

Trabajo Práctico 2

Integrantes: Ferrer Paira Iñaki, Spaggiari Fiamma

Respuestas

Problema 1 (Sala de Emergencias)

Para un correcto flujo de los pacientes en la sala de emergencias implementamos una **Cola de Prioridad** basada en la estructura de datos **Heap binario**, utilizando el módulo heapq.

Lo que tuvimos en cuenta para realizar nuestro código fue:

- **Atención al paciente más delicado y un criterio de desempate:** Con la cola de prioridad garantizamos que el elemento de mayor prioridad sea siempre el primero en ser atendido, en este caso el paciente más crítico era el que menor nivel de riesgo tenía por lo que hicimos uso de un Min-Heap.
Si tenemos el caso de que dos o más pacientes tengan el mismo nivel de riesgo para ser atendidos tenemos que priorizar el orden de llegada por lo que utilizamos una tupla (prioridad, contador, elemento), en este caso al tener las prioridades iguales se evalúa el segundo elemento (contador) que es el que indica que paciente ingresó primero.
- **Estructura de datos genérica:** La implementación de la Cola de Prioridad es genérica y desacopla de la clase Paciente. (permite almacenar cualquier tipo de dato) → es reutilizable

Orden de complejidad O de inserciones y eliminaciones:

- **Inserciones (Ingreso de un Paciente):** La operación de insertar un nuevo elemento en un Heap binario implica añadir el elemento al final y luego subirlo hacia arriba en el árbol para mantener la propiedad de heap.

El orden de complejidad de una inserción en un Heap es $O(\log N)$.

- **Eliminaciones (Atención de un Paciente):** La operación de extraer el elemento de mayor prioridad implica remover la raíz del árbol, después de la eliminación, el último elemento de la hoja se mueve a la raíz y luego se baja para restablecer la propiedad de heap.

El orden de complejidad de una eliminación en un Heap es $O(\log N)$.

Problema 2 (Temperaturas_DB)

El código de este problema se basa en un sistema de gestión de mediciones de temperatura, diseñado para almacenar y consultar datos de temperatura asociados a fechas. La solución se basa en una estructura de datos de árbol AVL

La solución se compone de tres clases:

- **NodoAVL**
- **ArbolAVL**
- **Temperaturas_DB**

Además cuenta con un test (test_temperaturas_db) para probar el correcto funcionamiento de los métodos en Temperaturas_DB y un main donde se hacen pruebas del código hecho.

Pasando a la complejidad temporal de los métodos principales de Temperaturas_DB:

MÉTODO	DESCRIPCIÓN	COMPLEJIDAD	ANÁLISIS
guardar_temperatura(fecha, temp)	Guarda/Actualiza una medición	$O(\log n)$	Inserción/Actualización en AVL → operaciones con altura logarítmica
devolver_temperatura(fecha)	Busca una temperatura por fecha	$O(\log n)$	Búsqueda en AVL → operaciones con altura logarítmica
max_temp_rango(f1, f2)	Máxima temperatura en un rango	$O(\log n + k)$	Búsqueda en AVL → $O(\log n)$ y recorrido de k elementos en el rango
min_temp_rango(f1, f2)	Mínima temperatura en un rango	$O(\log n + k)$	Búsqueda en AVL → $O(\log n)$ y recorrido de k elementos en el rango
temp_extremos_rango(f1, f2)	Mínima y máxima temperatura en un rango	$O(\log n + k)$	2 búsquedas y recorridos en AVL → $O(\log n + k)$
borrar_temperatura(fecha)	Elimina una medición	$O(\log n)$	Eliminación en AVL → operaciones con altura logarítmica
devolver_temperaturas_rango(f1, f2)	Listado de mediciones en un rango	$O(\log n + k)$	Recorrido en orden de los nodos dentro de un rango.
cantidad_muestras()	Cantidad total de mediciones	$O(n)$	Recorrido completo del árbol

Problema 3 (Palomas mensajeras)

El problema se modela como un **grafo no dirigido ponderado** donde las aldeas son nodos y las distancias entre ellas son los pesos de las aristas. Para encontrar las rutas más eficientes, se utiliza el **algoritmo de Dijkstra**. Este algoritmo calcula las distancias más cortas desde "Peligros" a todas las demás aldeas y registra el predecesor de cada aldea en su camino óptimo.

La **lógica de replicación** del mensaje se basa en: una aldea recibe la noticia (la del camino más corto) y replica el mensaje a aquellas aldeas que la tienen como el camino más corto.

La **distancia total recorrida** se calcula sumando las distancias mínimas desde "Peligros" a cada una de las aldeas alcanzables.