CS180 - Vectors

Announcements

- Today: checkpoint
- Program is due Tuesday
- Lab due Sunday
  (no lab next week)
- Review on Wed
  Test Thursday
Recap of runtimes:

Stack or Queue
push, pop, top, size, empty: \( O(1) \)
(\( \rightarrow \text{might be } O(n) \) in array-based
hidden cost:

Copy constructor \& operator:\( = O(n) \)

Destructor: array-based \( O(1) \)
linked-based \( O(n) \)

smart Stack fixes this
Vectors - Ch 6.1

Similar to lists in Python

Modification: single type

myvector [5] = 6;

Extendable:

Diagram: [Insert diagram here]
Code:

```c
private data:
int capacity;
int size;
Object* data;
```

Functions:
(see STL)
Insert

myvec. insert (2, 'c');
othervec. insert (11, "new");
anothervec. insert (7, -25);

How to insert?
(What if full?)

extendable:
- allocate bigger array
- copy data down
Erase

- take an index & delete that element

Ex: myVec.erase(3)

data => 0 1 2 3 4 5 6 7
Another: Operator `[]`

What else: 

Housekeeping!

In Python, shallow copies were default