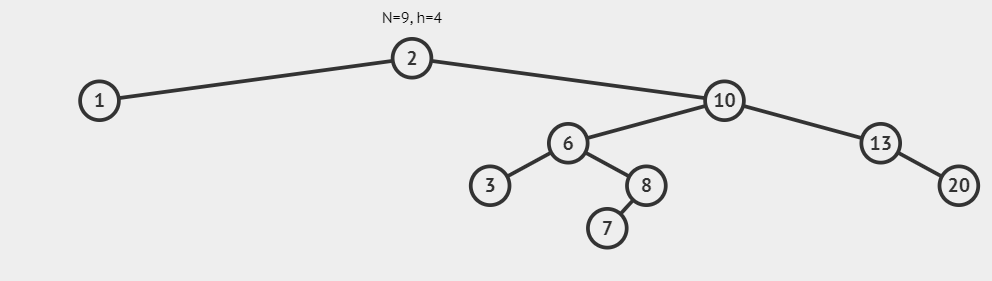
## Bài 1.

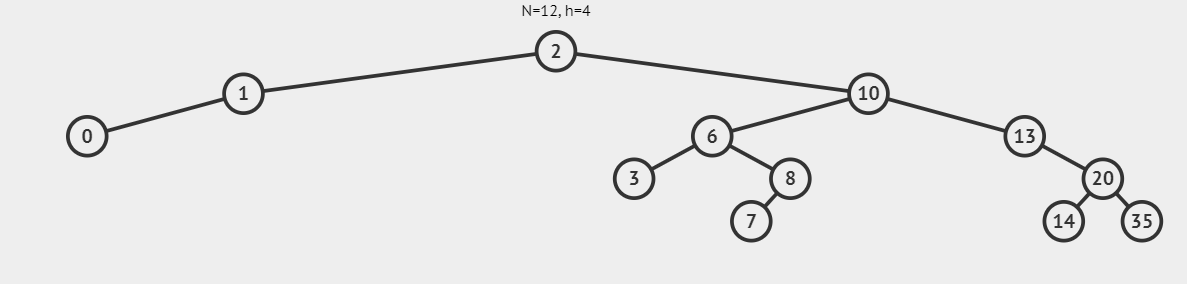
Bài 1 e làm gộp với file cài đặt cây ạ.

## Bài 2.

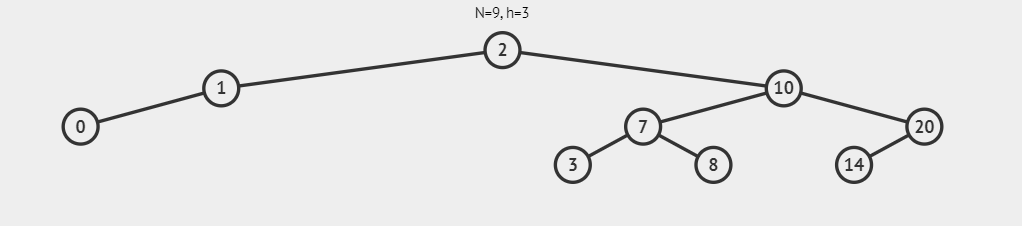
* Vẽ cây tìm kiếm nhị phân với dãy các node: 2, 1, 10, 6, 3, 8, 7, 13, 20:



* Vẽ cây nhị phân sau khi đã thêm các node 14, 0, 35:



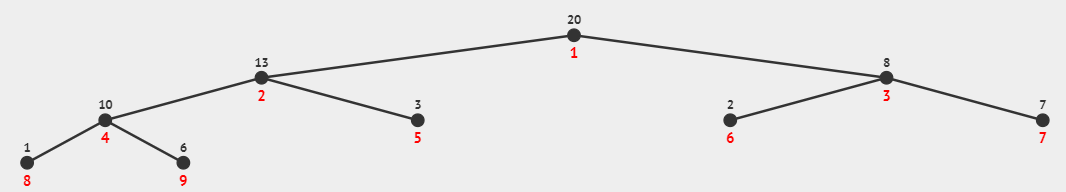
* Vẽ cây nhị phân sau khi xóa các node 6, 13, 35:



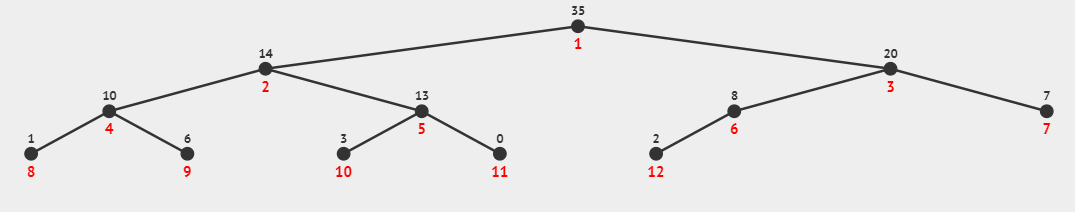
## Bài 3.

Loại Heap tree sử dụng là max heap.

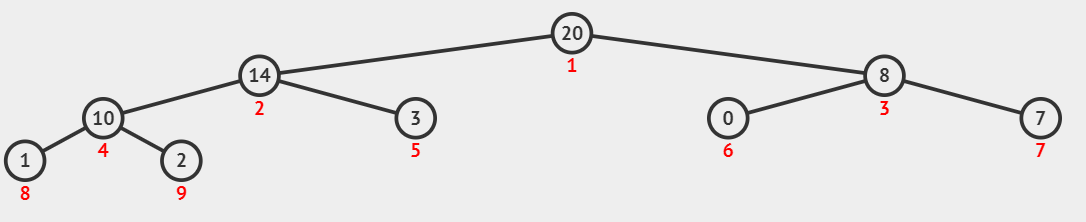
* Vẽ heap tree với dãy các node: 2, 1, 10, 6, 3, 8, 7, 13, 20:



* Vẽ heap tree sau khi đã thêm các node 14, 0, 35:



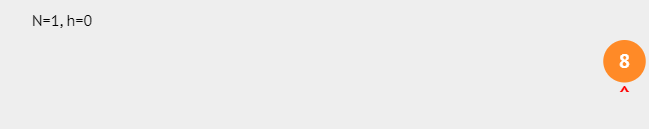
* Vẽ heap tree sau khi đã xóa các node 6, 13, 35:



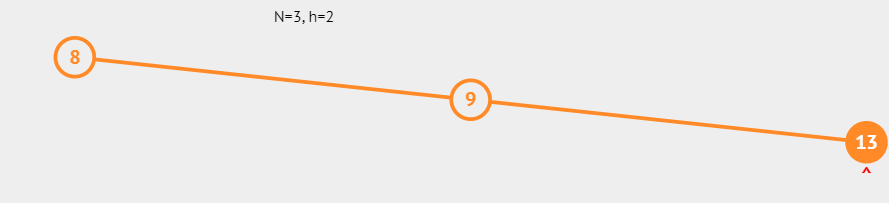
## Bài 4.

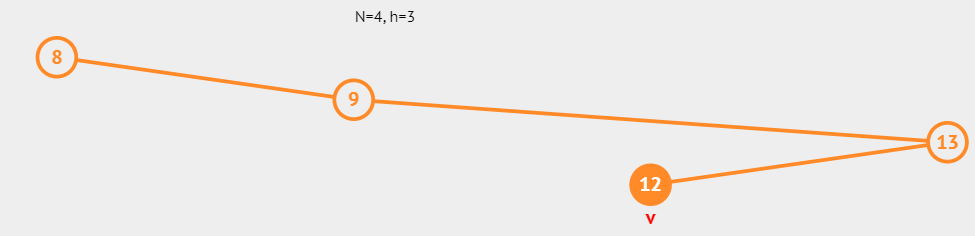
Dãy số: 8, 9, 13, 12, 3, 15, 6, 14, 7, 19 (S1)

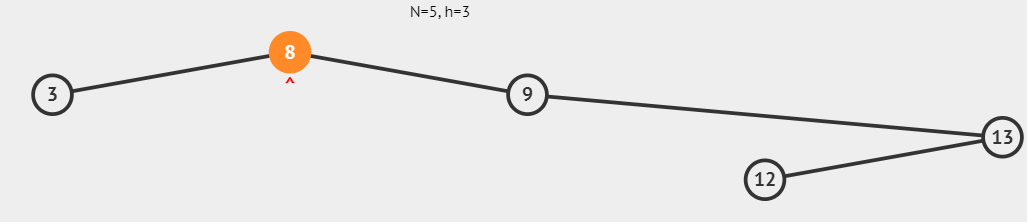
* Vẽ cây tìm kiếm nhị phân sau khi thêm từng node trong dãy trên:

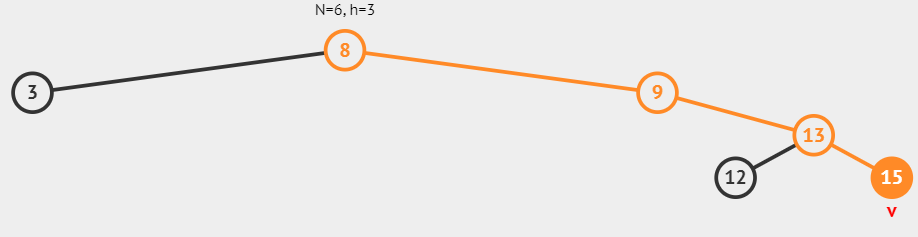


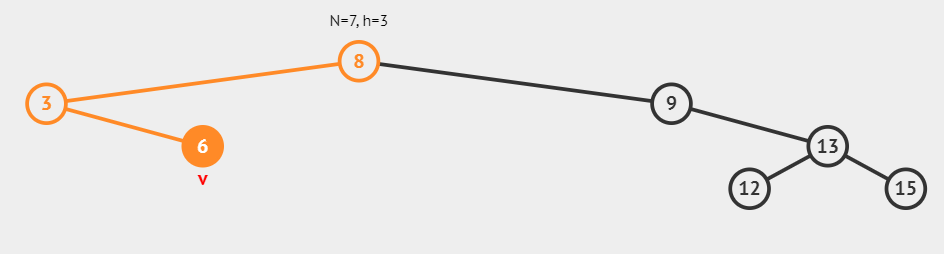


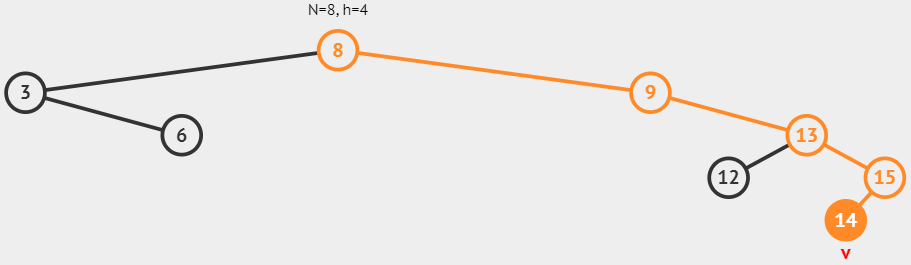


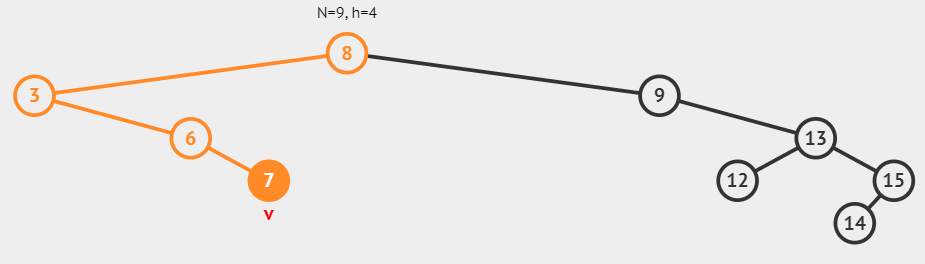


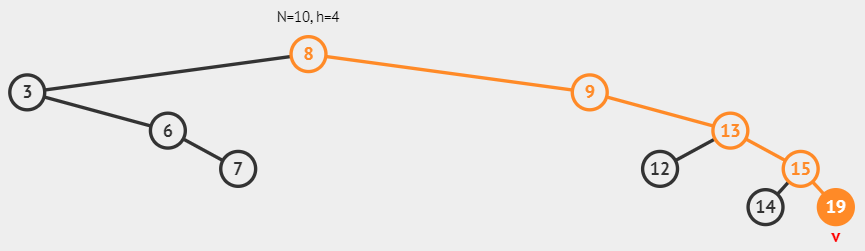












Cây tìm kiếm đã được hoàn thiện.

* Thủ tục tìm kiếm và xóa node có giá trị lớn nhất: node 19

Đây là trường hợp đặc biệt vì node 19 là node lớn nhất và được thêm vào cuối cùng, do đó ta chỉ cần tìm node bằng cách duyệt từng node rồi đi sang bên phải của node đó. Đến cuối sẽ là node 19 là node cần xóa.

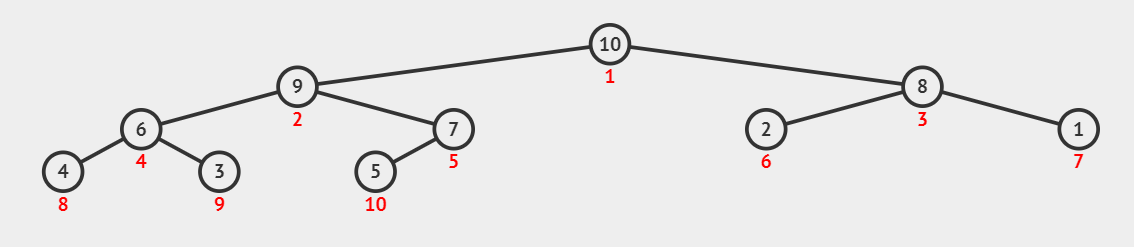
* Thủ tục tìm kiếm và xóa node nhỏ nhất: node 3

Đầu tiên ta sẽ đi tìm node 3. Vì 3 bé hơn node gốc 8 nên ta sẽ đi sang bên trái và tìm thấy node 3. Node 3 thuộc trường hợp chỉ có một con nên ta chỉ cần bypass node 3 và nối node gốc 8 với node successor của node 3 (cây ngoài cùng bên trái của cây con bên phải: node 6). Node 3 đã được xóa.

## Bài 5.

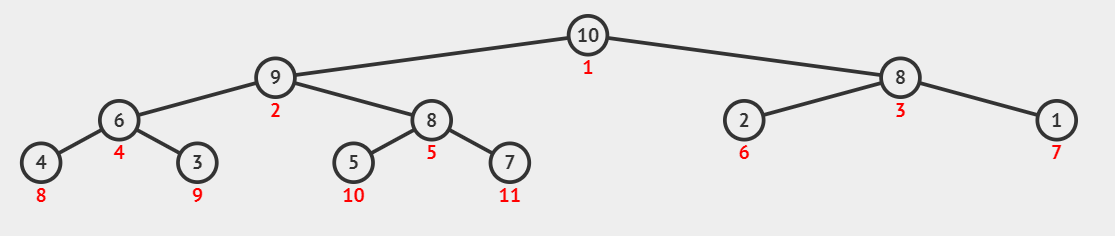
Dãy số: 1: 8, 4, 9, 10, 5, 2, 1, 6, 3, 7 (S2)

* Heap Tree thu được sau khi thêm vào các node theo thứ tự trên:



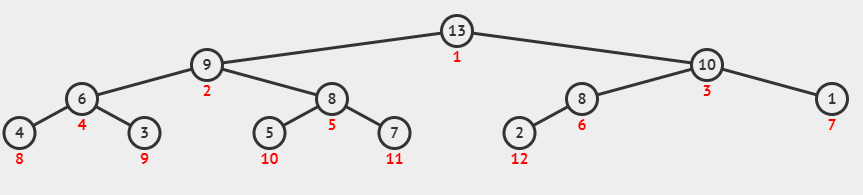
* Các bước thêm các node từ dãy S1: 8, 9, 13, 12, 3, 15, 6, 14, 7, 19 vào heap tree đã xây dựng ở trên:

+ Thêm 8:

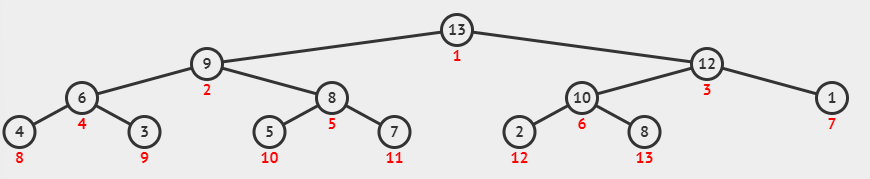


+ Thêm 9: đã có

+ Thêm 13:

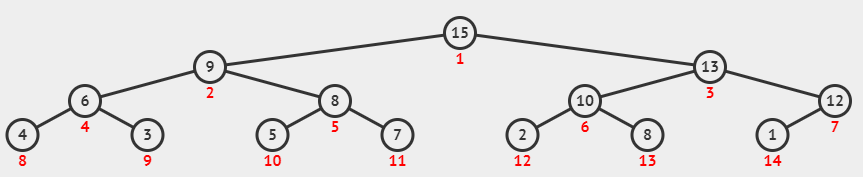


+ Thêm 12:



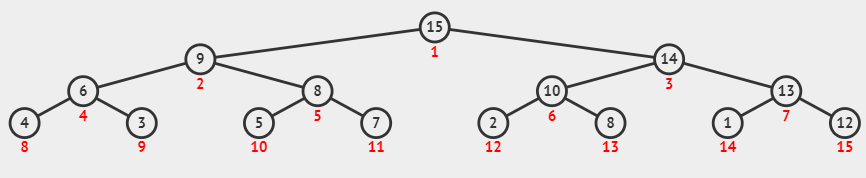
+ Thêm 3: đã có

+ Thêm 15:



+ Thêm 6: đã có

+ Thêm 14:



+ Thêm 7: đã có

+ Thêm 19:

