Lecture 21: Algorithms

CS 51P

November 27, 2023

al·go·rithm

/ˈalgəˌriT͟Həm/

noun

a process or set of rules to be followed in calculations or other problem-solving operations, especially by a computer.

"a basic **algorithm for** division"

Example: Sorting

Selection sort

- For each position in the list
  - Find the object that should be there
  - Put the object in the position

- Visualization: https://visualgo.net/bn/sorting

- Code in demo21.py

Insertion sort

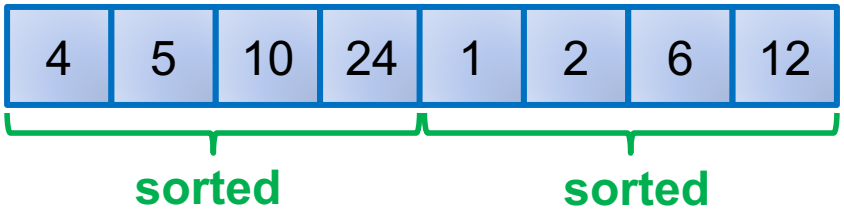
- For each object in the list
  - Compare with the object before it
  - Move it to the right position

- Visualization: https://visualgo.net/bn/sorting

- Code in demo21.py

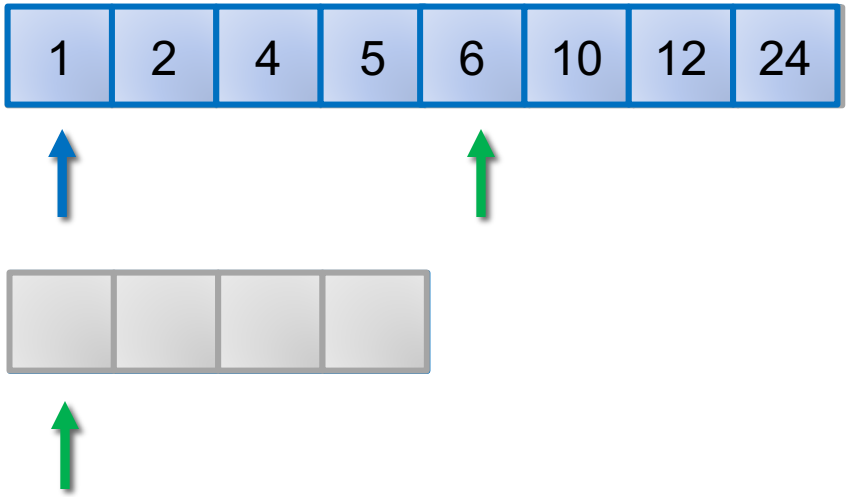
Merging

- What if our list looked like two sorted lists end to end?
- We could sort by merging the two lists!

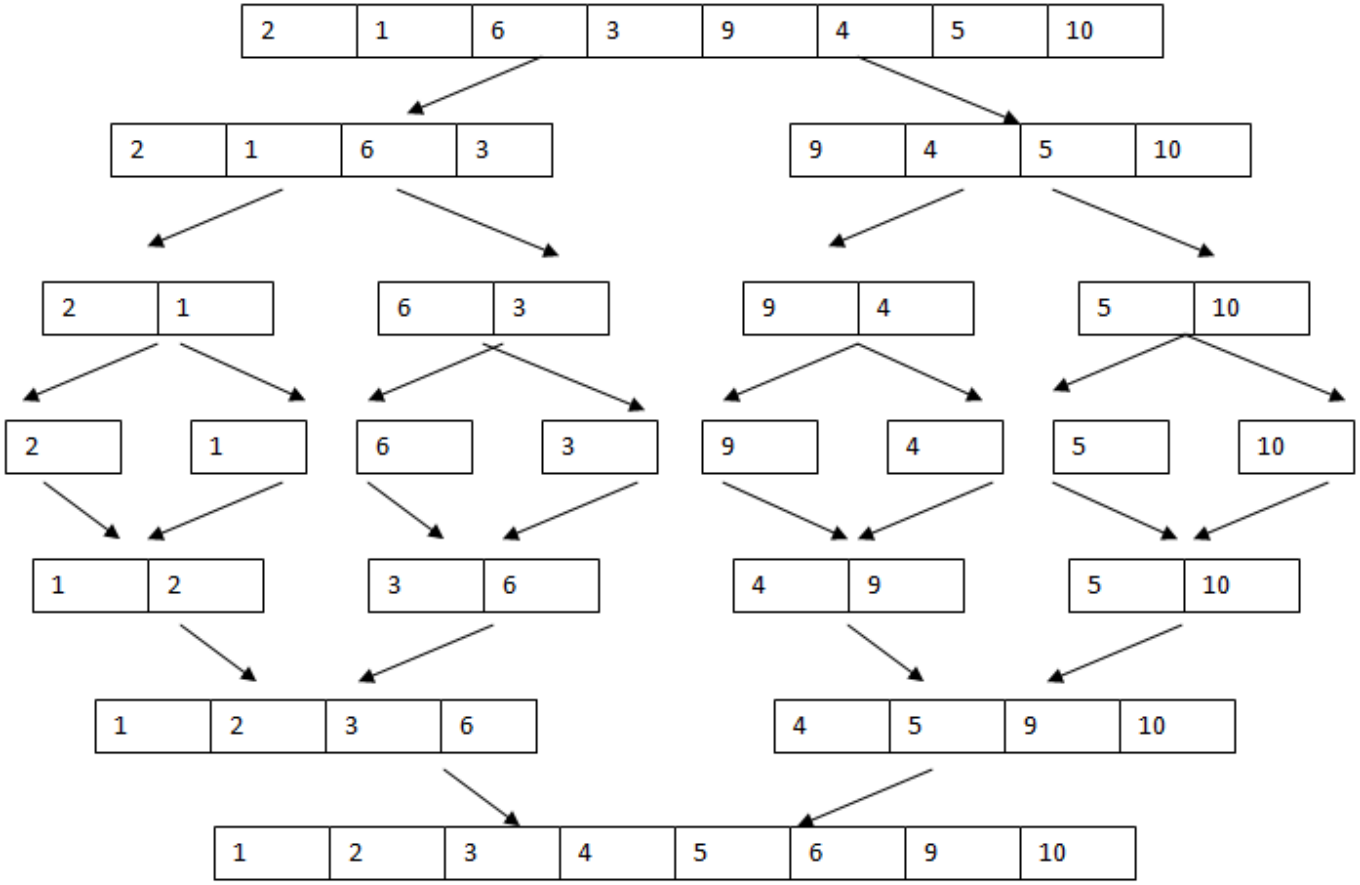


Merging

- What if our list looked like two sorted lists end to end?
- We could sort by merging the two lists!



Mergesort



- Recursively:
  - Sort the first half of the list
  - Sort the second half of the list
  - Merge the two halves together

- Code in demo21.py
- Exercise 21

Sorting Algorithms

Selection Sort

```
def selection_sort(l):

  # for each pos in list
  for pos in range(len(l)):

    # find obj
    min_index =
    for i in ra
      if l[i] <
        min_ind

    # swap that
    (l[pos], l[
        (l[min_
```

Insertion Sort

```
def insertion_sort(l):

  # for each obj in list
  for pos in range(len(l)):

    rect position

    0 and
    [curr_pos-1]:
    l[curr_pos-1])=\
    -1],l[curr_pos])
    _pos - 1
```

Merge Sort

```
def merge_sort_helper(lst, start, end):
    # Base Case
    if end-start < 2:
        return
    # Recursive Case
    else:
        middle = start + int((end-start) / 2)
        le)
        merge(lst, start, end)
```

Which algorithm is better?

Announcement

- Final project
  - Discuss the proposed questions with course instructors before implementation
- Checkpoint 3
  - 12/6/2023
  - Oct. 18 - File IO through Nov. 30, Algorithm analysis