MAX-HEAP-INCREASE-KEY(A, x, k)

```
1  if k < x.key
2        error "new key is smaller than current key"
3  x.key = k
4  find the index i in array A where object x occurs
5  while i > 1 and A[PARENT(i)].key < A[i].key
6        exchange A[i] with A[PARENT(i)], updating the information that maps
               priority queue objects to array indices
7        i = PARENT(i)
```

MAX-HEAP-INSERT(A, x, n)

```
1  if A.heap-size == n
2        error "heap overflow"
3  A.heap-size = A.heap-size + 1
4  k = x.key
5  x.key = −∞
6  A[A.heap-size] = x
7  map x to index heap-size in the array
8  MAX-HEAP-INCREASE-KEY(A, x, k)
```