

Informe – Entrega 2 (Árboles Binarios de Búsqueda)

Proyecto: SmartRoute Event

Integrantes:

- Sergio Andres Martinez Cifuentes 2242039
- Andres Felipe Guaqueta Rojas 2242034
- Andres Sebastian Pinzon Gutierrez 2221887
- Daniel Eduardo Rincon Arias

1. Introducción

En la primera entrega se implementó una versión básica del sistema SmartRoute Event utilizando listas enlazadas. En esta segunda versión, el sistema se mejora mediante la implementación de un árbol binario de búsqueda (ABB) para optimizar la administración y consulta de los datos, permitiendo acceder y ordenar la información de manera más eficiente.

2. Contexto del Proyecto

SmartRoute Event es una aplicación dirigida a turistas-eventuales que visitan una ciudad por un evento y desean conocer lugares cercanos de interés. Esta versión utiliza un árbol binario de búsqueda para mejorar el manejo de los lugares registrados, ofreciendo consultas más rápidas de una forma más eficiente.

3. Objetivo General

Diseñar e implementar una versión mejorada del sistema SmartRoute Event que utilice árboles binarios para optimizar el almacenamiento, la búsqueda y la visualización de los lugares de interés.

4. Objetivos Específicos

- Implementar un árbol binario de búsqueda (ABB) para organizar los lugares según su distancia.
- Permitir operaciones de inserción, recorrido y búsqueda sobre los datos.
- Agregar una nueva funcionalidad que aproveche la estructura del árbol.
- Comparar la eficiencia de la estructura con respecto a la lista enlazada.

5. Implementación en Python

Se implementaron dos clases principales: `Nodo` y `ArbolLugares`. Cada nodo representa un lugar con sus atributos y el árbol organiza los lugares según la distancia.

6. Nueva Funcionalidad:

Búsqueda del Lugar Más Cercano La función `buscar_mas_cercano()` permite encontrar rápidamente el lugar con la menor distancia registrada en el árbol. Aprovecha la estructura del ABB, donde el nodo más a la izquierda contiene el valor mínimo. Esto mejora la eficiencia de la búsqueda en comparación con la versión anterior basada en listas enlazadas.

7. Funcionalidades e Implementaciones (Versión con Árboles)

En esta sección se presenta el desarrollo técnico de la versión mejorada del sistema SmartRoute Event, en la cual se implementa una estructura de Árbol Binario de Búsqueda (ABB) para optimizar el manejo de los lugares registrados. A continuación, se describen las clases principales, los métodos implementados, la nueva funcionalidad agregada y las herramientas utilizadas durante el desarrollo.

7.1 Clase Nodo

La clase `Nodo` representa la estructura fundamental del Árbol Binario. Cada nodo almacena los datos de un lugar y mantiene dos referencias: una al subárbol izquierdo (lugares más cercanos) y otra al subárbol derecho (lugares más lejanos).

Esta estructura sustituye al puntero único 'siguiente' de la versión con listas enlazadas, permitiendo una organización jerárquica que facilita el ordenamiento y la búsqueda eficiente.

7.2 Clase ArbolLugares

La clase `ArbolLugares` administra los nodos del árbol y contiene las funciones principales de inserción, recorrido y búsqueda. A diferencia de la versión

anterior, las operaciones se realizan de forma recursiva, manteniendo automáticamente los datos ordenados según la distancia.

El método 'insertar' asegura que los lugares se organicen automáticamente en el árbol según la distancia. Esto elimina la necesidad de ordenar los datos manualmente, como ocurría con las listas enlazadas.

7.3 Recorrido en orden (Inorden)

El recorrido inorden permite mostrar los lugares ordenados desde el más cercano hasta el más lejano. Este recorrido aprovecha la naturaleza del árbol binario de búsqueda para obtener la información en orden creciente.

Gracias a este recorrido, los lugares se presentan organizados automáticamente sin necesidad de ordenar una lista, mejorando la eficiencia y claridad de salida del programa.

7.4 Nueva funcionalidad:

Búsqueda del lugar más cercano Una mejora significativa incorporada en esta versión es la función 'buscar_mas_cercano', que aprovecha la propiedad del árbol para identificar el nodo más a la izquierda, correspondiente al lugar con menor distancia.

Esta funcionalidad no existía en la versión basada en listas enlazadas. En lugar de recorrer todos los elementos para encontrar el más cercano, ahora el sistema identifica el nodo mínimo en tiempo $O(h)$, donde h es la altura del árbol.

7.5 Librerías y herramientas utilizadas

El desarrollo de esta versión se realizó utilizando exclusivamente las funcionalidades nativas de Python. No fue necesario el uso de librerías externas, lo que garantiza portabilidad y simplicidad.

8. Conclusiones

Con esta segunda entrega se logró mejorar el sistema SmartRoute Event mediante el uso de árboles binarios de búsqueda. Esta estructura permite un manejo más eficiente de la información y facilita la implementación de nuevas funcionalidades. La búsqueda del lugar más cercano agrega valor práctico al sistema y simula una aplicación real de recomendación de rutas.