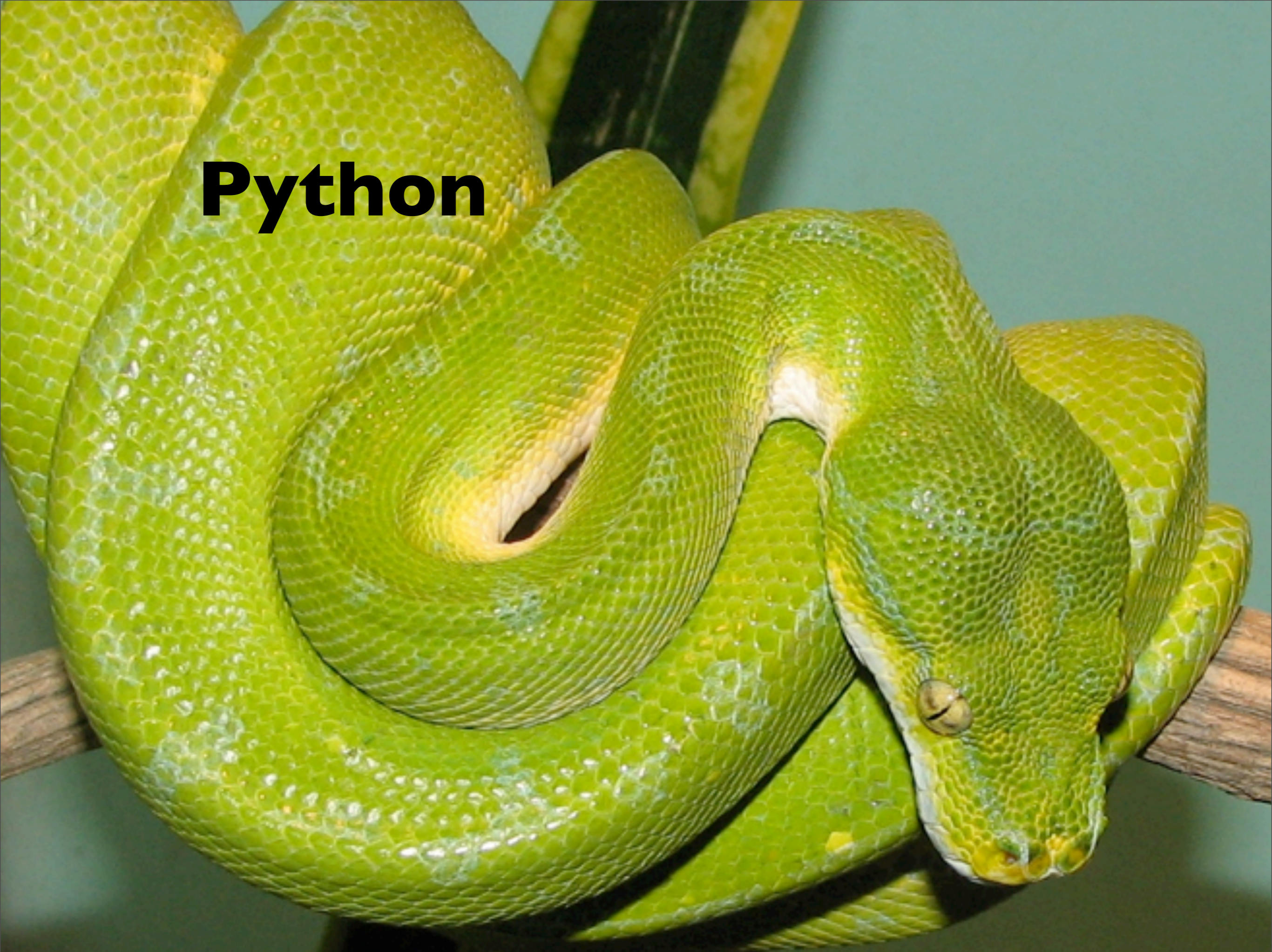


Python



What is Python?

Python is an interpreted, interactive, object-oriented programming language. It incorporates modules, exceptions, dynamic typing, very high level dynamic data types, and classes. Python combines remarkable power with very clear syntax. It has interfaces to many system calls and libraries, as well as to various window systems, and is extensible in C or C++. It is also usable as an extension language for applications that need a programmable interface. Finally, Python is portable: it runs on many Unix variants, on the Mac, and on PCs under MS-DOS, Windows, Windows NT, and OS/2.

To find out more, start with *The Python Tutorial*. The [Beginner's Guide to Python](#) links to other introductory tutorials and resources for learning Python.

<http://docs.python.org/2/faq/general.html#id1>

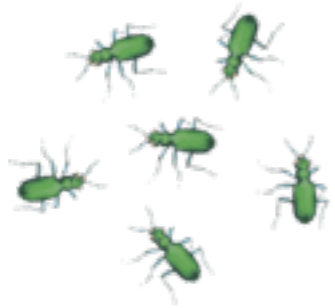
Why is it called Python?

When he began implementing Python, Guido van Rossum was also reading the published scripts from “[Monty Python’s Flying Circus](#)”, a BBC comedy series from the 1970s. Van Rossum thought he needed a name that was short, unique, and slightly mysterious, so he decided to call the language Python.

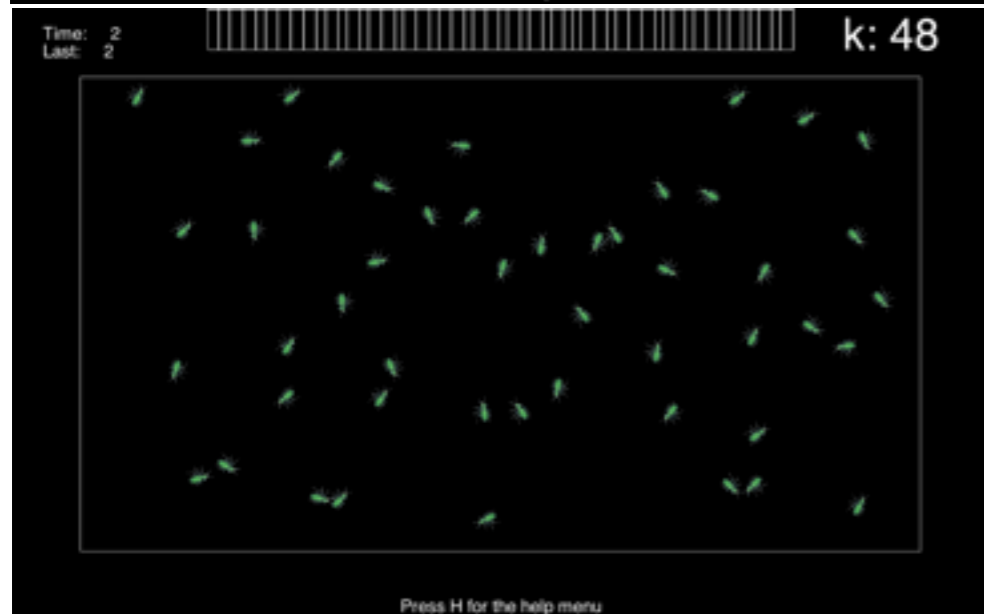
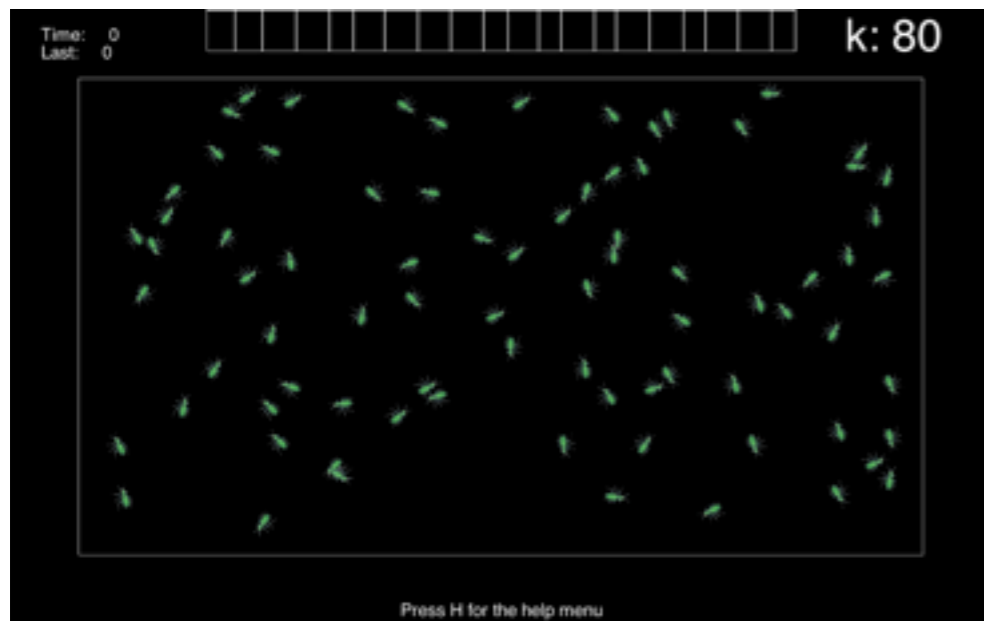
“The most important thing in the programming language is the name. A language will not succeed without a good name. I have recently invented a very good name and now I am looking for a suitable language.” — Donald Knuth

<http://docs.python.org/2/faq/general.html#id1>

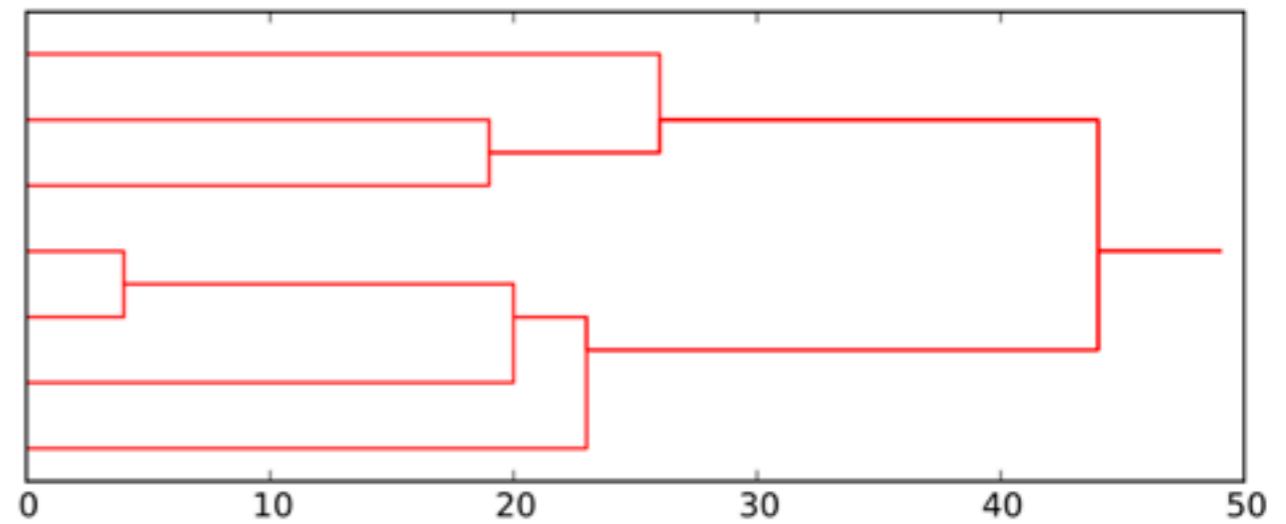
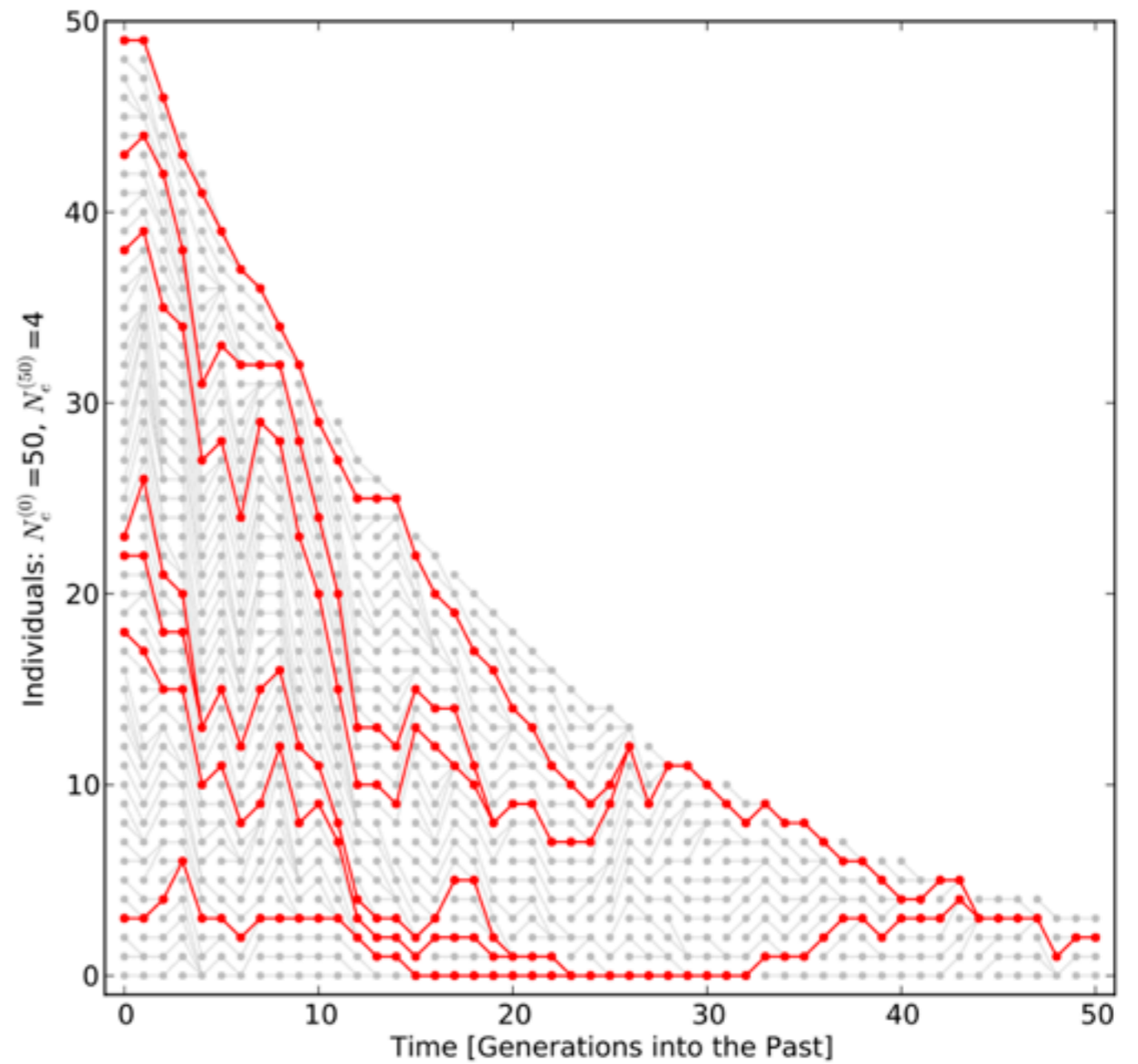
Examples



bugs in a box



popvizard



Python baby steps: we learn how to calculate Pi

Open two terminal windows that point to the same directory.
Use the text editor

`nano`

to edit a file in one window and in the other execute that file with something like this

`python file`

for python programs I often use the `.py` extension, for our examples use `hello.py` and `pi.py` as file names.
Again make sure that both terminal window point to the same directory (use `pwd` to check).

Python programming steps

my first program

Enter in file:

```
print "Hello world"
```

Result:

```
Hello world
```

Python programming steps

printing

Enter in file:

```
a = 1
b = 2
print a, b
print "---"
print a
print b
```

Result:

```
1 2
---
1
1
```

Python programming steps

Looping

Enter in file:

```
a = 0          # the # is a comment, a is assigned zero
b = 10        # b is assigned 10
while a < b:  # while a is smaller than b do the following
    a = a + 1 # add 1 to a and assign the result to a
    print a,  # print a, the "," means do not add an end-of-line
              # the indentation is important in python because
              # it marks that all the material belongs to the
              # while statement, a ":" marks such a statement.
```

Result:

```
1 2 3 4 5 6 7 8 9 10
```


Python programming steps

Looping

Enter in file:

```
a = range(10)    # creates a list from 0 to 9
b = range(1,11) # creates a list from 1 to 10
# loop over all b and print a running sum of the square of b[i]
sum = 0
for bi in b:
    sum = sum + bi * bi
    print sum,
print
```

Result:

```
1 5 14 30 55 91 140 204 285 385
```

Python programming steps

decisions

Enter in file:

```
a = 0
b = 10
c = 5
while a < b: # loop as long a is smaller than b
    a = a + 1 # increase a
    if a < c: # if a is smaller than c
        print a, # print a
    else: # otherwise
        print a*a, # print the square of a
        #
print "done" #
```

Result:

```
1 2 3 4 25 36 49 64 81 100 done
```

Python programming steps

list comprehension

Enter in file:

```
a = [] # a is initialized as an empty list
print "a=",a
b = [1,2,3,4,"5"] # b is a list with mixed types
print b
c = b # c is a clone of b
b[0] = 5 # changing the first element of b
b[-1] = 100 # changing the last element of b
print "b=",b # print b
print "c=",c # c is just another name for b
c = b[:] # c now is a independent copy of b
b[1] = 21
print "b",b
print "c=",c
```

Result:

```
a= []
b= [1, 2, 3, 4, '5']
b= [5, 2, 3, 4, 'five']
c= [5, 2, 3, 4, 'five']
b= [5, 21, 3, 4, 'five']
c= [5, 2, 3, 4, 'five']
```

Python programming steps

list comprehension

Enter in file:

```
a=[0,1,2,3,4,5,6,7,8,9]
b=a[0]
c=a[-1]
d=a[2:4]
e=a[3:-2]
print b
print c
print d
print e
```

Result:

```
0
9
[2, 3]
[3, 4, 5, 6, 7]
```

Python programming steps

strings are funny lists

Enter in file:

```
a="the quick fox jumps over the lazy dog"
b=a[-1]
e=a[3:-8]
f=list(a)
g = a.split()
h = a.upper()
print b
print e
print f
print g
print h
```

Result:

```
g
quick fox jumps over the
['t', 'h', 'e', ' ', 'q', 'u', 'i', 'c', 'k', ' ', 'f', 'o', 'x', ' ', 'j',
'u', 'm', 'p', 's', ' ', 'o', 'v', 'e', 'r', ' ', 't', 'h', 'e', ' ', 'l',
'a', 'z', 'y', ' ', 'd', 'o', 'g']
['the', 'quick', 'fox', 'jumps', 'over', 'the', 'lazy', 'dog']
THE QUICK FOX JUMPS OVER THE LAZY DOG
```

google
python string upper

How to calculate π

We know that the area of a circle is

$$\pi r^2$$

Looking only at the upper right corner we can see a green square with side r and we can calculate the area of the square as

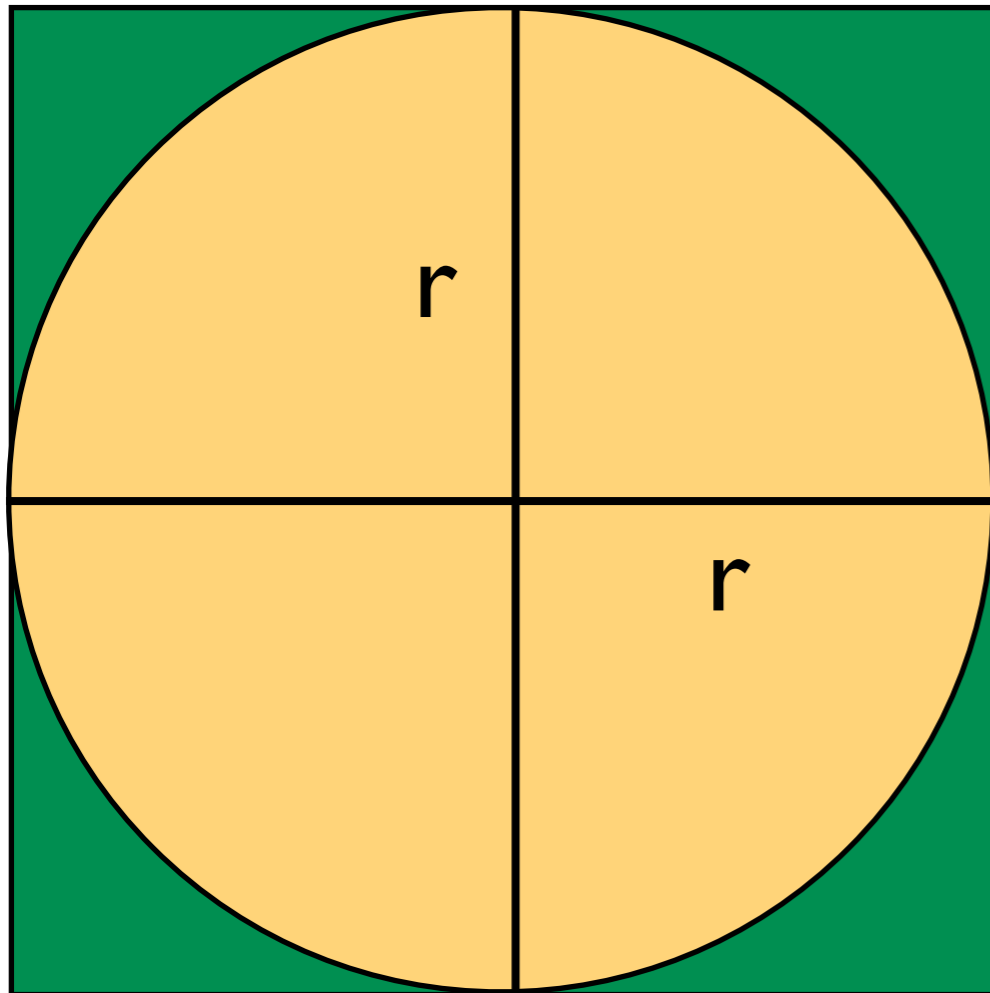
$$A_s = r^2$$

The quarter circle has the area

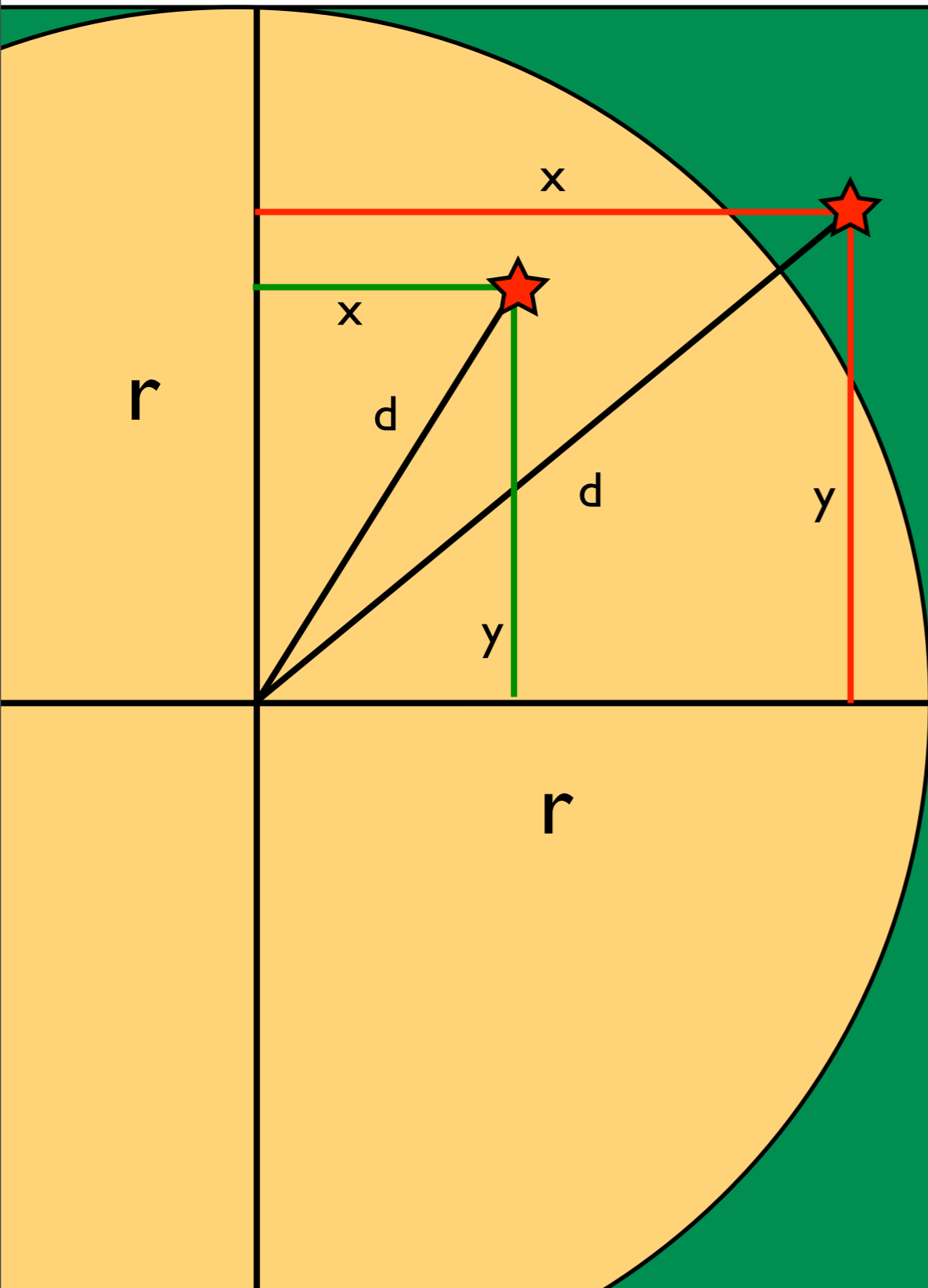
$$A_c = \frac{\pi}{4} r^2$$

So we can calculate the ratio of the two areas as

$$\frac{A_c}{A_s} = \frac{r^2}{\frac{\pi}{4} r^2} = \frac{\pi}{4}$$



How to calculate π



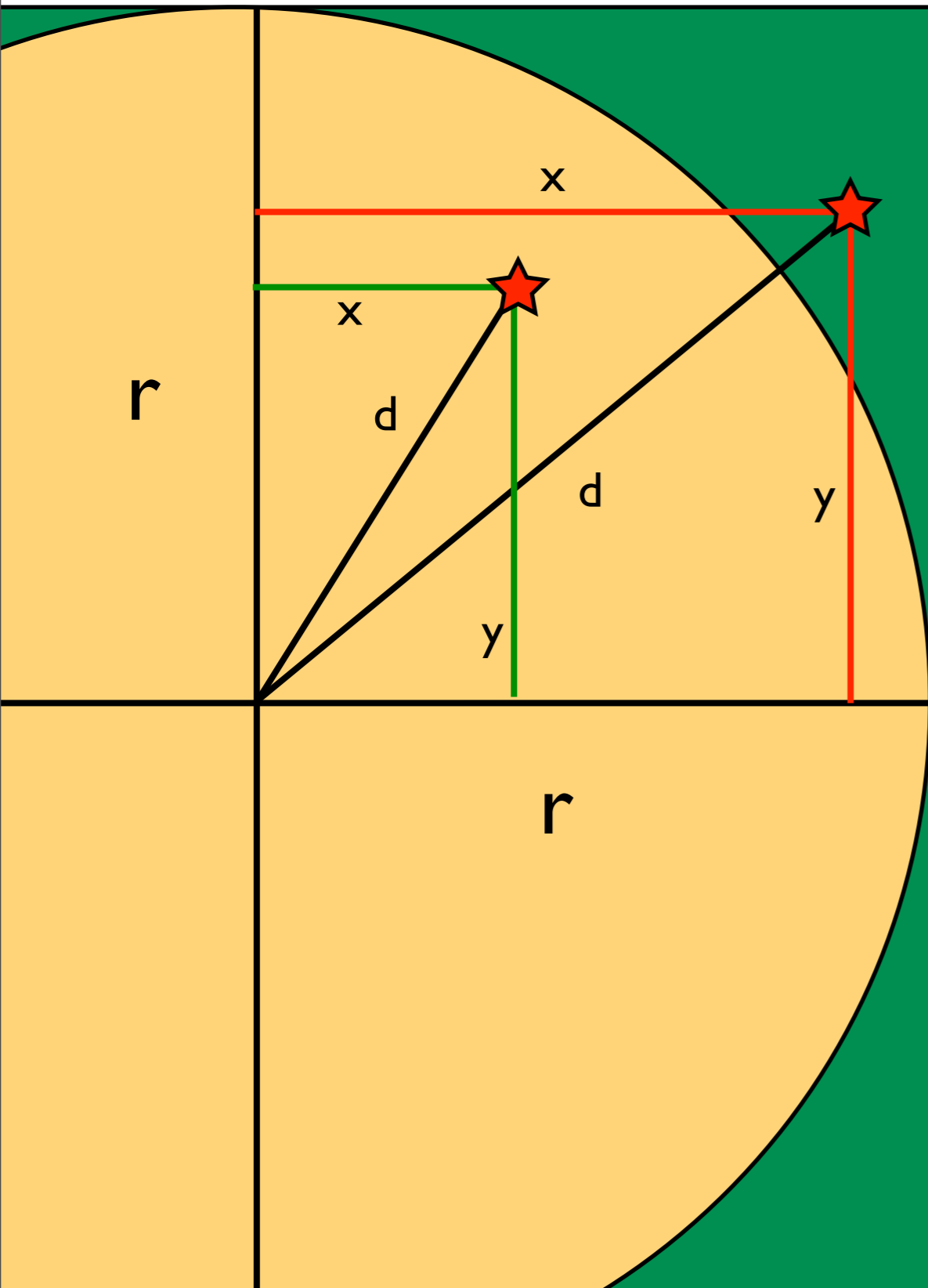
$$\frac{A_c}{A_s} = \frac{r^2}{\frac{\pi}{4}r^2} = \frac{\pi}{4}$$

The goal is now to estimate the ratio of the areas. We can devise an algorithm that draws random coordinates from the square and marks whether the coordinate fell into the circle or not. We can calculate the distance from the circle center using Pythagoras:

$$d = \sqrt{(x^2 + y^2)}$$

If d is smaller than r than we know the coordinate is in the circle otherwise only in the square. We can now create an algorithm for our program.

How to calculate π



```
# Algorithm in pseudo code
# Do many times:
#   draw x, y coordinate
#   calculate d from center
#   check whether d < r:
#     True: add 1 to circle
#     False: do nothing
#   add 1 to square
#
# print pi: ratio circle/square * 4
```

Python programming steps

Enter in file:

```
#!/usr/bin/env python
import random
import math
#initialize variables
i = 0
n = 100000
r = 1.0
circle = 0.0
square = 0.0
# Do many times:
while i < n:
    i = i + 1
    # draw x, y coordinate
    x = random.uniform(0.0,r)
    y = random.uniform(0.0,r)
    # calculate d from center
    d = math.sqrt(x**2 + y**2)
    # check whether d < r:
    if d < r:
        # True: add 1 to circle
        circle = circle + 1
        # False: do nothing

    # add 1 to square
    square = square + 1
# print pi: ratio circle/square * 4
print "pi = " + str(circle/square * 4.0)
```