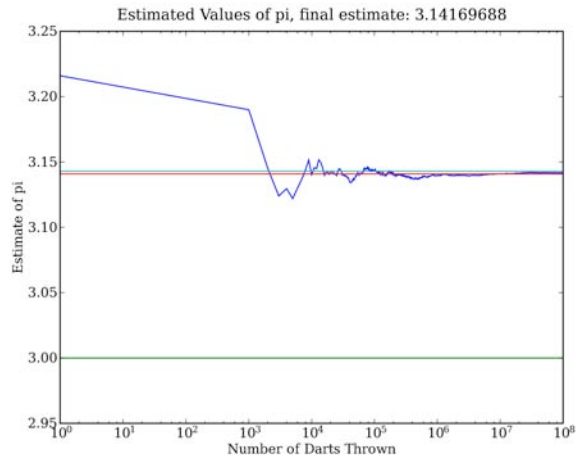
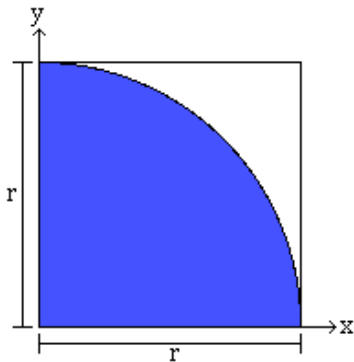


**6.00 Handout, Lecture 13**  
**(Not intended to make sense outside of lecture)**



```
import random, math, pylab

def throwDarts(numDarts, shouldPlot, maxPts):
    inCircle = 0.0
    estimates = []
    if numDarts > maxPts: divisor = numDarts/maxPts
    else: divisor = 1
    for darts in xrange(1.0, numDarts + 1, 1):
        x = random.random()
        y = random.random()
        dist = math.sqrt(x*x + y*y)
        if dist <= 1.0: inCircle += 1
        piGuess = 4*(inCircle/darts)
        if shouldPlot and darts%divisor == 0: estimates.append(piGuess)
    if shouldPlot:
        xAxis = pylab.arange(1, len(estimates)*divisor + 1, divisor)
        pylab.semilogx(xAxis, estimates)
        titleString = 'Estimated Values of pi, final estimate: ' + str(piGuess)
        pylab.title(titleString)
        pylab.xlabel('Number of Darts Thrown')
        pylab.ylabel('Estimate of pi')
        pylab.axhline(3.0)
        pylab.axhline(223/71.0)
        pylab.axhline(22/7.0)
    return 4*(inCircle/numDarts)

def findPi(numDarts, shouldPlot=True, maxPts=100000):
    piGuess = throwDarts(numDarts, shouldPlot, maxPts)
    print 'Estimated value of pi:', piGuess
```