Lecture 34: How can computation sort data in order for you?

What is sorting?

This is not what we mean by sorting:
Not organizing or classifying into categories
What is sorting?

Put keys (and associated data) in specified order
- Ascending or descending
- Numerical or alphabetical

Sort keys and keep data with it
- Key: High score
  - Data: name
- Key: Employee id
  - Data: Name, Position, Phone, Salary
- Key: Search terms for web page
  - Data: URL, cached version, similar pages

Why is sorting important?

General:
  Can find what you want faster given sorted data

Sorted data is easier to search through
- Can apply binary search instead of exhaustive search

Trivial to find minimum and maximum elements
- First and last in list

Easy to find duplicate values
- Adjacent to each other in list

Easy to find patterns, anomalies, gaps
Many Different Sorting Algorithms

Today: Slow algorithms, but easy to understand
- Selection sort
- Insertion sort

Friday: Faster algorithms
- Mergesort
- Quicksort

Review:
How to find Min in List?

Loop through List using index variable

Input:
- List : Unsorted List

Output:
- Min
- Min index

Local variable:
- index

Robust to length of List
How can you sort?

How can you sort list of numbers?
- Move numbers into “sorted list”

Hint: Use your knowledge of how to find minimum element in a list

Algorithm 1: Selection Sort

To sort data...

Repeat until nothing in unsorted list:
- Find minimum element
- Add element to sorted list
- Delete from unsorted list
Selection Sort in Action

Unsorted list

Sorted list

How to Implement Selection Sort in Scratch?

Control code

Asks Sorter Sprite to do work

Create unsorted list

Get the list sorted

Check that the list really is sorted
Sorter Sprite: Helper Functions

Make List

- Sort List
- Delete 0 of Sorted List
- Repeat List Length

Check List

- when I receive Check List
- set List Sorted to 1
- set to 1
- set previous to item 0 of Sorted List
- repeat List Length
  - if item 0 of Sorted List < previous
  - set List Sorted to 0
  - set previous to item 0 of Sorted List
  - change by 0

Selection Sorter Sprite

Selection Sort: Two Lists
- Finds minimum remaining element in unsorted
- Adds to Sorted (in order)
- Deletes minimum from unsorted
Selection Sort: One List Demo

Unsorted list

Sorted list

Selection Sort: Do we need two Lists?
Selection Sort in Scratch: One List

Variable i:
Number of sorted elements

Variable j:
Looks for min of remaining unsorted elements

Algorithm 2: Insertion Sort

What algorithm do you use to sort cards?
Insertion Sort

Divide cards into two groups: sorted and unsorted

Initial state: 1 sorted card, N-1 unsorted

Repeat for all cards
  • Remove 1st card from unsorted portion
  • Insert into correct location in sorted list
    – Repeat loop
    – Keep moving down list until card to left < new card
  • Change definition of sorted vs. unsorted portions

Insertion Sort in Scratch

Repeat for all cards
  Take 1st unsorted card
  Insert into correct location in sorted list: Repeat loop
    Keep moving down list until card to left is smaller than new card
    OR at beginning of list
Which Sorting Algorithm is Best?

Compare number of loop iterations as function of N – size of input list

Previously analyzed searching algorithms
- Linear search: O(N) operations
- Binary search: O (log₂ N) operations

Selection Sort: How many loop iterations?

Selection Sort: Two Lists

Size of list: N

2 loops: Outer and inner

How many iterations of outer loop?
- N

How many iterations of inner loop?
- N, N-1, N-2, ... 1
- Average: N/2

Total?
- N * N/2

Complexity?
- O(N²)
Insertion Sort in Scratch

```
when I receive Sort List
  set i to 1
repeat until i = List Length
  change i by 1
  set j to i
  repeat until j = 1 or List[j] < List[i]
    # Sort item j of unsorted list and item j of unsorted list
    swap item j of unsorted list with item j of unsorted list
    replace item j of unsorted list with temp
    change j by -1
```

Outer loop?  
Always N

Inner loop - Worst case? Data in reverse order!  
1, 2, ….. N-3, N-2, N → N/2  
Best case? Data sorted already!  
0  
Average case?  
O(N²)

Today’s Summary

Intuitive but Slow Sorting

- Selection sort: Find minimum and make next in list
- Insertion sort: Take next and insert in correct place
- Both require operations O(N²)
- Tip: Always write check code (easier than work code)

Reading
- Sorting descriptions in packet

Announcements
- Homework 8 due Monday
- Exam 2 will be returned Monday
- Project 3 available soon: Game of Chance