Algorithms and Data Structures 1
CS 0445

Fall 2022

Sherif Khattab

ksm73@pitt.edu

(Slides are adapted from Dr. Ramirez's and Dr. Farnan's CS 0445 slides.)

Announcements

- Upcoming Deadlines:

  - **Assignment 2: Monday 11/7 @ 11:59 pm**

  - Lab 7: next Monday 11/7 @ 11:59 pm

  - Midterm reattempts: Thursday 11/10 @ 11:59 pm

- Live Support Session for Assignment 2

  - Video and slides available on Canvas

- QA Session on Piazza every Friday 4:30-5:30 pm

Today …

- Sorting Algorithms

Lab 7

- Please check Misc.java in the code handouts repository

Word Search Problem

- Please check WordSearch.java in the code handouts repository under the Recursion folder

Muddiest Points

- **Q: I am confused on assignment 2 runtime of push, to find the index where to push you need a loop which breaks O(1)**

- The loop is over the alphabet array. The alphabet size is assumed to be constant in this assignment

Muddiest Points

- **Q: When would you use proof by induction vs. a recursion tree to find the runtime of a recursive method?**

- You can either technique in most of the cases. I personally prefer the recursion tree approach.

Muddiest Points

- **Q: Towers of Hanoi is very confusing**

- Let's reiterate the problem

Muddiest Points

- **Since today was a muddiest point review a lot of things were clarified, but I would also like a refresher on proof by induction.**

- Please check the RecursionTimeComplexity.pdf on Canvas for more examples

Sorting

- We have seen a few container data structures

  - Bag, Stack, List

- Sorting Problem: arrange items in a List such that
  entry 1 ≤ entry 2 ≤ . . . ≤ entry n

- Efficiency of a sorting algorithm is significant

- Sorting an array is usually easier than sorting a chain of linked nodes

Sorting Algorithms

- $O(n^2)$

  - Selection Sort

  - Insertion Sort

  - Shell Sort

- $O(n \log n)$

  - Merge Sort

  - Quick Sort

- $O(n)$ Sorting

  - Radix Sort

Sorting Algorithms

- For each algorithm

  - understand the main concept using an example

  - implement the algorithm

    - on an Array

      - iterative
      - recursive

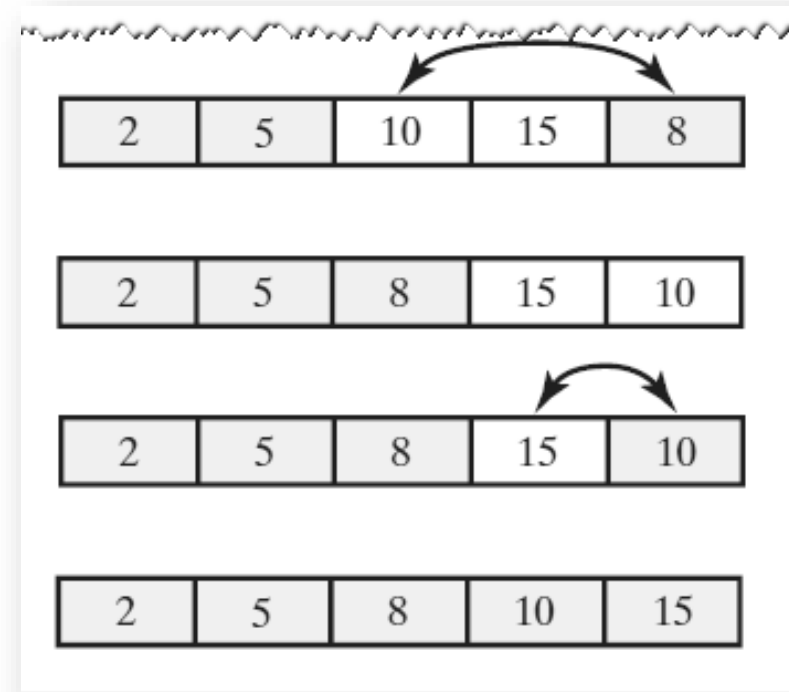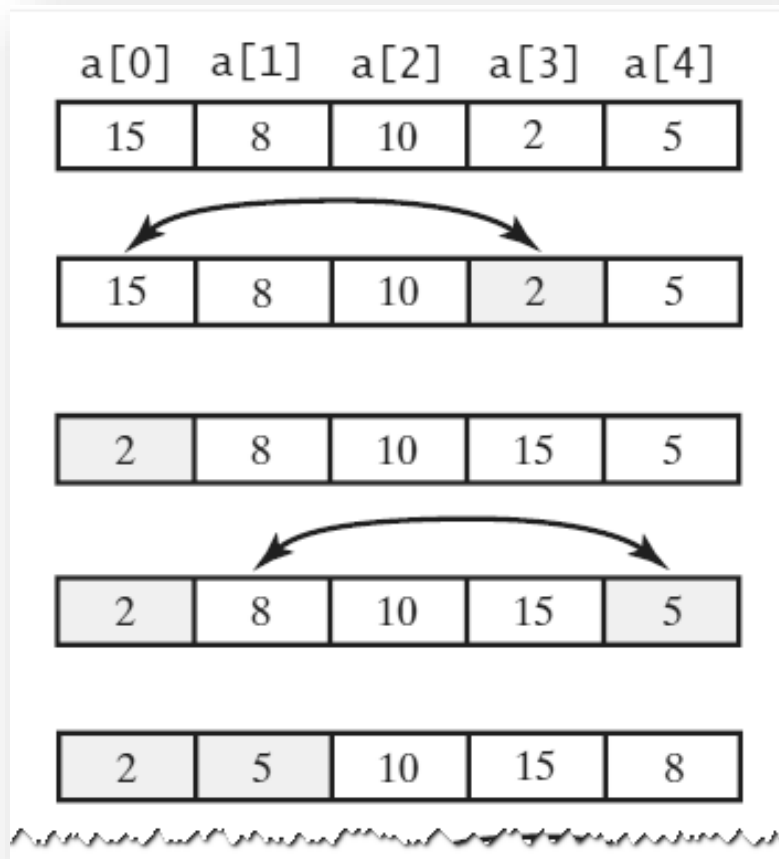   - on a linked list

      - iterative
      - recursive

Selection Sort

- Before and after exchanging
  the shortest book and the first book
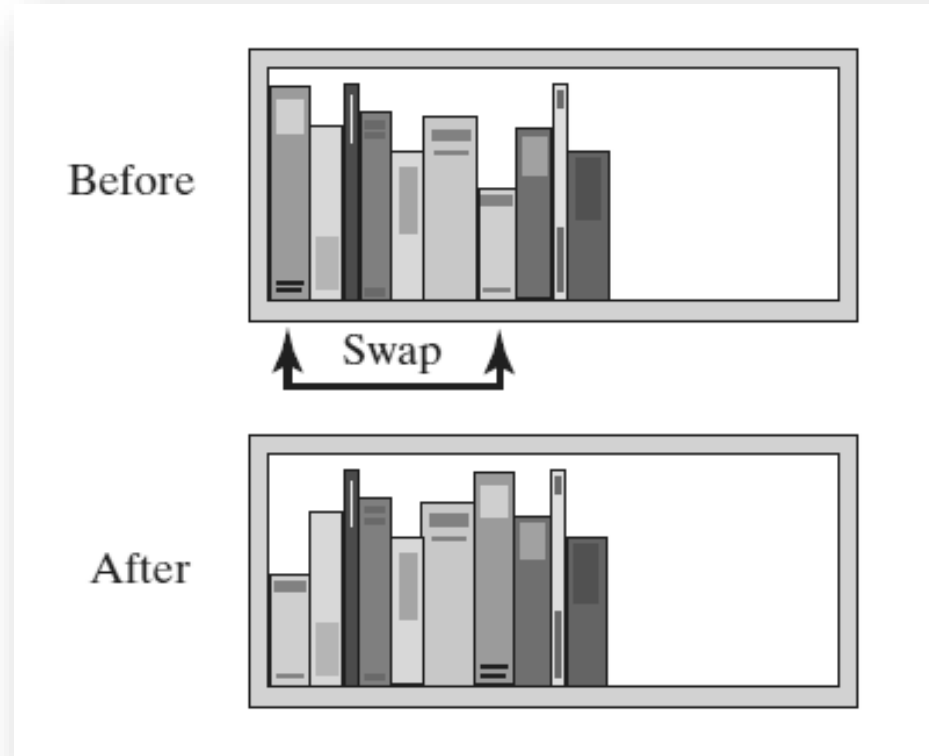
Selection Sort

- A selection sort of an array of integers into ascending order

Iterative Selection Sort

- This pseudocode describes an iterative algorithm for the selection sort

```
Algorithm selectionSort(a, n)
// Sorts the first n entries of an array a.

for (index = 0; index < n - 1; index++)
{
    indexOfNextSmallest = the index of the smallest value among
                          a[index], a[index + 1], . . . , a[n - 1]
    Interchange the values of a[index] and a[indexOfNextSmallest]
    // Assertion: a[0] ≤ a[1] ≤ . . . ≤ a[index], and these are the smallest
    // of the original array entries. The remaining array entries begin at a[index + 1].
}
```