



### Announcements

- This week reading : Chapter 3.4, 3.5
- Next week reading: Dictionaries 7.1, 7.2, 7.3
- Project 2 will be out tonight. Due 14
- Midterm 2: Nov 8

### Review: the virtues of negative thinking!

```

    0  1  2      3      4
   -5 -4 -3      -2      -1

myList = [42, 47, 23, [3.141, 2.718], 5]

>>> myList[len(myList)-3]
23

>>> myList[-3]
23

>>> myList[-2:]
[[3.141, 2.718], 5]
  
```

### Review: the virtues of negative thinking!

```

    0123456789
myString = "I luv spam"

>>> myString[:-1]
'I luv spa'

>>> myString[-3:]
'pam'
  
```

## Review: two types of for loop

```
def spamify(L):
    '''Add "n spam" to every string in L.'''
    newL = []
    for s in L:
        newL.append(s+"n spam")
    return newL
```

← For loop goes directly over the list L

```
>>> spamify(["eggs","sausage","oatmeal"])
['eggsn spam', 'sausagen spam', 'oatmealn spam']
```



Green eggs n spam!

## Review: two types of for loop

```
def spamCount(S):
    '''Count occurrences of "spam" in input S.'''
    counter = 0
    for i in range(len(S)):
        if S[i:i+4] == "spam":
            counter = counter + 1
    return counter
```

← For loop goes over indices into string S

```
>>> spamCount("gspamtspamspam")
3
```

## An alternate use of `in`

```
>>> for num in range(1, 100):
...
>>> 42 in [3, 67, 42, 18, 2001]
True
>>> 42 in [13, 33, 300]
False
>>> food = ["carrots","coffee","arugula","spam"]
>>> if "spam" in food: print("Yay!!!")
...
Yay!!!
>>> "bio" in "symbiont"
True
```

## while loops



```
def mystery(n):
    k = 1
    while k < n:
        k = k * 2
    return k
>>> mystery(1)
>>> mystery(5)
>>> mystery(10)
```