# **Double-Ended Priority Queues**

- Primary operations
  - Insert
  - Remove Max
  - Remove Min
- Note that a single-ended priority queue

### General Methods

- Dual min and max single-ended priority queues.
- Correspondence based min and max single -ended priority queues.

### Specialized Structures

- Min-max heaps.
- Deaps.
- Interval heaps.

# Dual Single-Ended Priority Queues

- Each element is in both a min and a max single-ended priority queue.
- Single-ended priority queue also must support an arbitrary remove.
- Each node in a priority queue has a pointer to the node in the other priority queue that has the same element.

### 9-Element Example

- Only 5 of 9 two-way pointers shown.
- Insert, remove min, remove max, initialize.
- Oion cost is more thTT1 1 48n doubl28 0 Tm d relTT1 1

# Correspondence Structures

- Use a min and a max single-ended priority queue.
- At most 1 element is in a buffer.
- Remaining elements are in the single-ended priority queues, which may be of different size.
- No element is in both the min and max single-ended priority queue.
- Establish a correspondence between the min and max single-ended priority queues.
  - Total correspondence.
  - Leaf correspondence.
- Single-ended priority queue also must support an arbitrary remove.

# Total Correspondence

- The min- and max-priority queues are of the same size.
- Each element of the min priority queue is paired with a different and >= element in the max priority queue.

### Total Correspondence Example



Min Heap

Max Heap

#### Insert



Min Heap

Max Heap

- Buffer empty => place in buffer.
- Else, insert smaller of new and buffer elements into min priority queue and larger into max priority queue; establish correspondence between the 2 elements.

### **Remove Min**



Min Heap

Max Heap

- Buffer is min => empty buffer.
- Else, remove min from min PQ as well as corresponding element from max PQ; reinsert corresponding element.

# Leaf Correspondence

- Min- and max-priority queues may have different size.
- Each leaf element of the min priority queue is paired with a different and >= element in the max priority queue.
- Each leaf element of the max priority queue is paired with a different and <= element in the min priority queue.

## Added Restrictions

- When an element is inserted into a single-ended PQ, only the newly inserted element can become a new leaf.
- When an element is deleted from a single-ended PQ, only the parent of the deleted element can become a new leaf.
- Min and max heaps do not satisfy these restrictions. So, leaf correspondence is harder to implement using min and max heaps.

### Leaf Correspondence Example



Min Heap

Max Heap





 Case when min and/or max heap originally have an even number of elements is more involved, because a nonleaf may become a leaf. See reference.

### **Remove Min**



- Buffer is min => empty buffer.
- Else, remove min from min PQ as well as corresponding leaf element (if any) from max PQ; reinsert removed corresponding element (see reference for details).