**12. Priority Queues:**

**12.1 Khái Niệm:**

* Tập hợp các đối tượng được ưu tiên.
* Chèn: Theo nguyên tắc đến trước được phục vụ trước.
* Xóa: Dựa theo mức độ ưu tiên của các đối tượng.
* Khóa được liên kết khi phần tử được chèn vào hàng đợi ưu tiên.
* Phần tử có khóa nhỏ nhất sẽ là phần tử tiếp theo được xóa.

**12.2 Cấu trúc dữ liệu Heap:**

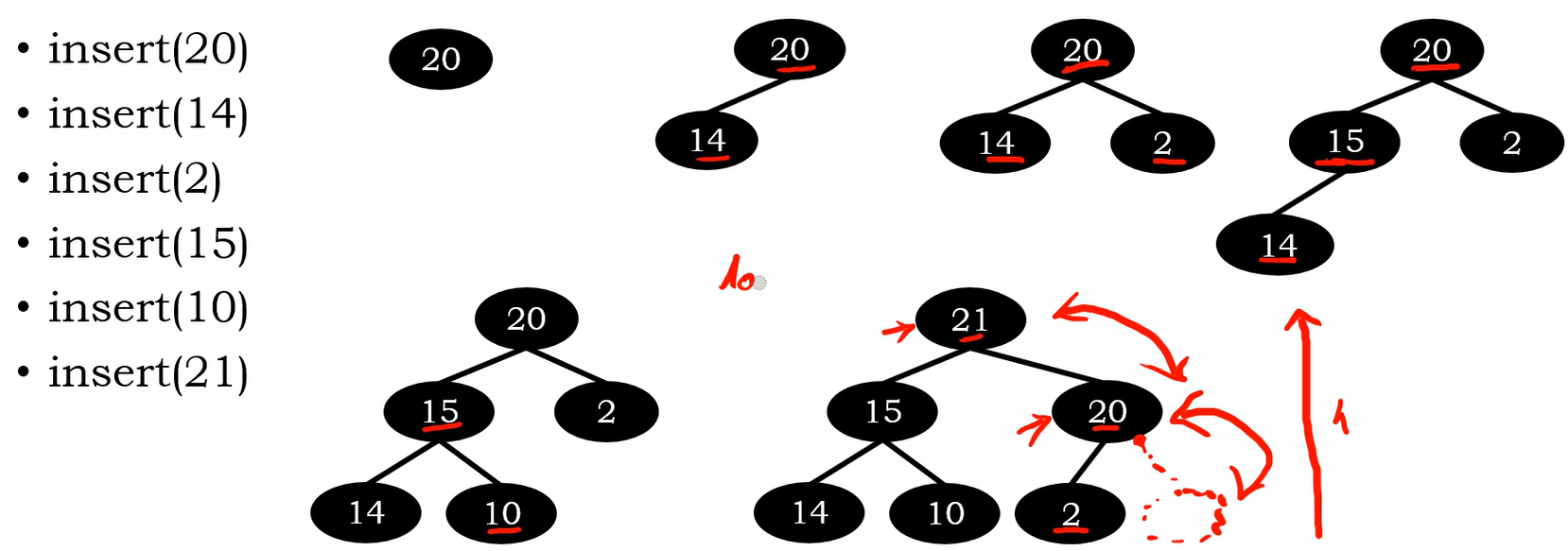
* Tập hợp các đối tượng hoặc phần tử được lưu trữ dưới dạng cây nhị phân.
* Heap nhị phân.
* Tính chất quan hệ: Khóa trong mỗi nút của cây nhị phân lớn hơn hoặc bằng với khóa trong các nút con của nó.
* Tính chất cấu trúc: Cây nhị phân phải là cây nhị phân hoàn chỉnh.
* Heap tối đa và Heap tối thiểu.

**12.3 Kiểu dữ liệu Abstract của Heap:**

* Thành Phần:
* Kích thước tối đa.
* Kích thước hiện tại.
* Toán Tử:
* Insert(object): chèn 1 phần tử vào heap.
* DeleteMax(object): Xóa và trả về phần tử lớn nhất bên trong Heap.
* Max(object): Trả về phần tử lớn nhất trong Heap.

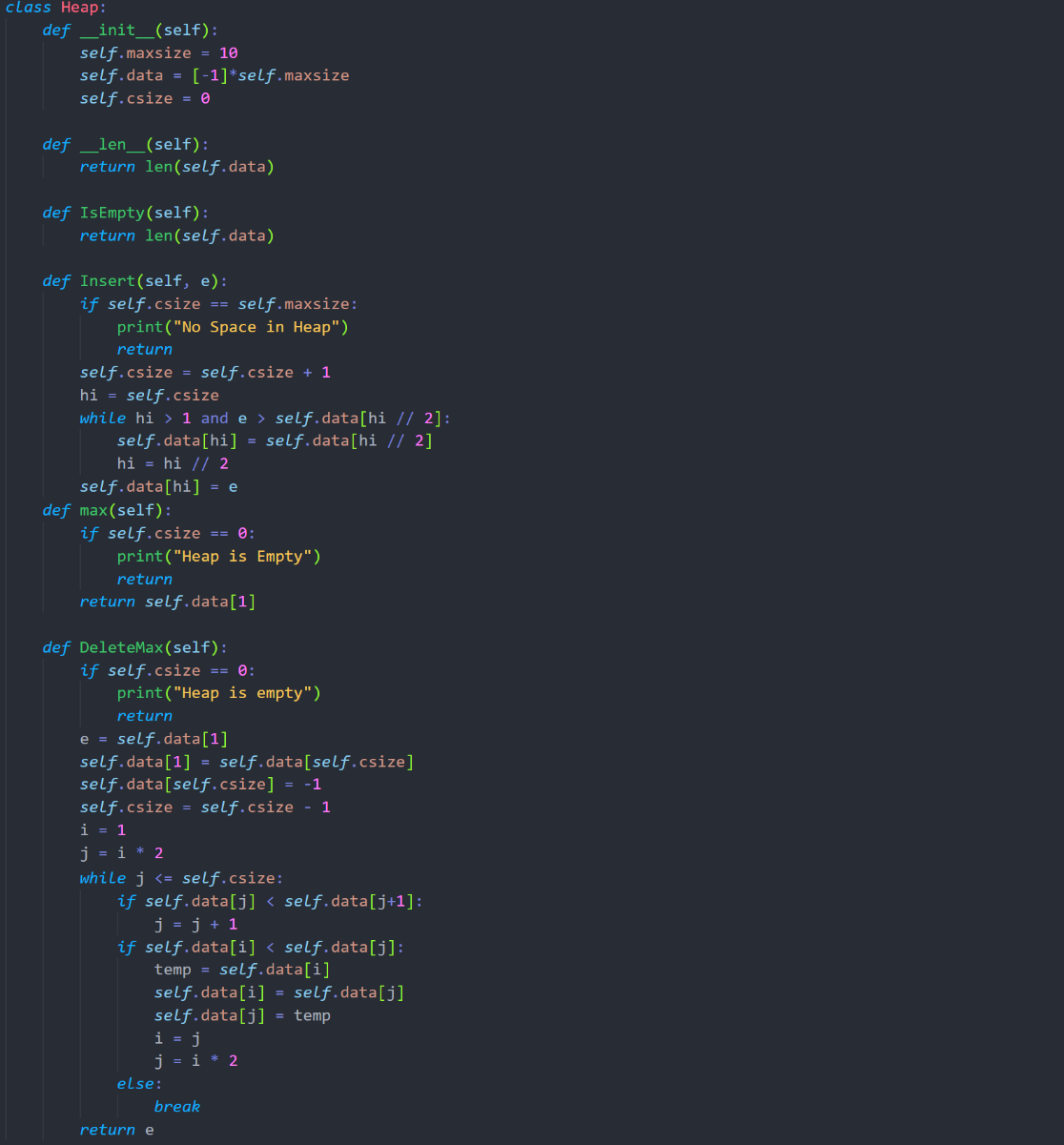
**12.4 Heap Insertion:**

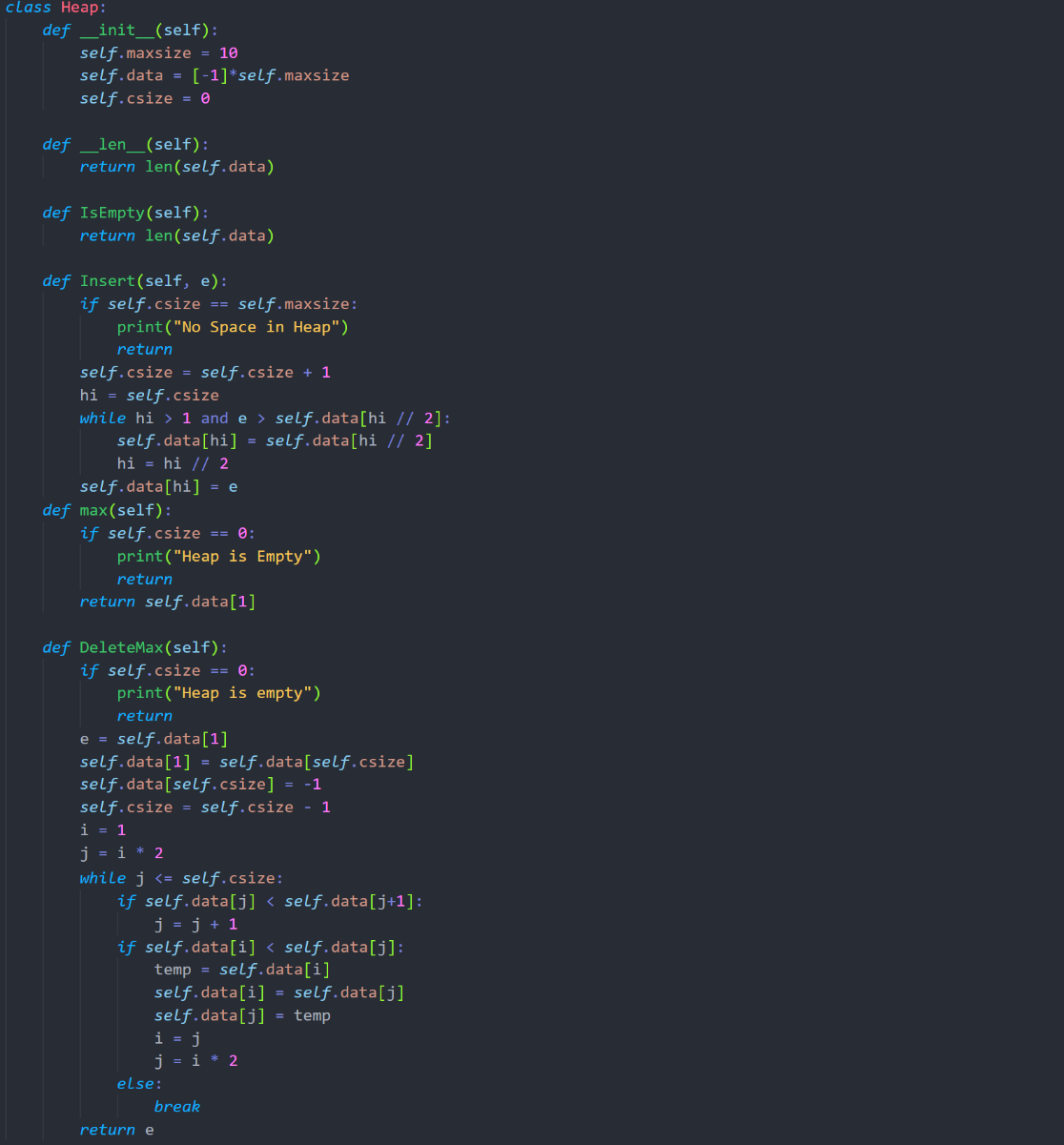
* Phần tử được chèn làm nút mới trong cây
* Tính chất cấu trúc: Nút mới được chèn sau nút cuối cùng
* Tính chất quan hệ: Thực hiện di chuyển lên trên (up-heap bubbling)



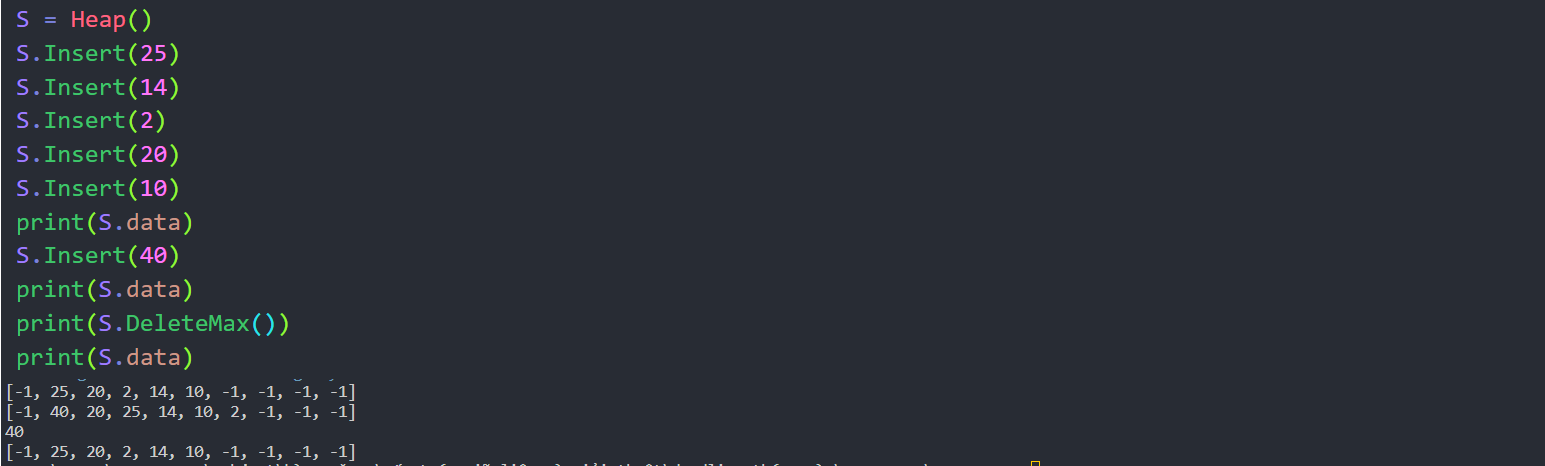
**12.5 Heap Deletetion**

Xây dựng Heap (Python):



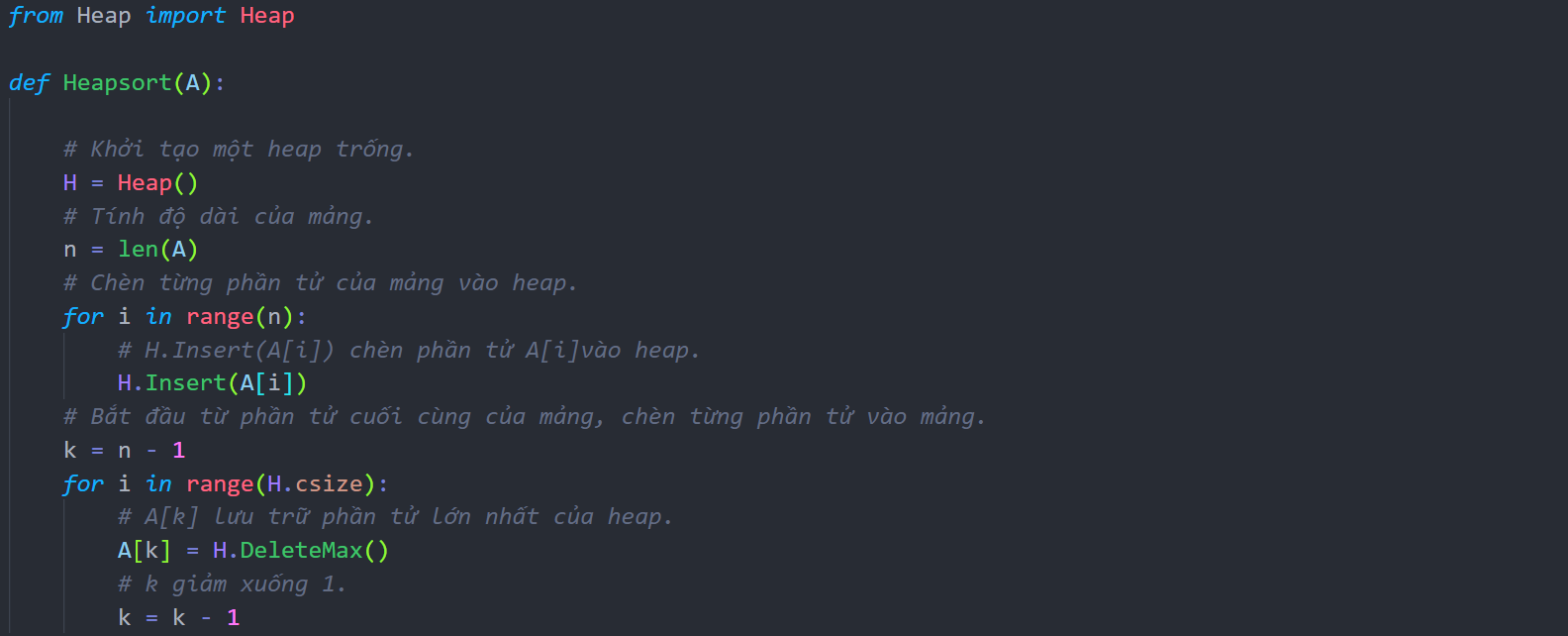


Thực Thi và Kết Quả:



**12.6 Heap Sort**

* Nguyên tắt:
* Sử dụng cấu trúc dữ liệu Heaps.
* Chèn các phần tử vào heap.
* Thực hiện xóa cho đến khi heap trống.
* Lưu trữ các phần tử đã xóa từ heap trở lại vào mảng.
* Độ phức tạp của thuật toán: O(nlog(n)):
* Xây dựng thuật toán Heap Sort (Python):



* Thực Thi và Kết Qủa:

