Last Time

• Some algorithms are fast, others are slow
• We can use timing to compare algorithms
• Two methods of presentation: charts and graphs, each with their own pros and cons
• The downside of timing
We said at the end of last class that we’ll measure the speed of an algorithm by counting its primitive operations:

- Assigning a value to a variable.
- Following an object reference.
- Performing an arithmetic operation.
- Comparing two numbers.
- Accessing an array index.
- Calling a method / returning from a method.

Which is more important – the difference between 10 and 20 operations, or between 10 and 1000 operations?
Orders of Growth

• Constant Function: \( f(n) = c \)
• Logarithmic Function: \( f(n) = \log(n) \)
• Linear Function: \( f(n) = n \)
• “Linearithmic” Function: \( f(n) = n \times \log(n) \)
• Quadratic Function: \( f(n) = n^2 \)
• Cubic Function: \( f(n) = n^3 \)
• Exponential Function: \( f(n) = 2^n \)
Any Questions?