

Informe - Sala de Emergencias

En este proyecto se simula una sala de emergencias donde los pacientes ingresan con distintos niveles de riesgo clínico: 1 (crítico), 2 (moderado) y 3 (bajo). El objetivo principal es atender primero a los pacientes con mayor urgencia, es decir, aquel con el número de riesgo más bajo. En caso de empate, se atiende primero al paciente que llegó antes.

Para cumplir con este criterio se eligió como estructura principal un montículo binario (binary heap), que permite mantener una cola de prioridad eficiente. Esta estructura es genérica, por lo tanto puede ser reutilizada para otros fines sin depender de los pacientes específicamente. La implementación del montículo almacena tuplas de la forma (riesgo, orden_llegada, paciente), lo que permite ordenar primero por riesgo y luego por orden de llegada en caso de empate.

El orden de complejidad temporal para las operaciones sobre el montículo binario es:

- Inserción: $O(\log n)$
- Eliminación del mínimo: $O(\log n)$

Esto garantiza un rendimiento eficiente tanto para registrar nuevos pacientes como para determinar rápidamente cuál debe ser el siguiente en ser atendido. La separación entre la estructura de datos y su aplicación específica se respetó mediante la clase genérica `MonticuloBinario`, y la lógica del triaje fue implementada dentro de `ColaConPrioridad`.

En resumen, se utilizó una estructura adecuada al problema planteado, logrando eficiencia y claridad en la implementación, con posibilidad de ser reutilizada para otros contextos que requieran priorización dinámica.