Indexed priority queue

Associate an index between 0 and $N - 1$ with each key in a priority queue.

- Client can insert and delete-the-minimum.
- Client can change the key by specifying the index.

public class IndexMinPQ<Key extends Comparable<Key>>	
IndexMinPQ(int N)	*create indexed priority queue with indices 0, 1, ..., N-1*
void insert(int i, Key key)	*associate key with index i*
void decreaseKey(int i, Key key)	*decrease the key associated with index i*
boolean contains(int i)	*is i an index on the priority queue?*
int delMin()	*remove a minimal key and return its associated index*
boolean isEmpty()	*is the priority queue empty?*
int size()	*number of entries in the priority queue*

Indexed priority queue implementation

Implementation.

- Start with same code as `MinPQ`.
- Maintain parallel arrays `keys[]`, `pq[]`, and `qp[]` so that:
  - `keys[i]` is the priority of `i`
  - `pq[i]` is the index of the key in heap position `i`
  - `qp[i]` is the heap position of the key with index `i`
- Use `swim(qp[i])` implement `decreaseKey(i, key)`.