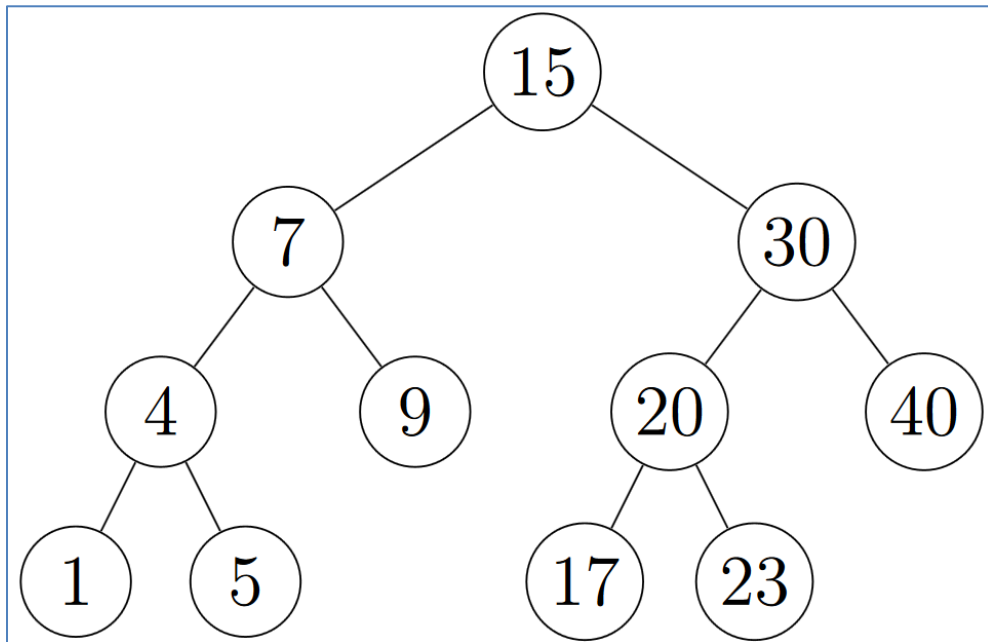


BÀI THỰC HÀNH TUẦN 6

CÂY NHỊ PHÂN TÌM KIẾM

Cây nhị phân tìm kiếm là cây có tính chất: Các node có giá trị khóa nhỏ hơn khóa ở node gốc sẽ nằm ở cây con trái, các node có giá trị khóa lớn hơn giá trị khóa node gốc thì nằm ở cây con bên phải.



Hình minh họa 1: Cây nhị phân tìm kiếm

1. Số lượng node cực đại ở cấp k là 2^k
2. Số lượng node cực đại của một cây BST có độ sâu d là

$$n = 2^0 + 2^1 + 2^2 + \dots + 2^{d-1} + 2^d = \sum_{k=0}^d 2^k = 2^{d+1} - 1$$

3. Một cây BST đầy đủ có n node thì sẽ có độ sâu là:

$$d = \log(n + 1) - 1$$

Các thao tác thực hiện trên cây BST:

- Thêm node vào cây.

Input: Cây BST t , node mới a .

Output: Cây BST t với node a được thêm vào.

Bước 1. Bắt đầu từ node gốc t

Bước 2. Kiểm tra giá trị node mới có bằng với giá trị tại t không, nếu bằng thì dừng do trùng khóa.

Bước 3. Nếu giá trị node mới nhỏ hơn t, thì:

3.1 Nếu t không có cây con trái thì thêm a vào cây con trái của t.

3.2 Nếu t có cây con trái thì gán t = cây con trái của t và lặp lại bước 2

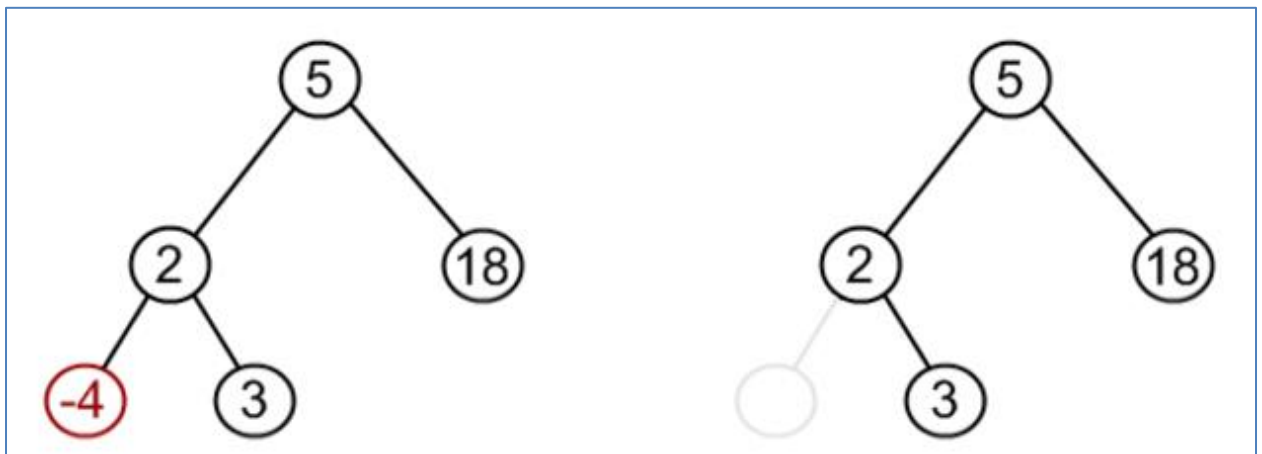
Bước 4. Nếu giá trị node mới lớn hơn t, thì:

4.1 Nếu t không có cây con phải thì thêm a vào thành cây con phải của t.

4.2 Nếu t có cây con phải thì gán t = cây con phải của t và lặp lại bước 2.

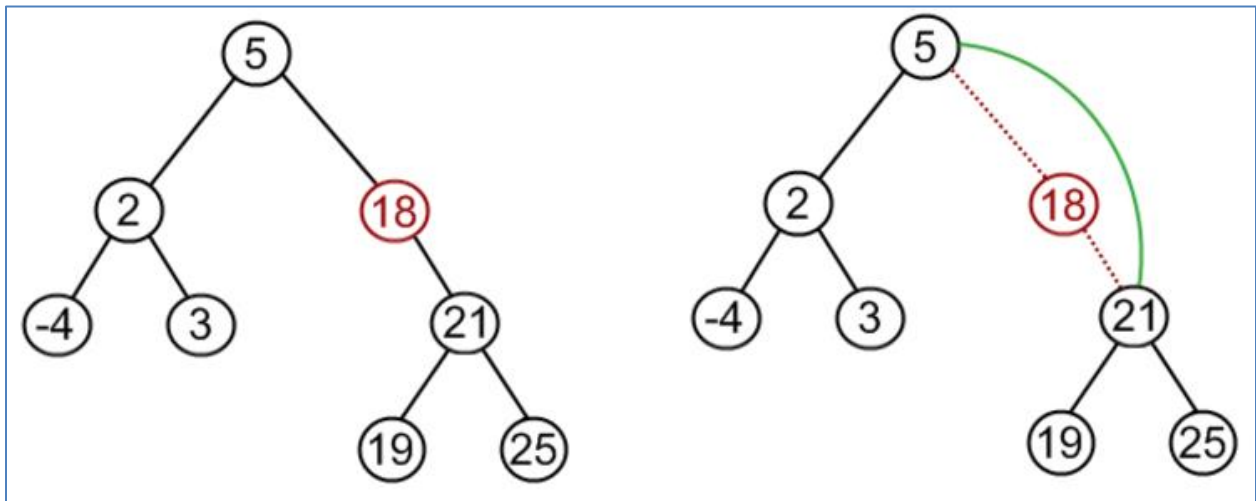
- **Xóa node trên cây:** Có 3 trường hợp có thể xảy ra khi xóa một node ra khỏi cây BST. Trong các ảnh minh họa bên dưới, các node màu đỏ là node sẽ bị xóa ra khỏi cây BST.

- Xóa node lá:

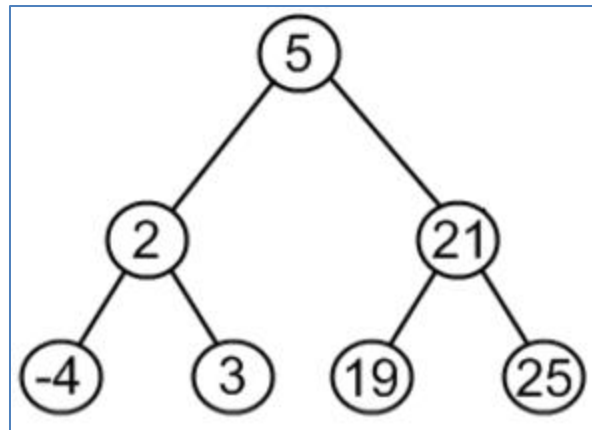


Hình minh họa 2: Xóa node lá

- Xóa node có một con:

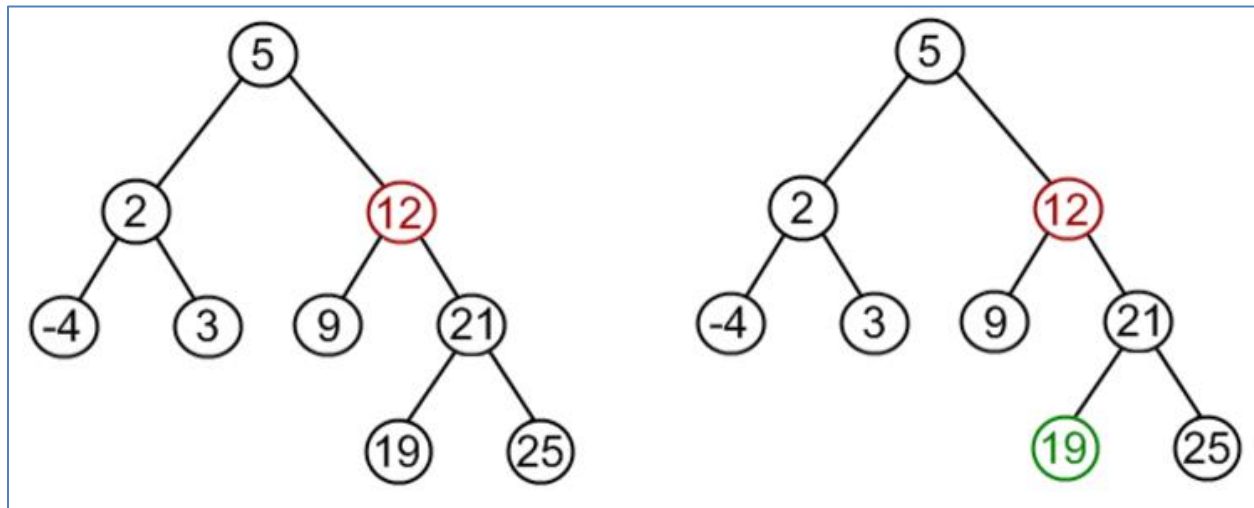


Hình minh họa 3: Tìm node con của node cần xóa (18 trong ví dụ này) sau đó xóa node 18 và thay bằng node con 21

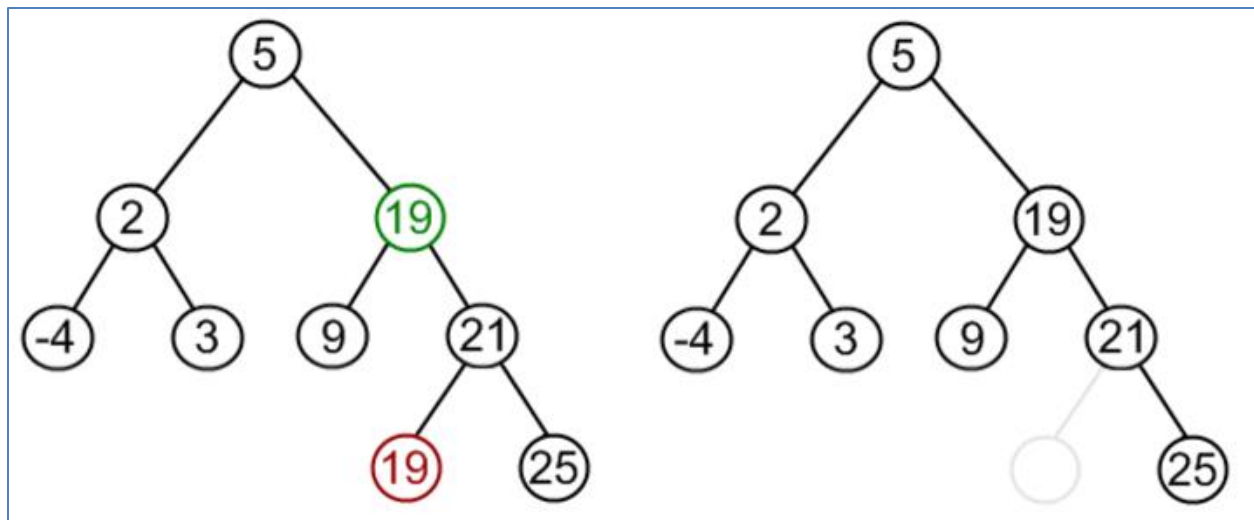


Hình minh họa 4: Kết quả sau khi xóa node 18

- Xóa node có đủ hai cây con: Đây là trường hợp phức tạp nhất trong việc xóa node ra khỏi cây BST. Ta sẽ bắt đầu công việc này bằng việc tìm **phần tử nhỏ nhất** của cây con phải và thực hiện thay thế phần tử bị xóa bằng phần tử này. Cuối cùng ta xóa phần tử thay thế.



Hình minh họa 5: Tìm phần tử thay thế. Trong trường hợp này là node 19



Hình minh họa 6: Thực hiện sao chép node thay thế và xóa node thay thế ở vị trí cũ

- **Duyệt theo Pre-order:** Với chiến lược này, cây BST được duyệt theo thứ tự: node gốc, cây con trái, cây con phải.
- **Duyệt theo In-order:** Với chiến lược này, cây BST được duyệt theo thứ tự cây con trái, node gốc, cây con phải.
- **Duyệt theo Post-Order:** Với chiến lược này, cây BST được duyệt theo thứ tự cây con trái, cây con phải, node gốc

Bài tập nộp trên Moodle:

1. Viết chương trình thực hiện các thao tác sau:
 - a. Định nghĩa cấu trúc cây BST với các node là các giá trị nguyên.
 - b. Thực hiện các thao tác thêm, xóa, duyệt cây.
2. Viết chương trình thực hiện các yêu cầu sau:
 - a. Đọc một mảng các số thực kiểu float từ tập tin văn bản và chuyển thành cây nhị phân tìm kiếm. Tập tin văn bản đầu vào có định dạng như sau:
 - b. Kiểm tra xem cây nhị phân tìm kiếm đọc từ file ở trên có “đầy đủ” không.
 - c. Nếu cây BST ở trên không “đầy đủ” thì chuyển nó thành cây BST đầy đủ. Xuất kết quả ra màn hình.