



Lecture 11:
Iteration and For-Loops
(Sections 4.2 and 10.3)
CS 1110
Introduction to Computing Using Python



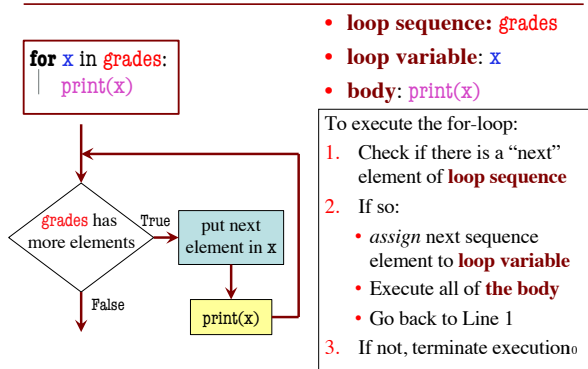
[E. Andersen, A. Bracy, D. Gries, L. Lee, S. Marschner, C. Van Loan, W. White]

Problem: Summing the Elements of a List

```
def sum(the_list):
    """Returns: the sum of all elements in the_list
    Precondition: the_list is a list of all numbers
    (either floats or ints)"""
```

6

For Loops: Processing Sequences



What gets printed? (Q1)

<pre>a = 0 for b in [1]: a = a + 1 print(a)</pre>	<pre>a = 0 for b in [1, 2]: a = a + 1 print(a)</pre>	<pre>a = 0 for b in [1, 2, 3]: a = a + 1 print(a)</pre>	<pre>a = 0 for b in [1, 2, 3]: a = b print(a)</pre>
□	□	□	□

12

What gets printed? (Q2)

<pre>a = 0 for b in [1, 2, 3]: a = a + b print(a)</pre>	<pre>a = 0 b = [1, 2, 3] for c in b: a = a + c print(a)</pre>	<pre>a = 0 b = [1, 2, 3] for c in b: a = a + c print(b)</pre>
□	□	□

14

For Loop with labels

```
def num_ints(the_list):
    """Returns: the number of ints in the_list
    Precondition: the_list is a list of any mix of types"""
    result = 0
    for x in the_list:
        if type(x) == int:
            result = result + 1
    return result
```

Accumulator variable
Loop sequence
Loop variable
Body

17

range: a handy counting function!

`range(x)`
returns $0, 1, \dots, x-1$

```
>>> first_six = list(range(6))
>>> print(first_six)
[0, 1, 2, 3, 4, 5]
```

`range(a,b)`
returns $a, \dots, b-1$

```
>>> second_six = list(range(6,13))
>>> print(second_six)
[6, 7, 8, 9, 10, 11, 12]
```

Important: range does not return a list

→ need to convert `range`'s return value into a list 19

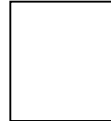
What gets printed? (Q3)

```
a = 0
for b in range(0, 1):
    a = a + 1

print(a)
```

```
a = 0
for b in range(0, 4):
    a = a + 1

print(a)
```



23

The Map Function

`map(function, list)`

```
map(f, [a,b,c,d])
```

↓

```
f(a), f(b), f(c), f(d)
```

- `function` takes 1 parameter
- Otherwise, **error**

Important: map does not return a list

→ need to convert `map`'s return value into a list

```
>>> len_list = list(map(len, ['a', 'bc', 'defg']))
>>> len_list
[1, 2, 4]
```

26

The Filter Function

`filter(Boolean_function, list)`

```
filter(f, [a,b,c])
```

↓

```
a if f(a)==True,
b if f(b)==True,
c if f(c)==True,
```

- `function` takes 1 parameter
- `function` returns a Boolean
- Collects elements of `list` for which `Boolean_function` returns True

Important: filter does not return a list

→ need to convert `map`'s return value into a list

See `ints.py` to see filter in action

27

For-Loop Mistake #1 (Q)

Modifying the loop sequence as you walk through it.

```
b = [1, 2, 3]
for a in b:
    b.append(a)
```

```
A: never prints b
B: [1, 2, 3, 1, 2, 3]
C: [1, 2, 3]
D: I do not know
```

```
print b
```

29

For-Loop Mistake #2 (Q)

Modifying the loop variable (here: x).

```
def add_one(the_list):
    """Adds 1 to every element in the list
    Precondition: the_list is a list of all numbers
    (either floats or ints)"""
    for x in the_list:
        x = x+1
```

```
a = [5, 4, 7]
add_one(a)
print(a)
```

What gets printed?

```
A: [5, 4, 7]
B: [5, 4, 7, 5, 4, 7]
C: [6, 5, 8]
D: Error
E: I don't know
```

31