1. ¿Qué significa el término completo cuando se aplica a árboles binarios?

a. Se han insertado todos los elementos de datos necesarios.

b. Todas las filas están llenas de nodos, excepto posiblemente la última.

c. Todos los nodos existentes contienen datos.

d. La disposición de los nodos satisface la condición de montón.

1. ¿Qué significa el término parcialmente ordenado cuando se aplica a montones?
2. Cuando se elimina un elemento de un montón, siempre se elimina de

la \_\_\_\_\_\_\_\_\_\_.

1. Cuando se inserta un elemento en un montón,

a. se introduce un agujero en el nodo raíz y se tamiza hasta que

alcanza la posición que debería ocupar el elemento.

b. se inicia una búsqueda desde la raíz para encontrar el elemento con la clave igual

o justo por encima de la clave a insertar, y el elemento se inserta como hijo de ese

nodo.

c. el elemento máximo en el montón se mueve a la primera celda vacía,

y el nuevo elemento se inserta y se tamiza hasta que alcanza la posición que debería ocupar.

d. el elemento se inserta en la primera celda vacía y luego se tamiza hasta

que alcanza la posición que debería ocupar.

1. Un montón se puede representar mediante una matriz porque un montón

a. es un árbol binario.

b. está parcialmente completo.

c. está parcialmente ordenado.

d. satisface la condición de montón.

1. El último nodo de un montón es

a. siempre un hijo izquierdo.

b. siempre un hijo derecho.

c. siempre en la fila inferior.

d. nunca menor que su hermano.

1. Un montón es a una cola de prioridad como un(a) \_\_\_\_\_\_\_ es a una pila.
2. ¿Qué operación es más compleja, tamizar hacia arriba o hacia abajo? ¿Por qué?
3. El concepto básico de heapsort implica

a. eliminar elementos de datos de un montón y luego insertarlos nuevamente.

b. insertar elementos de datos en un montón y luego eliminarlos.

c. copiar datos de dos montones a otro vacío, fusionando

sus elementos.

d. copiando datos de la matriz que representa un montón al árbol de montón.

1. ¿Cuántas matrices, cada una lo suficientemente grande para contener todos los datos, se necesitan

para realizar un heapsort?

1. La complejidad temporal de ejecutar heapsort es O(\_\_\_\_\_\_\_).
2. Comparado con quicksort, heapsort es \_\_\_\_\_\_\_.
3. Para heapificar una matriz:

a. se seleccionan los elementos de la segunda mitad de la matriz.

b. los elementos de la matriz se insertan secuencialmente en una estructura de

datos de montón separada.

c. la matriz se reordena en su lugar para establecer la condición de montón

entre nodos.

d. el algoritmo heapsort se aplica a los nodos de hoja del árbol de montón parcial.

1. La rutina heapify() en una matriz de N elementos toma O(\_\_\_\_\_\_\_) tiempo.
2. Obtener los K elementos con clave más alta de una matriz de N elementos toma

O(\_\_\_\_\_\_\_) tiempo.