I & C SCI 46 Spring 2022
Lecture 22: Fundamentals of Sorting

2 What is sorting?

- Input: sequence of n comparable values
- Reorder the input to be non-descending.
- Items we wish to sort are called "keys"

- Not here: retain associated information

3 Why discuss sorting?

`std::sort(v.begin(), v.end())`

- Standard library has sorting
- Why not use that and move on?

In this class, sorting is:

- a good intro for techniques
- a good intro to comparative algorithms

4 SelectionSort

Idea: Swap min into first spot, second-min to second, etc.
(This is hand-wavy on purpose)

85	24	63	45	17	31	96	50
17	24	63	45	85	31	96	50
17	24	63	45	85	31	96	50
17	24	31	45	85	63	96	50

5 SelectionSort

```
for i ← 1 to n − 1 do
    min ← i
    for j ← i + 1 to n do
        if A[j] < A[min] then
            min ← j
    Swap A[i] and A[min]
```

$O(1)$

$$\sum_{i=1}^{n-1} \sum_{j=i+1}^{n} 1$$

6 What's nice about SelectionSort?

- Easy to program
- Easy to explain
- Does it waste memory? no: $O(1)$
- Does it only work for numbers?
  Total ordering
- What other info do we need?
- Are there inputs that are sorted faster?
- Is there a lot of data movement?

7 BubbleSort

Idea: Think globally act locally

85	24	63	45	17	31	96	50
24	63	45	17	31	85	50	96
24	45	17 (63)	31 (63)	63 (63)	50	85	96

8 BubbleSort

for $i \leftarrow 1$ to $n-1$ **do**
  for $j \leftarrow 1$ to $n-i$ **do**
    if $A[j+1] < A[j]$ **then**
      Swap $A[j]$ and $A[j+1]$