

# Searching and Sorting

Hui Chen

Department of Computer & Information Science

Brooklyn College

# Objectives

- To search elements using the linear (§7.10.1) or binary (§7.10.2) search algorithm.
- To sort an array using the selection sort approach (§7.11)

# Searching Arrays

- Searching is the process of looking for a specific element in an array
- Example
  - Discovering whether a certain score is included in a list of scores.

# Linear Search

- The linear search approach compares the key element, key, *sequentially* with each element in the array list.

# Design Linear Search Method

- The method continues to do so until the key matches an element in the list or the list is exhausted without a match being found.
- If a match is made, the linear search returns the index of the element in the array that matches the key.
- If no match is found, the search returns -1

# Linear Search Animation

- <https://liveexample.pearsoncmg.com/dsanimation/LinearSearchBook.html>

# Questions?

# Binary Search

- For binary search to work, the elements in the array must already be ordered.
- Without loss of generality, assume that the array is in ascending order
  - e.g., 2 4 7 10 11 45 50 59 60 66 69 70 79
- The binary search first compares the key with the element in the middle of the array, then we have three cases



# Binary Search, Continued

- If the key is less than the middle element, you only need to search the key in the first half of the array
- If the key is equal to the middle element, the search ends with a match
- If the key is greater than the middle element, you only need to search the key in the second half of the array

# Binary Search Animation

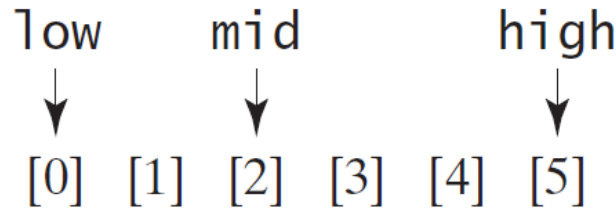
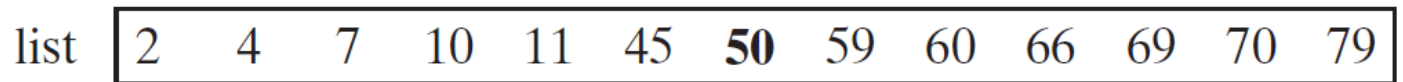
- <https://liveexample.pearsoncmg.com/dsanimation/BinarySearchBook.html>

# Implementing Binary Search

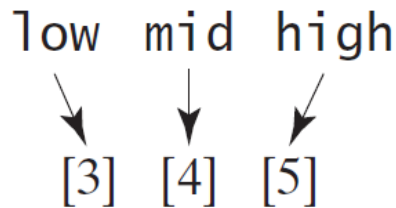
key is 11



key < 50



key > 7



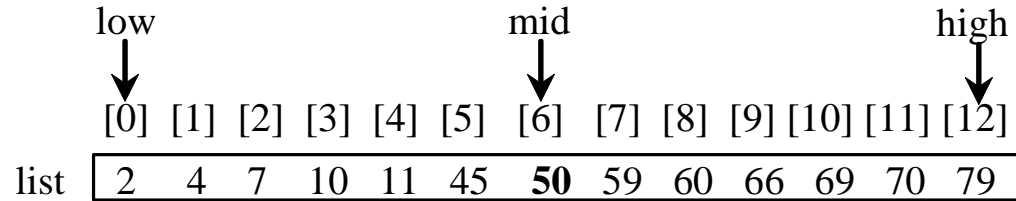
key == 11



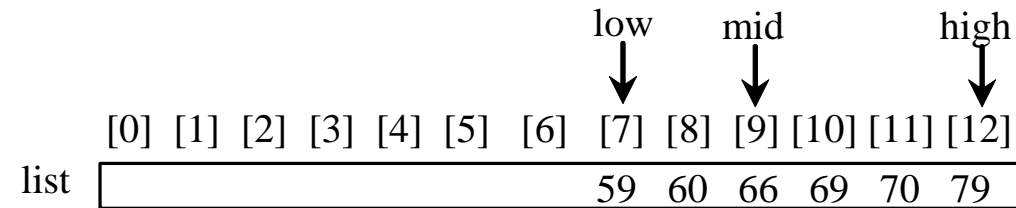
# Implementing Binary Search, Continued

key is 54

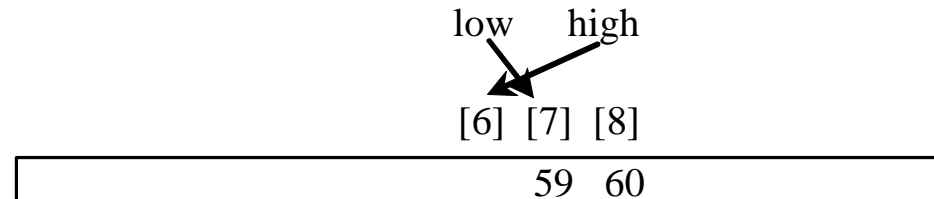
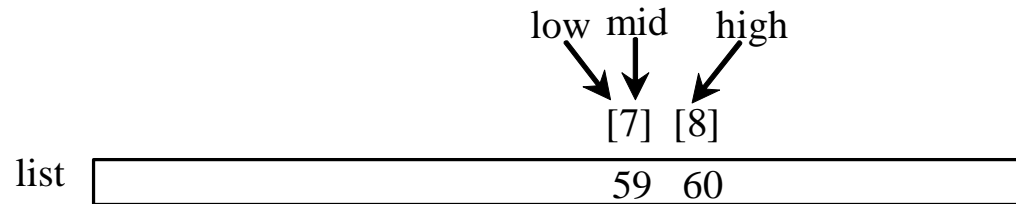
key > 50



key < 66



key < 59



# Implementing Binary Search, Continued

- The `binarySearch` method returns the index of the element in the list that matches the search key if it is contained in the list. Otherwise, it returns
  - insertion point – 1
- The insertion point is the point at which the key would be inserted into the list

# Questions?

# Sorting Arrays

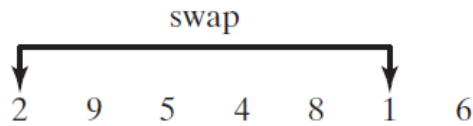
- Selection sort
- Bubble sort
- How efficient are they?  
([https://youtu.be/k4RRi\\_ntQc8](https://youtu.be/k4RRi_ntQc8))

# Selection Sort

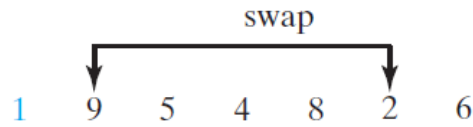
- Selection sort finds the smallest number in the list and places it first.
- It then finds the smallest number remaining and places it second, and so on until the list contains only a single number



Select 1 (the smallest) and swap it with 2 (the first) in the list.

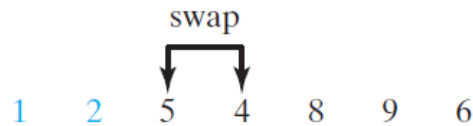


The number 1 is now in the correct position and thus no longer needs to be considered.



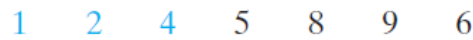
Select 2 (the smallest) and swap it with 9 (the first) in the remaining list.

The number 2 is now in the correct position and thus no longer needs to be considered.



Select 4 (the smallest) and swap it with 5 (the first) in the remaining list.

The number 4 is now in the correct position and thus no longer needs to be considered.



5 is the smallest and in the right position. No swap is necessary.

The number 5 is now in the correct position and thus no longer needs to be considered.



Select 6 (the smallest) and swap it with 8 (the first) in the remaining list.

The number 6 is now in the correct position and thus no longer needs to be considered.



Select 8 (the smallest) and swap it with 9 (the first) in the remaining list.

The number 8 is now in the correct position and thus no longer needs to be considered.



Since there is only one element remaining in the list, the sort is completed.

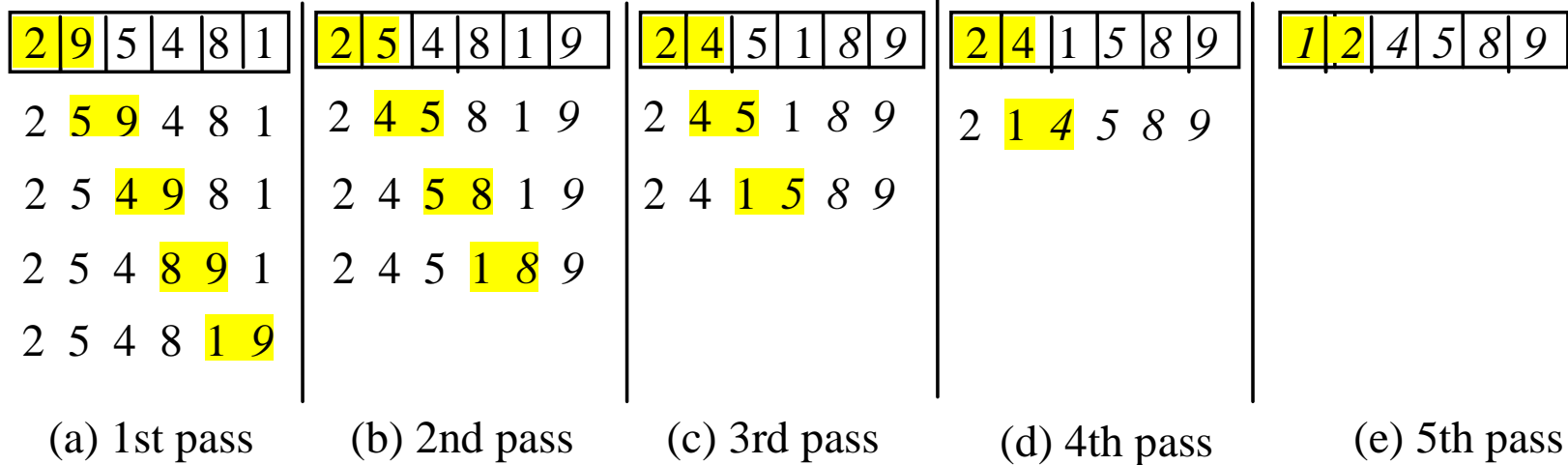
# Selection Sort Animation

- <https://liveexample.pearsoncmg.com/dsanimation/SelectionSortNew.html>

# Bubble Sort

- It makes passes through the array. On each pass,
  - compare successive neighboring
  - swap the values if they are not in order
- Observe the example in next slide

# An Example of Bubble Sort



# Bubble Sort Animation

<https://liveexample.pearsoncmg.com/dsanimation/BubbleSortNeweBook.html>

# Questions

# Sorting of Parallel Arrays

- We sometimes refer to sorting multiple parallel arrays together as parallel sort.
- Observe an example

# Questions?