Introduction Objectives

Objectives

Quadratic Sorting

Dr. Mattox Beckman

Illinois Institute of Technology Department of Computer Science This lecture covers the $\mathcal{O}(n^2)$ sorting algorithms.

- Your objective: be able to describe and implement:
 - selection sort
 - insertion sort
 - bubble sort

Dr. Mattox Beckman (IIT)	Quadratic Sorting	1 / 11	Dr. Mattox Beckman (IIT)	Quadratic Sorting	2 / 11
	Introduction Objectives			$\mathcal{O}(n^2)$ Sorts	
Complexity Measu	res		Selection Sort		

Complexity Measures

Dr. Mattox Beckman (IIT)

Input		Algorithm Speed							
Size	lg n	n	$n \lg n$	n^2	2^n	n!			
1	1s	1s	1s	1s	1s	<u>1s</u>			
2	1.1s	2s	2.2s	4s	2s	2s			
4	2s	4s	8s	16s	16s	24s			
8	3s	8s	24s	64s	256s	40320s (> 10h)			
16	4s	16s	64s	256s	65536s (>18h)	20,922,789,888,000s			
						(>663,457 years)			

Quadratic Sorting

Suppose you start with a pile of 1000 papers to sort by name, and an empty

- Go thru all 1000 papers, pick (i.e., *select*) the smallest.
- 2 Put it in the "already sorted" box.

box marked "already sorted".

- **⑤** Go through the remaining 999 papers, pick out the smallest.
- Put it underneath the other paper in the "already sorted" box.
- Repeat...

How long will this take? This is $\mathcal{O}(n^2)$. $1,000 + 999 + 998 + \cdots + 1 = 500,500$ steps.

Dr. Mattox Beckman (IIT) Quadratic Sorting $\mathcal{O}(n^2)$ Sorts

Selection Sorting Example

If we are using an array, we can sort "in-place" by exchanging the smallest element with the top-most unsorted element.

- Given an array, select the smallest element.
- We select the 10.
- Swap upward..

Dr. Mattox Beckman (IIT)

• Pass 1 done.

↓ロト→□ト→ミト→ミ りへで

Quadratic Sorting

Selection Sorting Example

If we are using an array, we can sort "in-place" by exchanging the smallest element with the top-most unsorted element.

- Given an array, select the smallest element.
- We select the 10.
- Swap upward..

Dr. Mattox Beckman (IIT)

• Pass 1 done.



Selection Sorting Example

If we are using an array, we can sort "in-place" by exchanging the smallest element with the top-most unsorted element.

- Given an array, select the smallest element.
- We select the 10.
- Swap upward..
- Pass 1 done.

Selection Sorting Example

If we are using an array, we can sort "in-place" by exchanging the smallest element with the top-most unsorted element.

Quadratic Sorting

40	40	40	40
23	23	23	10
67	67	10	23
10	10	67	67
36	36	36	36
52	52	52	52

Quadratic Sorting

- Given an array, select the smallest element.
- We select the 10.
- Swap upward..
- Pass 1 done.

◆□ > ◆□ > ◆豆 > ◆豆 > ・豆 * りへで

 $\mathcal{O}(n^2)$ Sorts Selection Sorting Example Selection Sorting Example If we are using an array, we can sort "in-place" by exchanging the smallest If we are using an array, we can sort "in-place" by exchanging the smallest element with the top-most unsorted element. element with the top-most unsorted element. • Given an array, select the smallest element. • Given an array, select the smallest element. • We select the 10. • We select the 10. • Swap upward.. • Swap upward.. • Pass 1 done. • Pass 1 done. **◆□▶◆□▶◆■▶ ● 夕**♀ 4□ → 4□ → 4 = → 4 = → 9 < 0</p> Dr. Mattox Beckman (IIT) Quadratic Sorting Dr. Mattox Beckman (IIT) Quadratic Sorting $\mathcal{O}(n^2)$ Sorts $\mathcal{O}(n^2)$ Sorts More steps More steps

Notice the pattern made by the black, blue, and green elements!

Notice the pattern made by the black, blue, and green elements!

(ロトイラトイミトイミト ミックスで Dr. Mattox Beckman (IIT) Quadratic Sorting 6/11 Dr. Mattox Beckman (IIT) Quadratic Sorting 6/11 Or. Mattox Beckman (IIT) Quadratic Sorting 6/11 $\mathcal{O}(n^2)$ Sorts $\mathcal{O}(n^2)$ Sorts More steps



Notice the pattern made by the black, blue, and green elements!

Notice the pattern made by the black, blue, and green elements!





Notice the pattern made by the black, blue, and green elements!

Notice the pattern made by the black, blue, and green elements!

 $\mathcal{O}(n^2)$ Sorts $\mathcal{O}(n^2)$ Sorts

Another Example

- 4 2 6 1 3 5 7
- Black = untouched
- Blue = checked for minimality
- Green = already sorted
- Red = selected

4	4
2	2
6	6
1	1
3	3
5	5
7	7

• Black = untouched

Another Example

- Blue = checked for minimality
- Green = already sorted
- Red = selected

Dr. Mattox Beckman (IIT)	Quadratic Sorting	7 / 11	Dr. Mattox Beckman (IIT)	Quadratic Sorting	7 / 11
	$\mathcal{O}(n^2)$ Sorts			$\mathcal{O}(n^2)$ Sorts	

Another Example

Quadratic Sorting

- Black = untouched
- Blue = checked for minimality
- Green = already sorted
- Red = selected

Another Example

- 4
 4
 1
 1

 2
 6
 6
 2
 4

 1
 1
 6
 6

 3
 3
 3
 3

 5
 5
 5
 5

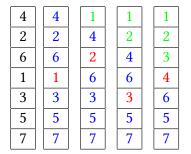
 7
 7
 7
 7
- Black = untouched
- Blue = checked for minimality
- Green = already sorted
- Red = selected

Dr. Mattox Beckman (IIT)

 $\mathcal{O}(n^2)$ Sorts

Another Example

Another Example



- Black = untouched
- Blue = checked for minimality
- Green = already sorted
- Red = selected

4	4	1	1	1	1
2	2	4	2	2	2
6	6	2	4	3	3
1	1	6	6	4	4
3	3	3	3	6	6
5	5	5	5	5	5
7	7	7	7	7	7

- Black = untouched
- Blue = checked for minimality
- Green = already sorted
- Red = selected

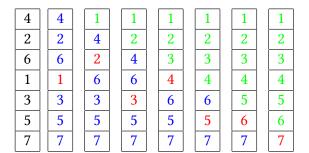
					₹
Dr. Mattox Beckman (IIT)	Quadratic Sorting	7 / 11	Dr. Mattox Beckman (IIT)	Quadratic Sorting	7 / 11
	$\mathcal{O}(n^2)$ Sorts			$\mathcal{O}(n^2)$ Sorts	

Another Example

3 3 6 3 5 5 5 5 5 6 5

- Black = untouched
- Blue = checked for minimality
- Green = already sorted
- Red = selected

Another Example

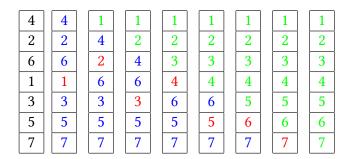


- Black = untouched
- Blue = checked for minimality
- Green = already sorted
- Red = selected

 $\mathcal{O}(n^2)$ Sorts $\mathcal{O}(n^2)$ Sorts

Another Example

Other $\mathcal{O}(n^2)$ sorts



- Black = untouched
- Blue = checked for minimality
- Green = already sorted
- Red = selected

- Insertion Sort is similar to Selection Sort. The difference is when you do the sorting.
 - Pick the first exam from the top of the pile.
 - ② Go thru the exams in the "already sorted" pile, and *insert* it in the proper location.
 - 3 Repeat...
 - This is used for linked lists.
- Bubble Sort uses only one box...

Example 2 with insertion sort

- Go through the exams in the box. When you are on exam n, check exam n + 1. If they are out of order, swap them.
- 2 Repeat...
- Never use this.



Insertion Sort Example

Arrays would look like this; linked lists are a bit more efficient.

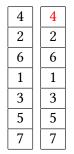
Out Input Red = "selected" Blue = shifted Green = not checked

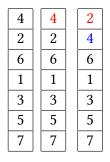
How quickly would this run if the input list were already sorted?

 $\mathcal{O}(n^2)$ Sorts $\mathcal{O}(n^2)$ Sorts

Example 2 with insertion sort

Example 2 with insertion sort

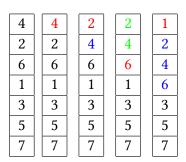






Example 2 with insertion sort

1 3 3 3 3 5 5 5 5



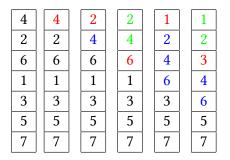
10 / 11

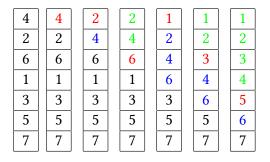
Dr. Mattox Beckman (IIT)

 $\mathcal{O}(n^2)$ Sorts

Example 2 with insertion sort

Example 2 with insertion sort





Dr. Mattox Beckman (IIT)

Quadratic Sorting

O(n²) Sorts

O(n²) Sorts

O(n²) Sorts

O(n²) Sorts

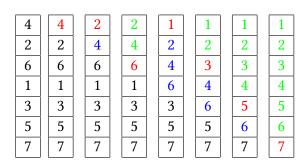
O(n²) Sorts

O(n²) Sorts

Example 2 with insertion sort

Bubble Sort Example, Pass 1

Pass 1: the 67 will "sink" to the bottom.



```
40
       23
              23
                     23
                           23
                                   23
23
       40
                     40
                           40
67
       67
              67
                     10
                           10
                                   10
10
       10
                     67
                           36
                                   36
              10
36
       36
              36
                     36
                           67
                                   52
52
       52
              52
                     52
                           52
                                   67
```

Quadratic Sorting

11 / 11

Bubble Sort Example, Pass 2 and 3

On pass 3 we detect that the array is sorted.

23	23	23	23	23	10	10	10	10
40	40	10	10	10	23	23	23	23
10	10	40	36	36	36	36	36	36
36	36	36	40	40	40	40	40	40
52	52	52	52	52	52	52	52	52
67	67	67	67	67	67	67	67	67

This sort is inefficient because exchanges don't move very far.

