MORE FUNCTIONS AND PARAMETERS

CSSE 120—Rose Hulman Institute of Technology

Function Review

- Functions can take multiple parameters
  - ```
    def distance (p1, p2): # p1, p2 are points
        xdist = abs(p1.getX()- p2.getX())
        ydist = abs(p1.getY()- p2.getY())
        return math.sqrt(xdist*xdist + ydist*ydist)
    ```
- Invoke a function; must supply actual parameters:
  - `d = distance(Point(-1,2), Point(2,6))`
- Functions can return values (see distance)
- More about parameters (details on later slides):
  - What happens when we modify them?
  - What is an optional parameter?
- More about return values:
  - Can return multiple values (details on later slides)
    ```

Passing parameters in Python

- What type of information do formal parameters receive?
- If we assign new values to formal parameters, does this affect the actual parameters?
- Consider this version of square:

```
def squareNext(x):
  x = x + 1
  return  x * x
```

Passing a mutable parameter

- Function can change contents of a mutable parameter:

```
def addOneToAll(listOfNums):
    for i in range(len(listOfNums)):
        listOfNums[ i ] +=1

myList = [1, 3, 5, 7]
addOneToAll(myList)
print myList
```

- What does this print?

  What actually gets passed to the function?

Optional parameters

- A python function may have some parameters that are optional.

Also look at calls to GraphWin

```
>>> int("37")
37
>>> int("37", 10)
37
>>> int("37", 8) # specify base 8
31
```

We can declare a parameter to be optional by supplying a default value.

```
>>> def printDate(month, day, year=2007):
        print month, str(day)+",", year


>>> printDate("January", 4, 2006)
January 4, 2006
>>> printDate("January", 4)
January 4, 2007
```

Multiple optional parameters

- If there are more than one, and it's not clear which argument you are providing, you can pass **variable=value:**

Note that all 3 are optional:

```
>>> def printDate(month = 'January', day = 1, year=2007):
        print month, str(day)+',', year
```

```
>>> printDate()
January 1, 2007
>>> printDate(26)
26 1, 2007
>>> printDate(day=26)
January 26, 2007
```

Nice!

I wanted the 26th. Whoops!

That's it.

Returning Multiple Values

- A function can return multiple values
  - **def powers(n):**
    return n2, n**3, n**4**
- What's the type of the value returned by this call?
  powers(4)
- Assign returned values individually, or to a tuple:

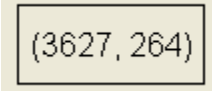
```
listOfPowers = powers(5)
p2, p3, p4 = powers(5)
```

Homework

- Because of break, you can do the homework with your partner from today's class, or do it alone.
- Some parts are not easy; we suggest that you start it today so you can get help during assistant lab hours this afternoon or evening if you get stuck.
- After you finish `threeSquares`, work on `triangles` until the end of class.
- If you also finish `triangles`, work on the other parts of the homework.

Pair Programming: Three Squares

1. Checkout **Session09** project from your SVN repository
2. Work with another student on one computer
3. Run the threeSquares program to be sure it works. Put **both** students' names in the initial comment.
4. Add a function, **stats**, that takes a Rectangle, **r**, as a parameter and returns the area of **r**
5. modify the program so that it displays the area of each rectangle inside the rectangle
6. Finally, change **stats** to return the area and perimeter (see figure at right)
7. Commit your project back to your repository; also email **threeSquares.py** to your partner.



Example Display