

Comenzado el	domingo, 1 de octubre de 2023, 17:02
Estado	Finalizado
Finalizado en	domingo, 1 de octubre de 2023, 17:06
Tiempo empleado	3 minutos 20 segundos
Calificación	10 de 10 (100%)

Pregunta **1**

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. Tienen muy mal rendimiento en tiempo de ejecución si el tamaño n del arreglo es grande o muy grande, y un rendimiento aceptable si n es pequeño. ✓
- ☐ b. Tienen muy mal rendimiento en tiempo de ejecución, cualquiera sea el tamaño n del arreglo.
- ☐ c. Tienen muy mal rendimiento en tiempo de ejecución si el tamaño n del arreglo es pequeño, y un rendimiento aceptable si n es grande o muy grande.
- ☐ d. Tienen muy buen rendimiento en tiempo de ejecución, cualquiera sea el tamaño n del arreglo.

Pregunta **2**

Correcta

Se puntúa 1 sobre 1

¿Cuál de las siguientes describe mejor la idea de funcionamiento en la que está basado el algoritmo conocido como ordenamiento por *Inserción Simple* para ordenar un arreglo de n componentes?

Seleccione una:

- ☐ a. Realizar n pasadas, de forma que en cada una se compare a cada elemento con el siguiente, logrando que en cada pasada los mayores vaya acomodándose al final del arreglo.
- ☐ b. Reacomodar los n elementos del arreglo en forma aleatoria, controlar si quedó ordenado, y en caso de negativo, volver a reacomodarlos en forma aleatoria, continuando así hasta que en algún momento se obtenga un arreglo ordenado...
- ☐ c. Realizar n pasadas, de forma que en cada una se determine el menor de los elementos analizados, y llevar ese menor a la casilla pivot.
- ☒ d. Suponer que el arreglo tiene un subconjunto inicialmente ordenado que contiene sólo al primer elemento, luego realizar n pasadas, de forma que en cada una agregue el siguiente elemento al grupo que está ordenado. ✓


Pregunta **3**

Correcta

Se puntúa 1 sobre 1

¿Cuál de las siguientes describe mejor la idea de funcionamiento en la que está basado el algoritmo conocido como ordenamiento por *Intercambio Directo* u *Ordenamiento de Burbuja* para ordenar *de menor a mayor* un arreglo de n componentes?

Seleccione una:

- ☒ a. Realizar n pasadas, de forma que en cada una se compare a cada elemento con el siguiente, logrando que en cada pasada los  mayores vayan acomodándose al final del arreglo.
- ☐ b. Realizar n pasadas, de forma que en cada una se determine el menor de los elementos analizados, y llevar ese menor a la casilla pivot.
- ☐ c. Suponer que el arreglo tiene un subconjunto inicialmente ordenado que contiene sólo al primer elemento, luego realizar n pasadas, de forma que en cada una agregue el siguiente elemento al grupo que está ordenado.
- ☐ d. Reacomodar los n elementos del arreglo en forma aleatoria, controlar si quedó ordenado, y en caso negativo, volver a reacomodarlos en forma aleatoria, continuando así hasta que en algún momento se obtenga un arreglo ordenado...


Pregunta **4**

Correcta

Se puntúa 1 sobre 1

¿Cuál de las siguientes describe mejor la idea de funcionamiento en la que está basado el algoritmo conocido como ordenamiento por *Selección Simple* o *Selección Directa* para ordenar un arreglo de n componentes?

Seleccione una:

- ☐ a. Realizar n pasadas, de forma que en cada una se compare a cada elemento con el siguiente, logrando que en cada pasada los mayores vaya acomodándose al final del arreglo.
- ☒ b. Realizar n pasadas, de forma que en cada una se determine el menor de los elementos analizados, y llevar ese menor a la casilla  pivot.
- ☐ c. Reacomodar los n elementos del arreglo en forma aleatoria, controlar si quedó ordenado, y en caso de negativo, volver a reacomodarlos en forma aleatoria, continuando así hasta que en algún momento se obtenga un arreglo ordenado...
- ☐ d. Suponer que el arreglo tiene un subconjunto inicialmente ordenado que contiene sólo al primer elemento, luego realizar n pasadas, de forma que en cada una agregue el siguiente elemento al grupo que está ordenado.


Pregunta **5**

Correcta

Se puntúa 1 sobre 1

¿Por qué motivo el algoritmo Bubblesort para ordenamiento de un arreglo usa una *bandera de corte* en la versión presentada en las fichas de clase?

Seleccione una:

- ☒ a. La bandera de corte se usa para terminar el proceso apenas se detecte que en la pasada actual no hubo intercambios, para  ahorrar tiempo.
- ☐ b. La bandera de corte se usa para determinar si el ordenamiento debe hacerse de menor a mayor (bandera = True) o de mayor a menor (bandera = False)
- ☐ c. No es cierto que la versión vista en clases use una bandera de corte.
- ☐ d. La bandera de corte se usa para garantizar que el arreglo quede ordenado.

Pregunta **6**

Correcta

Se puntúa 1 sobre 1

¿Cuál de los siguientes es el creador del famoso algoritmo de ordenamiento conocido como **Quicksort**?

Seleccione una:

- ☒ a. Charles Antony Richard Hoare ✓
- ☐ b. Donald Shell
- ☐ c. J. W. J. Williams
- ☐ d. Edsger Wybe Dijkstra

Pregunta **7**

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. Una muy buena serie de incrementos decrecientes a usar, es la serie $h = \{...16, 8, 4, 2, 1\}$
- ☐ b. El algoritmo Shellsort consiste en una mejora del algoritmo de Selección Directa, basada 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.
- ☒ c. El algoritmo Shellsort consiste en una mejora del algoritmo de Inserción Directa (o Inserción Simple), consistente en armar subconjuntos ordenados con elementos a distancia $h > 1$ en las primeras fases, y terminar con $h = 1$ en la última. ✓
- ☒ d. El algoritmo Shellsort es complejo de analizar para determinar su rendimiento en forma matemática, ya que ese rendimiento depende fuertemente de la serie de incrementos decreciente que se haya seleccionado. ✓

Pregunta **8**

Correcta

Se puntúa 1 sobre 1

¿Cuál es el problema si en el algoritmo *Shellsort* se elige una serie de incrementos decrecientes de la forma $\{..., 16, 8, 4, 2, 1\}$?

Seleccione una:

- ☐ a. El arreglo no quedará ordenado al final.
- ☐ b. Ningún problema: esa serie es tan buena como cualquier otra.
- ☒ c. Los subconjuntos analizados contendrán casi los mismos elementos cuando la distancia usada sea cada vez menor, sin garantías de lograr una buena organización del arreglo antes de la última pasada. ✓
- ☐ d. No sólo no hay ningún problema, sino que esa serie es la mejor posible para el algoritmo Shellsort.

Pregunta **9**

Correcta

Se puntúa 1 sobre 1

Sabemos que en el *algoritmo de Shell* se termina haciendo una última pasada sobre el arreglo con incremento de comparación **$h = 1$** ¿Cuál de las siguientes es cierta respecto de esa última pasada con **$h = 1$** ?

Seleccione una:

- ☐ a. Con $h = 1$ el algoritmo sólo controla si el arreglo está ya ordenado, y en caso de no estarlo relanza el proceso con otra sucesión de valores h .
- ☒ b. Con $h = 1$ el algoritmo se convierte en un ordenamiento por inserción simple, y sólo entonces garantiza que el arreglo quede **✓** ordenado.
- ☐ c. No es obligatorio que lo haga, pero favorece un ordenamiento más rápido.
- ☐ d. La pasada con $h = 1$ es obligatoria pero no es necesario que sea la última.

Pregunta **10**

Correcta

Se puntúa 1 sobre 1

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

Seleccione una o más de una:

- ☒ a. El algoritmo Heapsort se basa en encontrar sucesivamente el menor (o el mayor) de entre los elementos que quedan, para llevar **✓** ese valor a su casillero final, pero de forma que la búsqueda del menor (o el mayor) en cada vuelta se haga en forma muy veloz.
- ☒ b. El algoritmo Heapsort es muy eficiente en tiempo de ejecución, tanto para el caso promedio como para el peor caso. **✓**
- ☐ c. El algoritmo Heapsort utiliza una cantidad de memoria adicional igual al tamaño del arreglo, para armar el heap o grupo de ordenamiento con el que se ordena el vector.
- ☒ d. El algoritmo Heapsort arma el heap o grupo de ordenamiento con el que se ordena el vector, pero lo hace en el mismo vector, **✓** sin usar memoria extra.

[◀ Materiales Adicionales para la Ficha 22](#)

Ir a...



[Parcial 3 ▶](#)