Sorted Maps

- Unsorted table maps and hash maps
  - fine for *exact searches*
  - not fine for printing sorted data

- Sorted table map
  - store keys in increasing order in bucket array
    - same idea as sorted list priority queue (insertion sort)
  - allows use of *binary search*
    - (example on whiteboard)
  - `binary_search.py`
Sorted Maps, cont’d.

• $O(\log n)$ for many operations
  • (complexity table in notes/textbook)
• `sorted_table_map.py`
• Now also able to easily find range of values
  • flights between dates
  • hotels between dates
  • etc.
Sets, MultiSets, and Multimaps

• **Set**: unordered, unique elements
  • map with just keys

• **Multiset**: unordered elements, aka, bags
  • map with just keys and a counter for each key

• **Multimaps**: key can have multiple values
  • value is a container: list, tuple, dict, etc.
Lab 7 part 3

- Why “in-place?”
- Why “max-heap?”
Skip Lists (optional)

• Alternative for sorted map
  • improves on linear insertion/deletion
• Can think of as sparse 2D matrix where 0 entries are passed over instantaneously
• “towers” built randomly giving good probability that a search will skip down efficiently
• (diagram and complexity table in notes/textbook)