

Priority queue

```
class Node:  
    def __init__(self, info, priority):  
        self.info = info  
        self.priority = priority  
class PriorityQueue:  
    def __init__(self):  
        self.queue = list()  
    def insert(self, node):  
        if self.size() == 0:  
            # Add the new node if the queue is empty  
            self.queue.append(node)  
        else:  
            # Traverse the queue to find the right place for the new node  
            for x in range(0, self.size()):  
                if node.priority >= self.queue[x].priority:  
                    # If we have traversed the complete queue  
                    if x == (self.size() - 1):  
                        # Add the new node at the end  
                        self.queue.insert(x + 1, node)  
                    else:  
                        continue  
                else:  
                    self.queue.insert(x, node)  
        return True  
    def delete(self):  
        return self.queue.pop(0)  
    def show(self):  
        for x in self.queue:  
            print(str(x.info) + " - " + str(x.priority))  
    def size(self):  
        return len(self.queue)  
  
pQueue = PriorityQueue()  
  
node1 = Node("C", 3)  
node2 = Node("B", 2)  
node3 = Node("A", 1)  
node4 = Node("Z", 26)  
node5 = Node("Y", 25)  
node6 = Node("L", 12)  
  
pQueue.insert(node1)  
pQueue.insert(node2)  
pQueue.insert(node3)  
pQueue.insert(node4)  
pQueue.insert(node5)  
pQueue.insert(node6)  
  
pQueue.show()  
  
print("-----")  
  
pQueue.delete()  
pQueue.show()
```