**Grupo 6**

* **Lina Sofía Ramírez Mojica - 202414153**
* **Daniel Andres Gomez Rodriguez -202414483**

a) ¿Qué estructura de datos subyacente se utiliza comúnmente para implementar una cola de

prioridad en Python?

RTA// Generalmente para implementar una cola de prioridad se usa un heap binario.

b) ¿Cuál es la diferencia entre una cola FIFO tradicional y una cola de prioridad?

RTA// Como dice su nombre en las colas de prioridad se organizan según su prioridad, mientras que una cola FIFO se ordena por el orden de entrada, es decir, el primero en entrar es el primero en salir.

c) ¿Qué módulo proporciona Python para trabajar fácilmente con colas de prioridad

RTA// Usa módulos mínimos(heapq), por lo que se utiliza estructuras como priority queue.

d) ¿Qué ventajas tiene el uso de una cola de prioridad sobre una lista ordenada manualmente?

RTA// La mayor ventaja que tiene sobre la otra es su eficiencia y tiempo, por un lado una lista ordenada se demora más ya que es manual y puede ser O(n), por el otro lado, la cola de prioridad está ordenada en base a su prioridad y es más eficiente (O(log n)).

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

RTA// Generalmente se hace por un orden FIFO, sin embargo depende mayormente de su posición.

f) ¿Qué se debe hacer para que los elementos personalizados puedan ser almacenados en una

cola de prioridad en Python?

RTA// se debe definir bien los elementos y cómo se comparan, esto se puede lograr usando tuplas para prioridad y elemento, de esta forma al agregar elementos personalizados se pueden almacenar y organizar segun la prioridad de estos

g) ¿Qué situaciones del mundo real se pueden modelar con colas de prioridad? Menciona al

menos dos.

RTA// Atención médica en emergencias: Los pacientes no se atienden por orden de llegada, sino según su gravedad. Un paciente con un infarto ( que tiene prioridad alta) será atendido antes que alguien con un dolor leve ( que tiene prioridad baja) lo que permite una organizacion a partir de prioridad.

Planificación de tareas en sistemas operativos: Los procesos con mayor prioridad (como los de tiempo real) se ejecutan antes que los procesos comunes. Esto evita retrasos en operaciones críticas.

h) En un sistema de atención médica, ¿cómo se puede usar una cola de prioridad para

organizar a los pacientes?

RTA// En un sistema de atención médica, una cola de prioridad permite gestionar pacientes de manera eficiente. Cada paciente ingresa con un nivel de urgencia asignado (por ejemplo, 1 para emergencias, 2 para casos urgentes y 3 para consultas rutinarias). La cola procesa primero a los pacientes con prioridad más alta, asegurando que quienes necesitan atención inmediata sean atendidos sin demoras

En Python se usaría heap, insertando pacientes como tuplas (prioridad, paciente). Por ejemplo, un paciente con prioridad 1 (emergencia) saltaría al frente de la cola, mientras que uno con prioridad 3 esperaría su turno. Esto optimiza el tiempo de respuesta en situaciones de emergencia

i) ¿Cómo afectaría al rendimiento usar una lista simple en lugar de una estructura

especializada como heapq para manejar prioridades?

RTA// aunque insertar un elemento en una lista es O(1), el iterar sobre miles de datos o una cantidad superior para cosas como organizar por prioridad puede ser contraproducente ya que tienen que pasar uno por uno, sin embargo en un heapq es garantizado que tanto la inserción como la extracción del elemento más prioritario sean operaciones eficientes (O(log n))

j) ¿Qué complejidad tiene la inserción y extracción en una cola de prioridad basada en heap?

RTA// tanto la insercion como la extraccion en una cola de prioridad tiene una complejidad temporal de O(log(n))