CSC 212
Data Structures and Abstractions
Spring 2016

Lecture 06: Pointers and Dynamic Arrays
Announcements

PA #1 is out

submissions **must** be through Autolab

start early and ask questions

office hours and Piazza
Previously …

Recurrences
used to determine running time of recursive algorithms
Today …

Pointers and Arrays

Example with Dynamic Arrays
int a[] = {3, 4, 7, 1, 6, 9};
int a[] = {3, 4, 7, 1, 6, 9};
int a[] = {3, 4, 7, 1, 6, 9};
int a[] = {3, 4, 7, 1, 6, 9};
int a[] = {3, 4, 7, 1, 6, 9};

How to append more data?
Dynamic Arrays

Not dynamically allocated arrays
Dynamic Arrays

Not dynamically allocated arrays

Dynamic arrays can grow/shrink over time need to remember size and capacity
a

```
 3  4  7  1  6  9

0 1 2 3 4 5
```
Capacity = 7
Capacity = 7
Size = 6
append

Insert at end

Let's analyze the number of 'writes'
Grow by 1

If array is full, increase the capacity by 1

Cost of adding first N elements?

ignore cost of memory allocation

\[ N + [1 + 2 + 3 + \cdots + (N - 1)] = \frac{N(N + 1)}{2} = O(N^2) \]
Repeated Doubling

If full, create a new array of \textbf{twice} the size

Cost of adding first $N=2^k$ elements?

\[
N + [2^0 + 2^1 + 2^2 + \cdots + 2^{k-1}] = N + \sum_{i=0}^{k-1} 2^i \\
= N + 2^k - 1 \\
= 2N - 1 = O(N)
\]
How to Shrink arrays?

halve the array when array is one-half full?

or

halve the array when array is one-quarter full?