregunta **16**

Correcta

Se puntúa 1 sobre 1

¿Qué diferencia existe entre el *conteo exhaustivo* de operaciones críticas y el *análisis asintótico* del comportamiento de una función en el análisis de algoritmos?

Seleccione una:

- ☐ a. Ninguna. Ambas se refieren a la misma técnica básica del análisis de algoritmos
- ☒ b. El *conteo exhaustivo* busca determinar una expresión o fórmula que exprese de manera rigurosa la cantidad de operaciones críticas que lleva a cabo un algoritmo, mientras que el *análisis asintótico* busca determinar el comportamiento general de una función para valores muy grandes del tamaño del problema. ✓
- ☐ c. El *análisis asintótico* busca determinar una expresión o fórmula que exprese de manera rigurosa la cantidad de operaciones críticas que lleva a cabo un algoritmo, mientras que el *conteo exhaustivo* busca determinar el comportamiento general de una función para valores muy grandes del tamaño del problema.
- ☐ d. El *conteo exhaustivo* busca determinar una expresión o fórmula que exprese de manera rigurosa la cantidad de operaciones **críticas** que lleva a cabo un algoritmo, mientras que el *análisis asintótico* busca determinar una expresión o fórmula que exprese de manera rigurosa la cantidad de operaciones **no críticas** que lleva a cabo un algoritmo.

Pregunta **17**

Correcta

Se puntúa 1 sobre 1

¿Qué se entiende, en el contexto del Análisis de Algoritmos, por un *Orden de Complejidad*?

Seleccione una:

- ☒ a. Un conjunto o familia de funciones matemáticas que se comportan asintóticamente de la misma forma. ✓
- ☐ b. Un conjunto o familia de algoritmos que resuelven el mismo problema.
- ☐ c. Un conjunto o familia de subrutinas con similares objetivos (equivalente al concepto de *módulo*).
- ☐ d. Un conjunto de datos ordenados.

Pregunta **18**

Correcta

Se puntúa 1 sobre 1

Si los algoritmos de *ordenamiento simples* tienen todos un tiempo de ejecución $O(n^2)$ en el peor caso, entonces: ¿cómo explica que las mediciones efectivas de los tiempos de ejecución de cada uno sean diferentes frente al mismo arreglo?

Seleccione una:

- ☒ a. La notación *Big O* rescata el término más significativo en la expresión que calcula el rendimiento, descartando constantes y ✓ otros términos que podrían no coincidir en los tres algoritmos.
- ☐ b. Los tiempos deben coincidir. Si hay diferencias, se debe a errores en los instrumentos de medición o a un planteo incorrecto del proceso de medición.
- ☐ c. La notación *Big O* no se debe usar para estimar el comportamiento en el peor caso, sino sólo para el caso medio.
- ☐ d. La notación *Big O* no se usa para medir tiempos sino para contar comparaciones u otro elemento de interés. Es un error, entonces, decir que los tiempos tienen "*orden n cuadrado*".

Pregunta **19**

Correcta

Se puntúa 1 sobre 1

Se tiene un algoritmo que realiza cierta cantidad de procesos sobre un conjunto de n datos y un minucioso análisis matemático ha determinado que la cantidad de procesos que el algoritmo realiza en el peor caso viene descrito por la función $f(n) = 3n^3 + 5n^2 + 2n^{1.5}$. ¿Cuál de las siguientes expresiones representa mejor el orden del algoritmo para el peor caso?

Seleccione una:

- ☐ a. $O(3n^3 + 5n^2 + 2n^{1.5})$
- ☐ b. $O(n^{1.5})$
- ☐ c. $O(n^3 + n^2)$
- ☒ d. $O(n^3)$ ✓

Pregunta **20**

Correcta

Se puntúa 1 sobre 1

¿Cuáles de las siguientes *son correctas* en cuanto a los tiempos de ejecución de los algoritmos de ordenamiento clásicos? (Más de una puede ser cierta... marque TODAS las que considere válidas)

Seleccione una o más de una:

- ☐ a. Algoritmo Shell Sort: $O(n^2)$ en el peor caso para la serie de incrementos decrecientes vista en clase.
- ☒ b. Algoritmo Heap Sort: $O(n \log(n))$ tanto para el caso promedio como para el peor caso. ✓
- ☒ c. Algoritmos directos o simples: $O(n^2)$ en el peor caso para todos ellos. ✓
- ☒ d. Algoritmo Quick Sort: $O(n \log(n))$ en el caso promedio, pero $O(n^2)$ en el peor caso. ✓

[◀ Materiales Adicionales para la Ficha 27](#)

Ir a...



Parcial 4 ►