# AMATH 410
# Review:
# Structuring a program in R

Our aim is to illustrate a basic strategy for how to write a new R program from scratch. A .R file with the code below is available as `cell_reproduction_review_code_1.R` from our website.

1. define given parameters
2. make a dummy vector (or matrix) of values for the variable that are going to evolve over time
3. set the initial value of that dummy vector
4. run a for loop to fill in all values of that variable (solution) vector
5. process the result, if required
6. plot

Let's say we're given this problem – similar to the cell reproduction model discussed on the first day of class.

Implement the rule

$$n(t) = 4\,p\,n(t-1)\,(1 - n(t-1)/K)$$

starting from an initial population size of $n(1) = 1000$, and simulating for 100 timesteps (or generations). Then, compute a vector of all times $t$ where $n(t) > 1300$. Use $p = 0.75$, $K = 2000$.

## Step 1

Where do I start? In my first lines of code, I define all of the parameters.

```
p=.75; #fraction of cells that survive each reproduction
generation_end=100;  #predict up to this generation in the future
K=2000;   #reproduction capacity
```

## Step 2

OK, now I have to think about what it is that I am going to **evolve over time.** I am going to predefine a "dummy" vector or matrix of numbers representing the variable that I am going to simulate. That dummy vector is often just filled in with zeros. Then I will go through and fill it in later.

Here, I want a vector of values of $n$ at timesteps $t = 1, ..., 100$. So, let's just define a vector of 100 zeros to use as my dummy vector. There are a couple of ways to do this:

```
nvector=rep(0,generation_end);
```

or

```
nvector=0*(1:generation_end) ;
```

## Step 3

Alright. Now I need to set my initial conditions. What's the first value of nvector? That's

```
nvector[1]=1000;
```

## Step 4

Now I need to loop over time. At each timestep, I implement my dynamical rule.

```
for (t in 2:generation_end)
  nvector[t] <- 4*nvector[t-1]*p*(1 - nvector[t-1]/K)
```

So, I went through and filled in the dummy zero values of nvector with their real values.

## Step 5

Now, I need to make a new vector – of generations t when `nvector[t]>30`. We can do this in two ways: first, append a vector called `vector_of_times_over_limit` as we calculate `nvector`. Secondly, we can calculate `nvector` for all `t`, and then create a vector that has the values greater than 30.

### Method 1

```
vector_of_times_over_limit <- vector()

for (t in 1:generation_end)
  if (nvector[t] > 30)
    vector_of_times_over_limit <- c(vector_of_times_over_limit, t)
```

### Method 2

```
vector_of_times_over_limit <- which(nvector[t] > 30)
```

## Step 6

OK, let's plot $n$ vs $t$. I'll make a vector of times $t$ first:

```
tvector=1:generation_end

plot(tvector, nvector, pch = ".", cex = 1.2,
     main = "My First Plot in R", xlab = "t", ylab = "N(t)")
```