Last name ______________________

First name ______________________

LARSON—MATH 255–CLASSROOM WORKSHEET 14
Coin Flips, Simulation & Riemann Integration.

1. Log in to your Sage/Cocalc account.
   (a) Start the Chrome browser.
   (b) Go to `http://cocalc.com` and sign in.
   (c) You should see an existing Project for our class. Click on that.
   (d) Click "New", call it **c14**, then click "Sage Worksheet".
   (e) For each problem number, label it in the Sage cell where the work is. So for Problem 1, the first line of the cell should be `#Problem 1`.

- **Coin Flip Questions** When you flip a coin 100 times would you expect to see 6 heads or tails in a row at some point?
- If you flip a coin 100 times, you would expect about 50 heads. Its possible that you could get 100 heads. What distribution of outcomes should you expect?

Here are functions we've discussed and worked on so far:

```
def coin flip():
    if random() < 0.5:
        return 'H'
    else:
        return 'T'
def coin_flips(n):
    return [coin_flip() for i in range(n)]

def longest_run(n):
    flip_results = coin_flips(n)
    countT = 0 #counter for tails
    countH = 0 #counter for heads
    longestT = 0 #longest string of tails
    longestH = 0 #longest string of heads
    for flip in flip_results:
        if flip == "H":
            countT = 0 #i'm looking at a heads, so tails counter should be 0
            countH = countH + 1
            if countH > longestH:
                longestH = countH

        else:   #flip must be "T" for tails
            countH = 0 #i'm looking at a tails, so heads counter should be 0
            countT = countT + 1
            if countT > longestT:
                longestT = countT
    return max(longestH, longestT)
```

2. If you flip a coin 100 times what is the average length of a longest run of heads or tails? We can get an idea by repeating our experiment several times, collecting the data and finding the average.

```
def repeat_experiments(n):
    total = 0.0
    for i in [1..n]:
        current_experiment = longest_run(100)
        total = total + current_experiment
    return total/n
```

Try `repeat_experiments(10)`, `repeat_experiments(100)`, and `repeat_experiments(1000)`.

3. What is the *probability* of getting a run of at least 6 heads or tails when you flip a coin 100 times?

4. Run `longest_run(100)` many times (1000 might be good) and find the average. So what would you *expect*?

Riemann Integration

Given a continuous function $f(x)$ on an interval $[a, b]$ we want to find the *area* between the curve, the x-axis and the lines $y = a$ and $y = b$. One way to do this is to use the Fundamental Theorem of Calculus and integrate. Unfortunately, it is difficult to find anti-derivatives for many (most) functions. So we need a different approach to get at least an approximate integral.

One way to do this is to slice up $[a, b]$ into n equal-sized intervals $[a_0, a_1]$, $[a_1, a_2]$,$\ldots$, $[a_n, a_{n+1}]$ (where $a_1 = a$ and $a_{n+1} = b$), pick a point c_i from each interval $[a_i, a_{i+1}]$ and compute the area $f(c_i) \cdot \Delta$ of a rectangle, where Δ is the interval length $a_{i+1} - a_i$. There are different ways to pick the c_i's. You could pick the leftmost point of the interval, the midpoint, the rightmost point, or even a random point.

The *Riemann Integral* is defined to be the *limit* of these area approximations as n goes to infinity of this quantity.

Here is our function `leftpoint_riemann(f,a,b,n)` which computes the leftpoint Riemann sums for n equal intervals.

```
def leftpoint_riemann(f,a,b,n):
    area=0
    Delta=(b-a)/n
    for i in [0..(n-1)]:
        leftpoint=a+i*Delta
        area=area+f(leftpoint)*Delta
    return area
```

5. Given a continuous function f(x) on [a,b], define a function `rightpoint_riemann(f,a,b,n)` which computes the rightpoint Riemann sums for n equal intervals.

6. Find the values of `rightpoint_riemann(f,a,b,n)`for f(x)=x**2 on [0,3] with $n = 2$, $n = 5$, $n = 10$ and $n = 100$. Compare with your results for `leftpoint_riemann(f,a,b,n)`.