

1. ¿Qué estructura de datos subyacente se utiliza comúnmente para implementar una cola de prioridad en Python?

En Python comúnmente se implementa una cola de prioridad utilizando una estructura de datos de tipo heap binario, con la implementación de min-heap, el cual se basa internamente en una lista.

2. ¿Cuál es la diferencia entre una cola FIFO tradicional y una cola de prioridad?

Una cola FIFO esta ordenada dependiendo de él orden de entrada donde todos los elementos tienen igual importancia, razón por la cual se ordenan de manera que el primer dato que entra es el primer dato que saldría de la lista. Una cola de prioridad en cambio ordena sus datos teniendo en cuenta un valor de prioridad, en este tipo de cola cada elemento tendrá un valor de prioridad el cual definirá su posición en la lista en donde a diferencia de una cola FIFO el orden de llegada no se tiene en cuenta.

3. ¿Qué módulo proporciona Python para trabajar fácilmente con colas de prioridad?

Heapq

4. ¿Qué ventajas tiene el uso de una cola de prioridad sobre una lista ordenada manualmente?

Una cola de prioridad es más eficiente para insertar y extraer datos. En una lista ordenada manualmente toca reordenar de forma manual todos los datos después de cada inserción, en cambio al insertar un elemento en una cola de prioridad la estructura mantiene el orden automáticamente según su prioridad. Tiene un uso de memoria. En general una cola de prioridad tiene una mayor eficiencia general que una lista ordenada manualmente.

5. Si dos elementos tienen la misma prioridad, ¿cómo decide la cola cuál atender primero?

Si dos elementos tienen la misma prioridad, la cola atenderá los datos por el principio de FIFO, donde el primer dato en entrar será el primero en salir, únicamente si tienen la misma prioridad.

**6. ¿Qué situaciones del mundo real se pueden modelar con colas de prioridad?
Mencione al menos dos.**

Una primera aplicación puede ser en sistemas de logísticas de envíos, donde dependiendo de la prioridad del paquete se manejará la prioridad de los envíos. Otro caso, puede ser la entrada a algún evento, en donde dependiendo de la prioridad del tiquete de entrada se permitirá el acceso al recinto.

7. En un sistema de atención médica, ¿cómo se puede usar una cola de prioridad para organizar a los pacientes?

En un sistema medico puede ser útil utilizar una cola de prioridad en los sistemas de atención medica porque permite atender a los pacientes por la prioridad/urgencia de su condición, y no por su orden de llegada.

8. ¿Cómo afectaría al rendimiento usar una lista simple en lugar de una estructura especializada como heap_pq para manejar prioridades?

Usar en una lista afectaría el rendimiento para manejar prioridades, porque muchas de las funciones de listas tienen una complejidad mayor porque requieren el recorrido completo de la lista para la consulta, eliminación o inserción de los datos. Sin embargo, al usar las estructuras especializadas como heap_pq las complejidades se reducen y por ende el rendimiento es mas eficiente.

9. ¿Qué complejidad temporal tiene la inserción y eliminación/atención en una cola de prioridad basada en heap?

En colas de prioridad basada en heap la inserción y eliminación en el peor caso y en promedio tienen una complejidad temporal de $O(\log N)$.