Sorting with a PQ

• Key ordering must satisfy *strict weak ordering*:
  • Irreflexive property
    • key can’t be less than nor greater than itself
    • could arise if keys come from unordered sets (rare?)
  • Transitive property
    • if key1 < key2 and key2 < key3, then key1 < key3
    • aka the “no rock, paper, scissors rule”
Different PQs, Different Sorts

- `priority_queue_sorts.py`
- **unsorted** -- selection sort
  - sort by selecting out of queue
  - O(n**2)
- **sorted** -- insertion sort
  - sort by inserting into queue
  - O(n**2), but dumb luck could give O(n)
- **heap** -- heap sort
  - O(n log n)
- (demos on board)
Adaptable Priority Queues

- Modifiable priorities
  - maximizing or minimizing a window
  - bribing door man (cough)

- `adaptable_heap_priority_queue.py`
  - `Locator`
    - Generalization of `Position`, can encapsulate references/pointers or array indices.
  - update and remove `Locator` objects
    - priority queue state is handled invisibly to the user.