Announcements

**Grades for Lab 1** should all be posted in CMS. Please verify that you have a 1 if you checked off the lab. Let course staff know if your grade is missing!

**Assignment 1** is out. The first submission is due **February 24**. CMS will show an extension, but this is only for your corrected revisions later on.

**No labs** next week due to Winter Break.

**Many office hours** are available. See the course website.
How Do Functions Work?

To evaluate a function call expression:
1. Create a frame for the call
2. Assign arguments to parameters
3. Execute function body
4. Erase the frame

The value of the function call expression is the returned value

---

```python
def add_em(x, y):
    return x + y

print(add_em(3, 4))
```

Return value: 7
How Do Functions Work?

To evaluate a function call expression:
1. Create a frame for the call
2. Assign arguments to parameters
3. Execute function body
4. Erase the frame

The value of the function call expression is the returned value

```
def add_em(x, y):
    sum = x + y
    return sum
```

```
print add_em(3, 4)
```

Return value: 7
To evaluate a function call expression:
1. Create a frame for the call
2. Assign arguments to parameters
3. Execute function body
4. Erase the frame

The value of the function call expression is the returned value

 Execute this call on paper. What gets printed out?

A: x, y: 1 2
B: x, y: 2 1
C: x, y: 2 2
D: x, y: 1 1
E: I don’t know
To evaluate a function call expression:
1. Create a frame for the call
2. Assign arguments to parameters
3. Execute function body
4. Erase the frame
The value of the function call expression is the returned value
def swap(x, y):
    t = x
    x = y
    y = t
    print 'x, y:', x, y

a = 1
b = 2
swap(a, b)
print 'a, b:', a, b

To evaluate a function call expression:
1. Create a frame for the call
2. Assign arguments to parameters
3. Execute function body
4. Erase the frame

The value of the function call expression is the returned value

Execute this call on paper. What gets printed out?

A: x, y: 1 2
B: x, y: 2 1
C: x, y: 2 2
D: x, y: 1 1
E: I don’t know
To evaluate a function call expression:
1. Create a frame for the call
2. Assign arguments to parameters
3. Execute function body
4. Erase the frame

The value of the function call expression is the returned value

Execute this call on paper. What gets printed out?

A: a, b: 1 2
B: a, b: 2 1
C: a, b: 2 2
D: a, b: 1 1
E: I don’t know
To evaluate a function call expression:
1. Create a frame for the call
2. Assign arguments to parameters
3. Execute function body
4. Erase the frame
The value of the function call expression is the returned value
```python
import point
p = point.Point(1, 2, 3)
q = point.Point(3, 4, 5)
swap_x(p, q)
print 'p:', p
print 'q:', q
```

```
def swap_x(p, q):
    t = p.x
    p.x = q.x
    q.x = t
```

Execute this code on paper. You will draw 2 objects and a frame. What is in \( p.x \) at the end?

- A: 1
- B: 2
- C: 3
- D: I don’t know
```python
def swap_x(p, q):
    t = p.x
    p.x = q.x
    q.x = t

import point
p = point.Point(1,2,3)
q = point.Point(3,4,5)
swap_x(p, q)
print 'p:', p
print 'q:', q
```

def f(x, y):
    return x * (y**2)

g = 3e8
print g(3)

1. Create a frame for the call
2. Assign arguments to parameters
3. Execute function body
4. Erase the frame
How to Draw Things

- **object**
- **variable**
- **class**
- **identifier**
- **class name**
- **attributes**
- **function name**: **program counter**
- **parameters**
- **local variables**
- **old value**
- **value**
- **class name**
- **methods**
- **module name**
- **call frame**
Online Python Tutor

pythontutor.com

type in whatever code you want

controls for stepping through code

module (global) variables

frame for call of g

frame for call of f

output from print goes here