1. Aparte de los métodos de inserción, eliminación, búsqueda y recorrido

comunes a todas las estructuras de datos de “base de datos”, las estructuras de datos de matriz deben

tener métodos de acceso directo.

1. Al construir una nueva instancia de una matriz (Listado 2-3):

a. se debe establecer el valor inicial de al menos una de las celdas de la matriz.

b. se debe establecer el tipo de datos de todas las celdas de la matriz.

c. se debe establecer la clave para cada elemento de datos.

d. se debe establecer el número máximo de celdas que puede contener la matriz.

e. ninguna de las anteriores.

1. ¿Por qué es importante utilizar atributos de instancia privados como \_\_nItems

en la definición de la estructura de datos de matriz?

R: Los atributos privados encapsulan los detalles de implementación y evitan modificaciones accidentales desde fuera de la clase. En este caso, \_\_nItems rastrea el número de elementos actualmente en la matriz, lo que es fundamental para muchas operaciones.

1. Insertar un elemento en una matriz desordenada

a. lleva tiempo proporcional al tamaño de la matriz.

b. requiere múltiples comparaciones.

c. requiere desplazar otros elementos para hacer espacio.

d. lleva el mismo tiempo sin importar cuántos elementos haya.

1. Verdadero o falso: cuando eliminas un elemento de una matriz desordenada, en

la mayoría de los casos desplazas otros elementos para rellenar el espacio vacío.

Verdadero

1. En una matriz desordenada, permitir duplicados

a. aumenta los tiempos para todas las operaciones.

b. aumenta los tiempos de búsqueda en algunas situaciones.

c. siempre aumenta los tiempos de inserción.

d. a veces disminuye los tiempos de inserción.

1. Verdadero o falso: en una matriz desordenada, generalmente es más rápido descubrir

que un elemento no está en la matriz que descubrir que sí está.

Falso

1. Las matrices ordenadas, en comparación con las matrices desordenadas, son

a. mucho más rápidas en la eliminación.

b. más rápidas en la inserción.

c. más rápidas en la creación.

d. más rápidas en la búsqueda.

1. Las claves se utilizan con matrices

a. para proporcionar un valor único para cada elemento de la matriz que se puede utilizar para

ordenar los elementos.

b. para descifrar los valores almacenados en la celda de la matriz.

c. para disminuir el tiempo de inserción en matrices desordenadas.

d. para permitir que los tipos de datos complejos se almacenen como un único valor de clave en

la matriz.

1. Los módulos OrderedArray.py (Listado 2-6) y OrderedRecordArray.py

(Listado 2-9) tienen un método find() y un método search().

¿En qué se parecen y en qué se diferencian los dos métodos?

1. Un logaritmo es el inverso de \_\_\_\_Una exponenciación\_\_\_\_\_\_\_\_\_.
2. El logaritmo en base 10 de 1000 es \_\_3\_\_\_.
3. La cantidad máxima de elementos que se deben examinar para

completar una búsqueda binaria en una matriz de 200 elementos es

a. 200.

b. 8.

c. 1.

d. 13.

1. El logaritmo en base 2 de 64 es \_\_6\_\_\_\_.
2. Verdadero o falso: el logaritmo en base 2 de 100 es 2.

Falso

1. La notación O grande indica

a. cómo se relaciona la velocidad de un algoritmo con la cantidad de elementos.

b. el tiempo de ejecución de un algoritmo para una estructura de datos de tamaño determinado.

c. el tiempo de ejecución de un algoritmo para una cantidad determinada de elementos.

d. cómo se relaciona el tamaño de una estructura de datos con la velocidad de uno de sus

algoritmos.

1. O(1) significa que un proceso opera en tiempo \_\_\_\_ constante\_\_\_\_.
2. Las ventajas de usar matrices incluyen

a. el tamaño variable de las celdas de la matriz.

b. la longitud variable de la matriz durante la vida útil de la

estructura de datos.

c. el tiempo de acceso O(1) para leer o escribir una celda de la matriz.

d. el tiempo O(1) para recorrer todos los elementos de la matriz.

e. todas las anteriores.

1. Un colega le pide sus comentarios sobre una estructura de datos que usa una matriz

desordenada sin duplicados y búsqueda binaria. ¿Cuál de los siguientes comentarios tiene sentido?

a. Debido a que la matriz puede almacenar cualquier tipo de datos, una búsqueda binaria

no será eficiente.

b. Debido a que la matriz no está ordenada, una búsqueda binaria no

puede garantizar que se encuentre el elemento que se busca.

c. Debido a que la búsqueda binaria lleva O(N) tiempo, sería mejor

utilizar una matriz ordenada.

d. Debido a que la matriz no tiene duplicados, la búsqueda binaria

realmente no tiene una ventaja sobre la búsqueda lineal más simple.

1. Se le ha pedido que adapte un código que mantiene un registro

sobre cada planeta y sus lunas en un sistema solar como el nuestro en un

sistema que almacenará un registro sobre cada planeta y luna en cada

galaxia conocida. La estructura del registro será un poco más grande para cada

planeta para contener algunos atributos nuevos. Es probable que los registros se

agreguen y actualicen "aleatoriamente" a medida que los telescopios y otros sensores apuntan

a diferentes partes del universo a lo largo del tiempo, completando algunos

atributos iniciales de los registros y luego actualizando otros durante observaciones frecuentes. El código actual utiliza una matriz desordenada para los

registros. ¿Recomendaría algún cambio? Si es así, ¿por qué?

R: Dado que los registros se agregarán y actualizarán aleatoriamente, una matriz desordenada puede no ser la mejor opción, ya que las inserciones y eliminaciones en el medio de la matriz pueden ser costosas en términos de tiempo. Consideraría las siguientes alternativas:

* Lista enlazada: Permite inserciones y eliminaciones eficientes en cualquier posición, pero la búsqueda puede ser más lenta que en una matriz.
* Árbol binario: Permite búsquedas eficientes y ordenamiento natural de los datos, pero puede ser más complejo de implementar.
* Tabla hash: Si se necesita un acceso rápido por clave, una tabla hash puede ser una buena opción.

La elección de la estructura de datos dependerá de las operaciones más frecuentes y de los requisitos de rendimiento específicos de la aplicación.