Priority Queues

Priority + Queue

Outline

- Basics of *Priority Queues*
- Heap
- Heap Sort
- Min-max Heap

Basics of *Priority Queues*

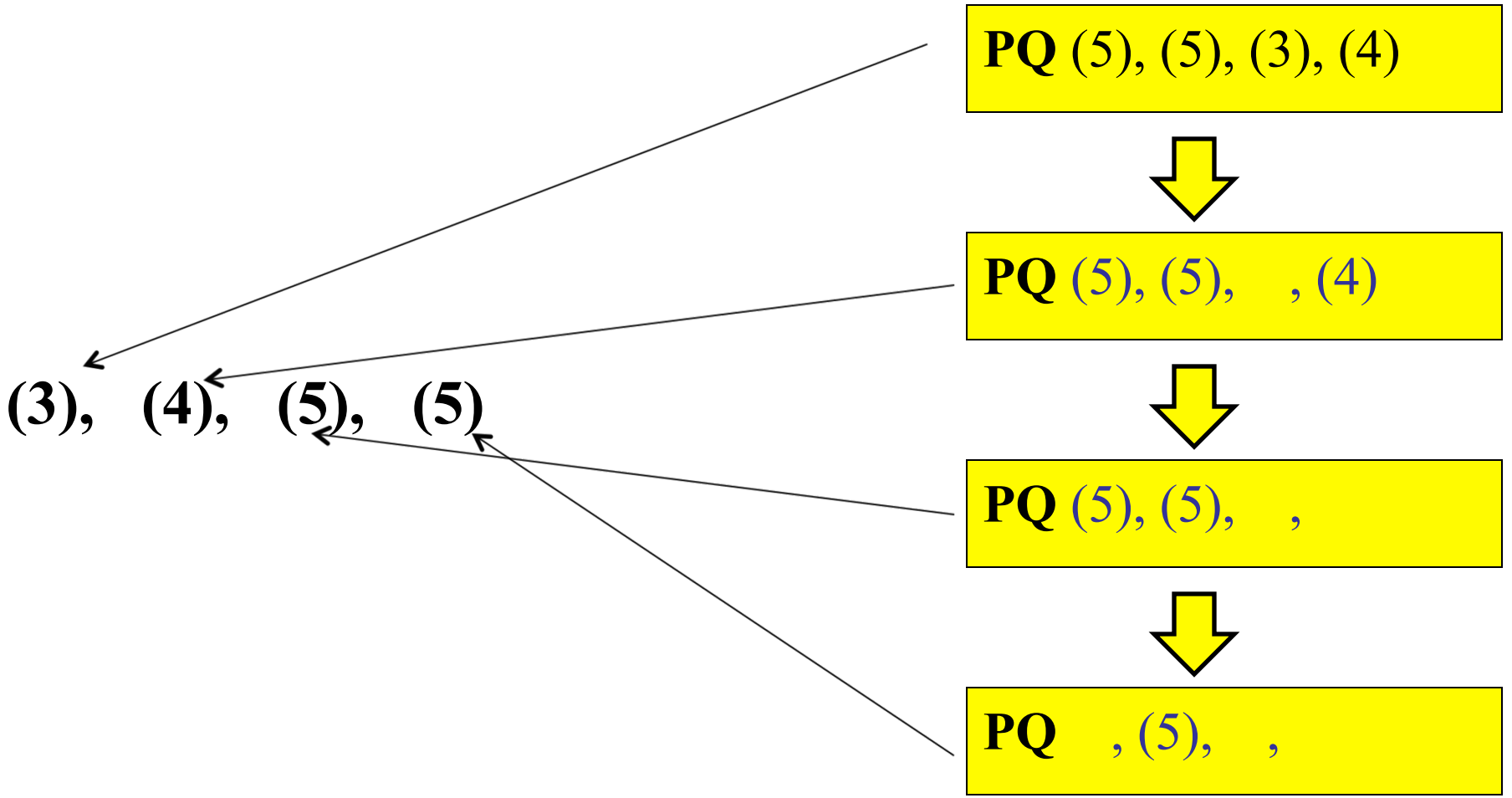
PQ (P1, 5, 23:55), (P2, 5, 00:05), (P3, **3**, 00:10), (P4, 4, 00:30)

PQ (P1, 5, 23:55), (P2, 5, 00:05), (P4, **4**, 00:30)

PQ (P1, **5**, **23:55**), (P2, 5, 00:05)

PQ (P2, 5, 00:05)

Basics of *Priority Queues*



Basics of *Priority Queues*

(P1, 5, 23:55), (P2, 5, 00:05), (P3, 3, 00:10), (P4, 4, 00:30)

pqInsert(dataItem, *priority*)

PQ

pqDelete() or called *Pull()*

(P3, 3, 00:10) (P4, 4, 00:30) (P1, 5, 23:55) (P2, 5, 00:05)

Basics of *Priority Queues*

(P1, 5, 23:55), (P2, 5, 00:05), (P3, 3, 00:10), (P4, 4, 00:30)

pqInsert(dataItem, *priority*)

Sorting Algorithm?

pqDelete() or called *pull()*

(P3, 3, 00:10) (P4, 4, 00:30) (P1, 5, 23:55) (P2, 5, 00:05)

Basics of *Priority Queues*

Sorting Algorithm

	Worst case	Average case
Selection sort	n^2	n^2
Bubble sort	n^2	n^2
Insertion sort	n^2	n^2
Mergesort	$n * \log n$	$n * \log n$
Quicksort	n^2	$n * \log n$
Radix sort	n	n

Basics of *Priority Queues*

(P1, 5, 23:55), (P2, 5, 00:05), (P3, 3, 00:10), (P4, 4, 00:30)

pqInsert(): O(1)

Selection Sort: Unsorted List

pqDelete(): O(n)

(P3, 3, 00:10) (P4, 4, 00:30) (P1, 5, 23:55) (P2, 5, 00:05)

Basics of *Priority Queues*

Sorting Algorithm

	Worst case	Average case
Selection sort	n^2	n^2
Bubble sort	n^2	n^2
Insertion sort	n^2	n^2
Mergesort	$n * \log n$	$n * \log n$
Quicksort	n^2	$n * \log n$
Radix sort	n	n

Basics of *Priority Queues*

(P1, 5, 23:55), (P2, 5, 00:05), (P3, 3, 00:10), (P4, 4, 00:30)

pqInsert(): O(n)

Insertion Sort: Sorted List

pqDelete(): O(1)

(P3, 3, 00:10) (P4, 4, 00:30) (P1, 5, 23:55) (P2, 5, 00:05)

Basics of *Priority Queues*

(P1, 5, 23:55), (P2, 5, 00:05), (P3, 3, 00:10), (P4, 4, 00:30)

pqInsert()

Other Sorting Algorithm?

pqDelete()

(P3, 3, 00:10) (P4, 4, 00:30) (P1, 5, 23:55) (P2, 5, 00:05)

Basics of *Priority Queues*

(P1, 5, 23:55), (P2, 5, 00:05), (P3, 3, 00:10), (P4, 4, 00:30)

pqInsert()

Tree Sort?

pqDelete()

(P3, 3, 00:10) (P4, 4, 00:30) (P1, 5, 23:55) (P2, 5, 00:05)

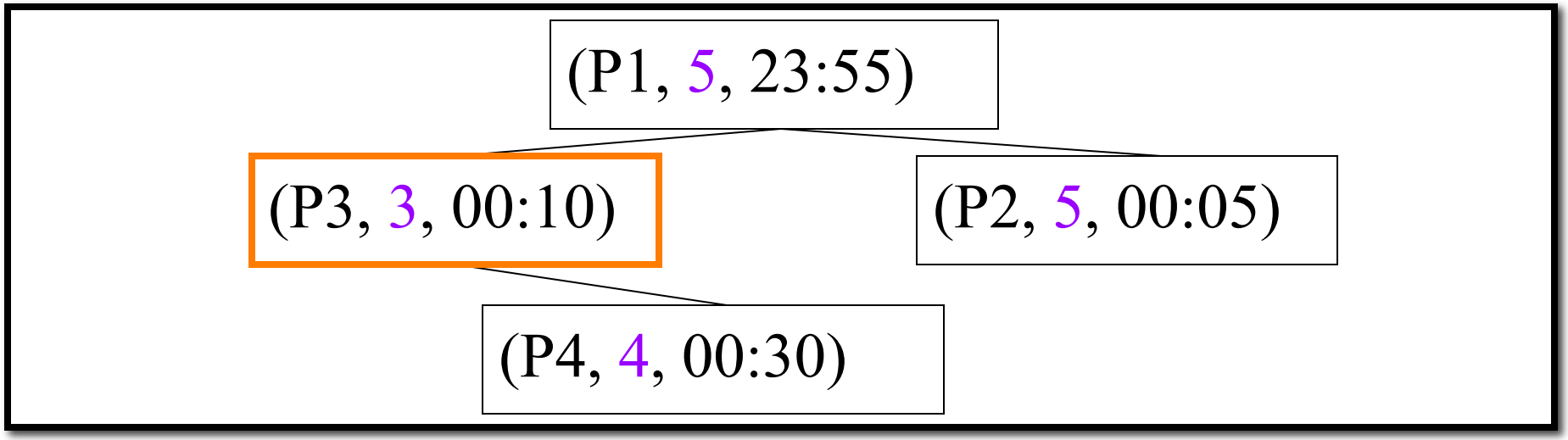
Basics of *Priority Queues*

(P1, 5, 23:55), (P2, 5, 00:05), (P3, 3, 00:10), (P4, 4, 00:30)

pqInsert(): O(?)

Tree Sort: Binary Search Tree

pqDelete(): O(?)



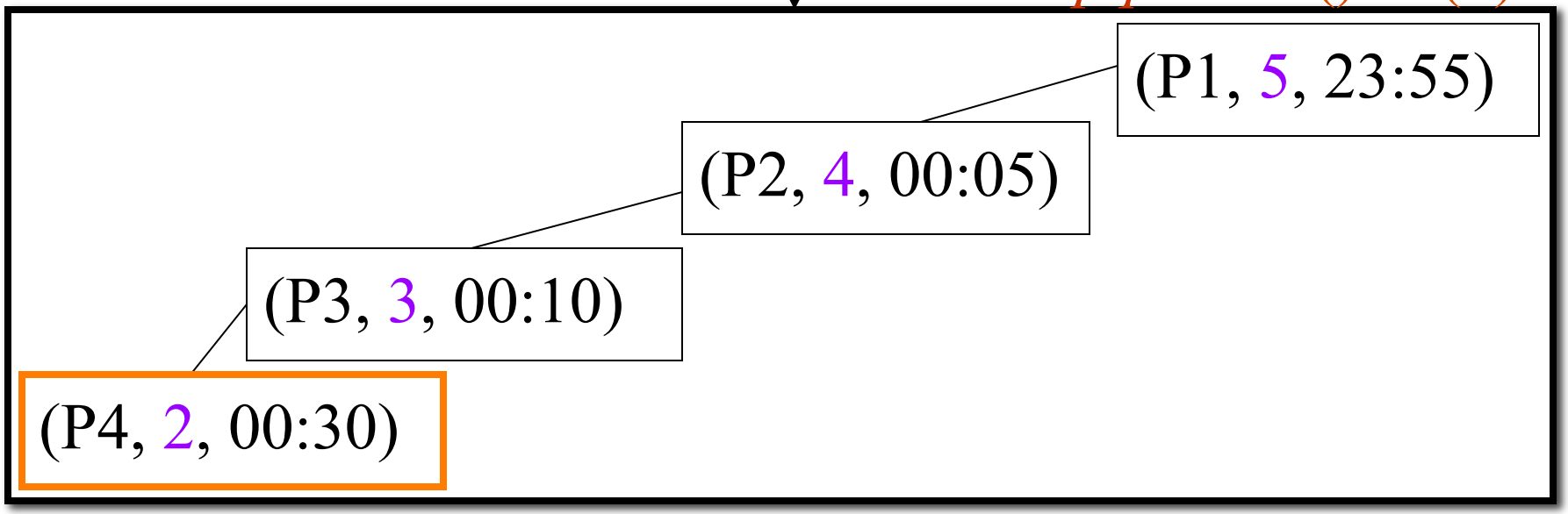
Basics of *Priority Queues*

(P1, 5, 23:55), (P2, 4, 00:05), (P3, 3, 00:10), (P4, 2, 00:30)

pqInsert(): O(n)

Tree Sort: Binary Search Tree

pqDelete(): O(n)



Heap

(P1, 5, 23:55), (P2, 4, 00:05), (P3, 3, 00:10), (P4, 2, 00:30)

pqInsert(): < O(n)

Heap!

pqDelete(): < O(n)