Recursion

CS 101 - Spring 2016
Creating a recursive solution

• **Base case**  
  • A trivial and easily solvable instance of the problem

• **Recursive case**  
  • Break the problem up into solvable problems and smaller versions of the same problem *[must make progress toward the base case]*  
  • Make the problem smaller by looking at smaller numbers, less data, or fewer choices  
  • Figure out how to combine the solutions to smaller problems to get the solution to the overall problem
```python
def spaceCount(text):
    spaceLocation = text.find(' ')
    if spaceLocation == -1:
        return 0
    else:
        return 1 + spaceCount(text[spaceLocation+1:])
```
def spaceCount(text):
    spaceLocation = text.find(' ')
    if spaceLocation == -1:
        return 0
    else:
        return 1 + spaceCount(text[spaceLocation+1:])

find first space

is there a space?

return 0

create slice of text that doesn't include first space

count spaces in new string

return 1 + number of spaces after first one
def spaceCount(text):
    spaceLocation = text.find(' ')
    if spaceLocation == -1:
        return 0
    else:
        return 1 + spaceCount(text[spaceLocation+1:])

find first space

is there a space?

no

return 0

yes

create slice of text that doesn't include first space

count spaces in new string

return 1 + number of spaces after first one