**CHƯƠNG 4: Cây nhị phân tìm kiếm**

**Câu 1: Hãy trình bày các vấn đề sau: Định nghĩa và đặc điểm của Cây nhị phân tìm kiếm; Các thao tác thực hiện tốt trong kiểu này; Hạn chế của kiểu cấu trúc dữ liệu này?**

* **Định nghĩa và đặc điểm của Cây nhị phân tìm kiếm:**

**Cây nhị phân tìm kiếm**, hay còn được biết đến với tên gọi Tiếng Anh là Binary Search Tree (BST), là một cây mà trong đó tất cả các nút (node) đều có các các đặc điểm như sau:

* Cây con bên trái của một nút có khóa (key) nhỏ hơn giá trị khóa của nút cha của cây con này.
* Cây con bên phải của một nút có khóa (key) lớn hơn giá trị khóa của nút cha của cây con này.

Vì vậy, một Cây nhị phân tìm kiếm phân chia tất cả các cây con của nó thành hai phần: *cây con bên trái* và *cây con bên phải* theo điều kiện **left < info < right.**

* **Các thao tác trên Cây nhị phân tìm kiếm:**

Một số thao tác cơ bản có thể thực hiện trên Cây nhị phân tìm kiếm như:

* *Tìm kiếm:* Tìm một nút trên Cây nhị phân tìm kiếm.
* *Chèn:* Thêm một nút mới vào Cây nhị phân tìm kiếm.
* *Duyệt:* Có 3 hình thức duyệt phổ biến trên Cây nhị phân tìm kiếm là duyệt theo thứ tự LNR, LRN, và NLR.
* *Xóa:* Thực hiện xóa một nút trên Cây nhị phân tìm kiếm.

Trong đó, thao tác thực hiện tốt nhất là tìm kiếm, còn thao tác chèn, xóa khá phức tạp và tốn nhiều thời gian.

* **Hạn chế của Cây nhị phân tìm kiếm:**

Bên cạnh ưu điểm là thực hiện thao tác xóa và chèn nút rất nhanh, có thể cấp phát thêm nút hoặc thu hồi bộ nhớ đã cấp phát cho nút, đặc biệt là kích thước của cây không bị giới hạn bởi 64KB, thì Cây nhị phân tìm kiếm có một hạn chế lớn đó là gây tốn bộ nhớ vì phải lưu trữ địa chỉ con trỏ của các nút con bên trái, phải.

**Câu 2: So sánh Cây nhị phân tìm kiếm với các cấu trúc dữ liệu cơ bản như Danh sách đặc, Danh sách liên kết, Danh sách hạn chế.**

* **Giống nhau:**
* Đều thực hiện được các thao tác cơ bản như thêm, chèn, xóa, tìm kiếm, các phép duyệt tương tự các kiểu dữ liệu khác.
* Tương tự như kiểu cấu trúc dữ liệu Danh sách liên kết, Cây nhị phân tìm kiếm cũng có khả năng quản lý một tập các phần tử có số lượng lớn, được cấp phát rời rạc trong bộ nhớ.
* Cây nhị phân tìm kiếm còn có điểm giống với cấu trúc dữ liệu Danh sách liên kết là đều tốn bộ nhớ để lưu trữ các biến con trỏ
* **Khác nhau:**
* Mỗi nút (phần tử) trong Cây nhị phân tìm kiếm ngoài thành phần lưu trữ giá trị của nút (info), còn có hai thành phần kiểu địa chỉ bộ nhớ dùng để lưu trữ địa chỉ của nút con bên trái (left) và lưu địa chỉ nút con bên phải (right). Trong đó, *cây con bên trái* và *cây con bên phải* luôn tuân theo điều kiện **left < info < right.**
* Xét về tốc độ thực hiện thao tác, thì Cây nhị phân tìm kiếm có *khả năng tìm kiếm nhanh* hơn các kiểu dữ liệu khác. Bởi vì tính chất, giá trị của một nút sẽ lớn hơn các giá trị của nhánh con bên phải và nhỏ hơn các giá trị của nhánh con bên phải, nên việc tìm kiếm trên Cây nhị phân tìm kiếm sẽ tiết kiểm được nhiều thời gian dễ dàng hơn. Bên cạnh đó, thì việc thực hiện thao tác *thêm/xóa nút trong cây* lại tốn khá nhiều thời gian và khá phức tạp vì phải thực hiện nhiều phép so sánh.