CS21: INTRODUCTION TO
COMPUTER SCIENCE

Prof. Mathieson
Fall 2017
Swarthmore College
Outline Nov 13:

• Selection Sort
• Bubble Sort
• Insertion Sort
• Runtime of these sorting algorithms

Notes

• Lab 8 due Saturday night (read BEFORE coming to lab!)
• Lab 9 due Monday after Thanksgiving
• Quiz 4 this Friday (let me know if you have conflicts)
Sorting
Types of sorting

- **Out-of-place**: leaves the original list alone and creates a new sorted list (returns new list)

- **In-place**: modifies (mutates) the original list via swaps so that it is now sorted

- **Pros of in-place sort**: no new data structure needed (saving space)

- **Cons of in-place sort**: original order destroyed (in some cases it might be important), can be more difficult to implement
3 sorts for today

- **Selection Sort**: iteratively find the minimum element and place it in the correct position

- **Bubble Sort**: move through the list, swapping adjacent elements if they are out of place (repeat until sorted)

- **Insertion Sort**: for each element of the list, move it down until it is in the correct position
Selection Sort Example + Runtime

Index: 0 1 2 3 4 5 6 7 8 9 10
Data: 18 1 21 10 2 8 3 6 20 25 7

\[
\left( \frac{n+1}{2} \right) \cdot \frac{n}{2} \rightarrow O(n^2) \\
(n+1) + (n-1) + (n-2) + \ldots + 3 + 2 + 1 \rightarrow O(n^2)
\]

Time:
- \( i \)
- \( m = \frac{n}{2} \)

\[
\begin{array}{c|c|c}
0 & 1 & \text{time} \\
1 & 4 & n-1 \\
2 & 6 & n-2 \\
\vdots & \vdots & \vdots \\
10 & 10 & 1 \\
\end{array}
\]

Bubble Sort:
- \( n \)
- \( n-1 \)
- \( n-2 \)

\[
\sum_{k=1}^{n} k = \frac{n(n+1)}{2} 
\]