CompSci 6
Introduction to Computer Science

September 27, 2011

Prof. Rodger

Announcements

- No Reading for next time
- No Reading Quiz for next time
- Apt-02 due tonight, Assignment 3 due Thursday

- Exam next Tuesday
- Work the practice exam before Thursday's class. Will go over next class.

List Comprehension review

- Take advantage of patterns, make a new list based on per element calculations of another list

- Format:

[<expression with variable> for <variable> in <old list>]

- Format with filtering:

[<expression with variable> for <variable> in <old list> if <filter with variable>]

List Comprehensions from last time

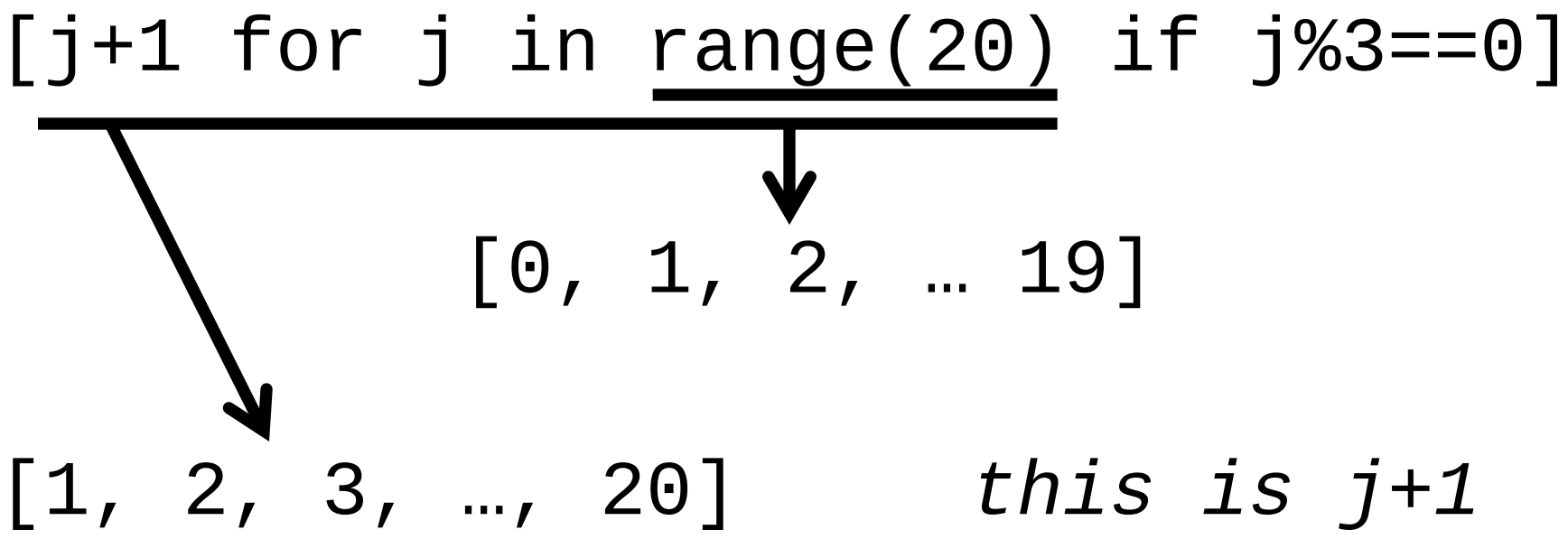
```
[j+1 for j in range(20) if j%3==0]
```

List Comprehensions from last time

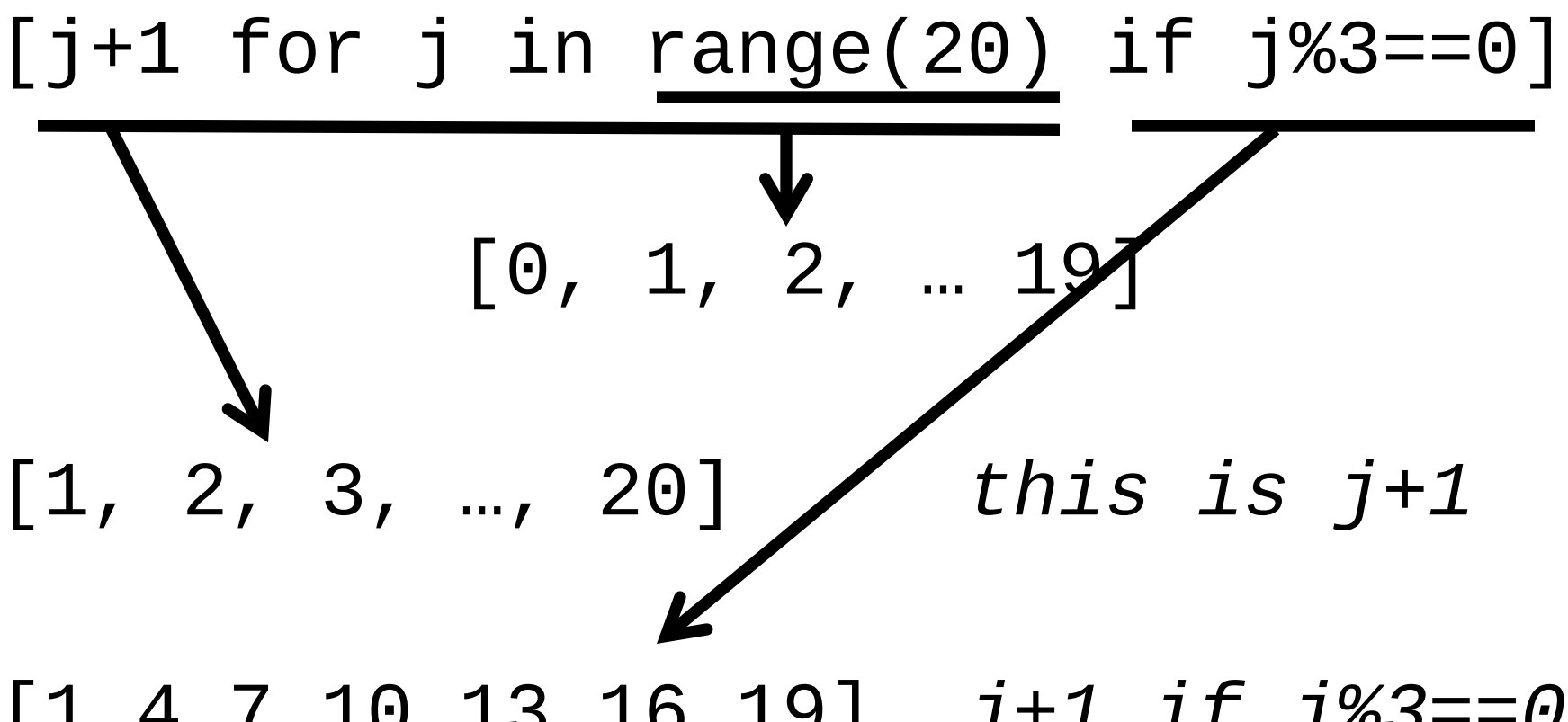
```
[j+1 for j in range(20) if j%3==0]
```

```
[0, 1, 2, … 19]
```

List Comprehensions from last time



List Comprehensions from last time



List Comprehensions from last time

```
[i*2 for i in [j+1 for j in
range(20) if j%3==0] if i*i > 19]
```

List Comprehensions from last time

```
[i*2 for i in [j+1 for j in range(20) if j%3==0] if i*i > 19]
```

```
[i*2 for i in [1,4,7,10,…,19] if i*i > 19]
```

List Comprehensions from last time

```
[i*2 for i in [j+1 for j in
range(20) if j%3==0] if i*i > 19]
```

```
[i*2 for i in [1,4,7,10,…,19] if
i*i > 19]
```

```
[[2,8,14,20,26,32,38]  if i*i > 19]
```

```
[14, 20, 26, 32, 38]  <- answer
after filtering
```

List Comprehension examples

- List comprehension creates a new list. Use wherever you need a new list.
- Can use list comprehension to create a list to return
- From Uppity.py – words is list of words

```
def uppify_list(words):
    return [w.upper() for w in words]
```

- Classwork problem 1
- Show additional examples

Problem: complete program

- Problem:
- Given a file of words, for each line print out only those words that are longer than 4, thus removing all the “short” words.

Example:

Where are all the wild things?

becomes

Where things?

Tasks in solving this problem

- First understand the code given
- Then fill in missing code (TODO)
  - In Eclipse, “Window”, “Show view”, “Tasks”
- Test it with data file you create
  - May need to create data folder
- Where is the fence post problem?
- Debugging – what do you do when it doesn’t work?

Fence post problem

- How many posts, how many supports between all those posts?

- Build fence:

```
for post in posts:
        fence = fence + post + supports
```

Passing Functions as Parameters

```
def upperWord(word):
  return word.upper()
def argWord(word):
  return word + “arg”
def transformWord(func, word):
  return func(word)

print transformWord(upperWord, “train”)
print transformWord(argWord, “truck”)
```

Assignment 3 Transform

- Look over assignment
- What parts are similar to what we just did?
- Passing functions as parameters
- What are the imports?
- Has file browser for you

```
import InputGUI as Input
```

- To input from Console change line to:

```
import InputConsole as Input
```