

Sorting Algorithms



Share what you did over break!  

What we're going to learn

- Iterative Sorting Algorithms
 - Bubble
 - Selection
- Recursive Sorting Algorithms
 - Merge
 - Quick

Why Learn Sorting Algorithms?

- Sorting algorithms serve as excellent practice for algorithm analysis and design
- There are a ton of different ways to approach sorting so you can level up your problem solving skills
- Sorting questions will show up in technical interviews
- The important part isn't so much to memorize, it's to think critically and be able to create algorithms and analyze them

Where are sorting algorithms used in the real world?



Students, write your response!

Bubble Sort

Big Idea: Repeatedly Compare pairs of adjacent items, swap positions if they are in the wrong order

Called bubble sort because with this method items will move up into the correct order like bubbles rising to the surface of a boiling pot



Let's Visualize Bubble Sort

Bubble Sort with the power of Dance

Bubble Sort

1. Pull out 5 cards with different numbers and place them in unsorted order
2. Write pseudocode for the steps you would take to bubble sort this card deck
3. What would be the time complexity of the algorithm you came up with?
4. Prepare to present your groups findings to the rest of the class



Students, write your response!

Let's Turn It Into Code!



Students browse: repl.it/@JessDahmen/bubblesort?lite=true

Divide the list into two sections, the unsorted section and the sorted section

Initially the sorted section is empty

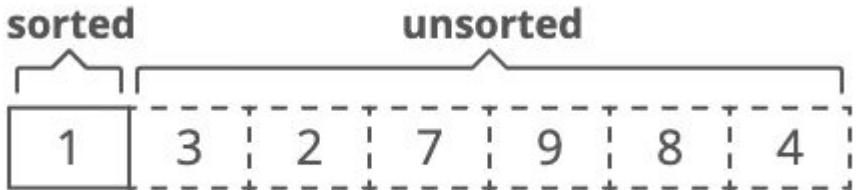
Select the smallest element in the unsorted section and move it to the front of the sorted section

Repeat until all items are sorted

Selection Sort



Selection Sort



Repeat!



Let's Visualize Selection Sort

Selection Sort

1. Pull out 5 cards with different numbers and place them in unsorted order
2. Write pseudocode for the steps you would take to sort these cards using selection sort
3. What would be the time complexity of the algorithm you came up with?
4. Prepare to present your groups findings to the rest of the class



Students, write your response!

Let's Turn it into Code



Students browse: repl.it/@JessDahmen/selectionsort?lite=true

Recursive Sorting

What are the two main components of a recursive algorithm?



Students choose an option

How would you describe how recursion works?



Students, write your response!

What is the purpose of the base case?



Students, write your response!

Recursion Review



Students browse: repl.it/@JessDahmen/isPalindrome?lite=true

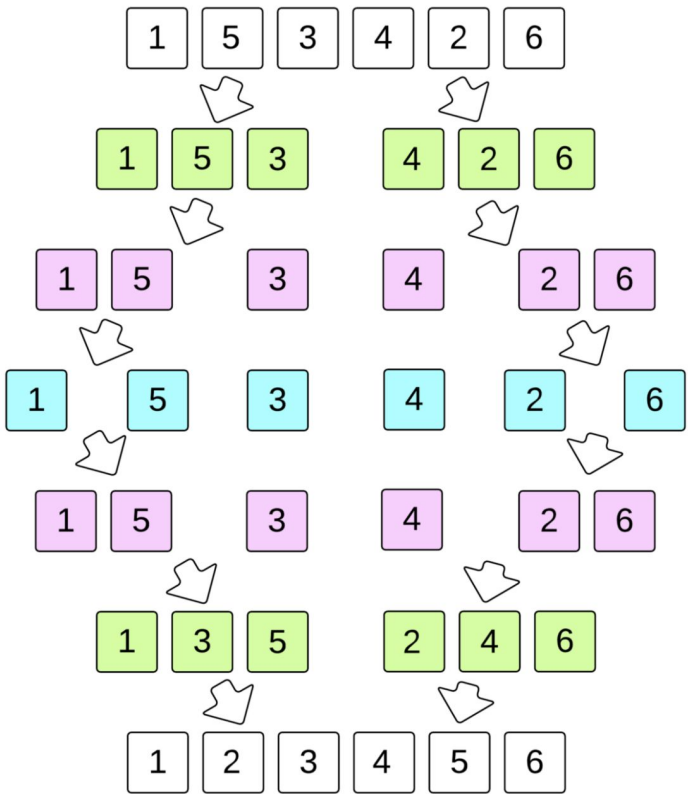
Divide and Conquer Algorithms

1. Breaks a problem into subproblems that are similar to the original problem
2. Recursively solves the subproblems
3. Combines the solutions to the subproblems to solve the original problem

1. Recursively break the problem of sorting the entire list into smaller and smaller subproblems
2. If the list only has one element in the list it is already sorted, return.
3. Divide the list recursively into two halves until it can no more be divided.
4. Merge the smaller lists into new list in sorted order

<https://medium.com/karuna-sehgal/a-simplified-explanation-of-merge-sort-77089fe03bb2>

Merge Sort



Let's visualize merge sort

**Write some pseudocode for
merge sort**

Use this card deck

Let's code merge sort!



Students browse: repl.it/@JessDahmen/mergesort?lite=true

Quick Sort

- Another example of a divide and conquer algorithm
- Can be solved recursively
- Uses a pivot as the basis of comparison
- Swap elements based on comparison to the pivot value
- Let's watch [this video](#)
- Check out [this resource](#)

Let's Visualize Quick Sort

Quick Sort With Cards

1. Work with your group to come up with pseudocode for the quicksort algorithm
2. Use [these cards](#) if needed
3. Choose someone from your group to share what you came up with



Students, write your response!

Let's Code Quick Sort



Students browse: repl.it/@MakeSchool/quicksort?lite=true