**1.Giới thiệu**

* 🌳 Cây tìm kiếm nhị phân là cấu trúc dữ liệu có tốc độ tìm kiếm nhanh, nhưng có một số hạn chế khi dữ liệu được chèn vào theo thứ tự đã được sắp xếp.
* 🌳 Giải pháp cho vấn đề này là cây đỏ đen, có một vài đặc điểm đặc biệt giúp cây cân bằng hiệu quả.

**2.Định nghĩa cây đỏ đen**

* 🔴 Cây đỏ đen là BST tuân thủ các quy tắc về màu sắc của các node.
* 🔴 Các quy tắc đảm bảo cây luôn cân bằng và tối ưu hóa thời gian truy xuất.

**3.Phép quay**

* 🔄 Phép quay thực hiện sự thay đổi vị trí của các node để đảm bảo quy tắc cân bằng của cây.
* 🔄 Node đỉnh của phép quay chuyển xuống và về bên trái/phải dựa vào loại phép quay.

**4.Thêm node mới**

* 🆕 Thêm node mới có thể làm vi phạm các quy tắc của cây đỏ đen.
* 🆕 Quy trình gồm các bước kiểm tra và áp dụng phép quay hoặc lật màu để phục hồi cân bằng.

**4.1.Các phép lật màu trên đường đi xuống**

* ⬛ Khi một node đen có hai node con đỏ, tiến hành đổi màu các node để đảm bảo tuân thủ các quy tắc màu sắc.
* ⬛ Phép lật màu không làm thay đổi chiều cao đen của cây.

**4.2.Các phép quay khi chèn node**

* 🔄 Sau khi chèn node mới, kiểm tra và thực hiện các phép quay nếu cần để đảm bảo quy tắc đỏ đen không bị vi phạm.
* 🔄 Node X có thể là node cháu nội hoặc cháu ngoại, thực hiện các thao tác để đưa node về vị trí đúng.