Motivation

- Have already seen binary search - fast to search through data if it is sorted
- Many other applications to sorting as well
- Example:
  - Unsorted: A = [1 3 2 4 3 5]
  - Sorted: A = [1 2 3 3 4 5]
- There are both simple and complex ways to sort, will see that it is generally possible in O(n log n) time

Sorting methods

- Naive methods - $O(n^2)$:
  - Bubble sort, Selection sort, Insertion sort
- Divide and conquer - O(n log n):
  - Merge sort, Quick sort
- Linear (depend on characteristics of data type) - close to O(n):
  - Counting sort, Bucket sort, Radix sort
- Hybrid methods - Timsort (used in Python)