

Parte 1

2	<p>Explica las estructuras de datos usadas y su justificación: En el caso que haya usado árboles de búsqueda basta con mencionar las ventajas del tipo de árbol que escogieron para relacionarlo con el problema.</p> <ul style="list-style-type: none">• 1.0 pts: Por mencionar una estructura adecuada. En general, se espera un árbol de búsqueda balanceado como AVL o KD-Tree¹.• 1.0 pts: Por justificar la estructura elegida con una explicación general del porqué se ajusta a las búsquedas solicitadas. <p>Por ejemplo: Si utiliza un KD-Tree, explicar cómo funciona y explicar por qué es útil para búsquedas en rango.</p>
4	<p>Explica brevemente como la implementación de cada consulta, según la EDD escogida, cumple con la complejidad exigida.</p> <ul style="list-style-type: none">• 0.5 pts: Por la consulta BY-ID.• 0.5 pts: Por la consulta BY-YEAR.• 0.5 pts: Por la consulta BY-YEAR-REGION.• 1.0 pts: Por la consulta IN-X-RANGE.• 1.5 pts: Por la consulta IN-CIRCLE. <p>Para cada consulta:</p> <ul style="list-style-type: none">• Debe indicar la estructura usada para resolverla.• Indicar la complejidad de tiempo lograda y justificarla.

¹Para una explicación práctica y ejemplos de búsqueda en rango con KD-Trees, se puede revisar este artículo de Nawaz y estas diapositivas del curso CMSC420 de la Universidad de Maryland.

Parte 2

4	<p>Explica las estructuras de datos utilizadas y su justificación. Como solución propuesta se recomienda el uso de Segment Tree².</p> <ul style="list-style-type: none"> • 2 pts: Explica correctamente qué estructura se usó (idealmente Segment Tree), y por qué cumple con la complejidad esperada. Debe detallar: <ul style="list-style-type: none"> – Cómo se representa la estructura (por ejemplo, array o punteros). – Cómo se construye (función de construcción con recursión o iterativa). – Complejidad de construcción (esperado: $O(N)$). • 1 pt: Menciona correctamente la estructura y su uso general, pero omite detalles relevantes como la construcción o la justificación de complejidad. Puede tener errores menores o inexactitudes. • 0 pts: La explicación es incorrecta o muy incompleta. No se entiende claramente qué estructura usa o cómo se construye. <p>Explica cómo la estructura cumple la complejidad en los eventos RANGE-SUM y UPDATE</p> <ul style="list-style-type: none"> • 2 pts: Explica cómo se implementan ambos eventos, RANGE-SUM y UPDATE, usando la estructura elegida. Debe: <ul style="list-style-type: none"> – Explicar cómo se dividen los rangos. – Mostrar cómo se propagan los cambios si aplica. – Justificar que ambas operaciones toman tiempo $O(\log N)$. • 1 pt: Explica solo uno de los dos eventos, o explica ambos pero de manera superficial o con errores menores. • 0 pts: No explica correctamente ninguno de los eventos, o no queda claro cómo se logran las complejidades.
---	---

²Explicación Segment Tree

2	<p>Se espera que se proponga usar <i>Lazy Propagation</i>, que sirve para este tipo de operaciones.</p> <ul style="list-style-type: none"> • 2 pts: Describe correctamente qué es Lazy Propagation y cómo se implementa para RANGE-UPDATE. Debe incluir: <ul style="list-style-type: none"> – Cómo se marca un nodo con una operación pendiente. – Cómo se aplica la operación al bajar en el árbol. – Cómo se asegura que las otras operaciones mantengan su complejidad. • 1 pt: Menciona Lazy Propagation correctamente, pero con errores menores o sin detallar bien su implementación (por ejemplo, no queda claro cómo se actualizan los hijos). • 0 pts: No menciona Lazy Propagation o lo explica incorrectamente. Indica que no se puede lograr lo pedido o da una solución que no cumple la complejidad.
---	--