Practice Exam # 2:

1. (10 points) Plot the function \( f(x) = x^2 \) in red for values of \( x \) ranging between -10 and 10. Plot the function \( g(x) = x^3 \) in green for values of \( x \) ranging between -5 and 5. Overlay these plots. (Note: you are not required to define the functions.)

2. (10 points) The script below should iterate the function \( f(x) = 3x \) five times with an initial value of 1, but has five errors. Correct the errors and explain what each line does in a comment. If the script works correctly, it should generate the output at the bottom of the script. In a new cell, convert the script into a function that takes the number of iterations as input and returns the list of values. Test the function for three different numbers of iterations.

```python
mult3 = [1]
ums = srange(0,5)
for n in nums
    test = 3*mult3[i]
    append.mult3(test)
Mult3
# Desired output: [1, 3, 9, 27, 81, 243]
```

3. (10 points) Modify the example code below to simulate the model given for time steps (step size) of 0.2 and initial conditions of \( N=100 \) and \( P=20 \). Plot the trajectory as a smooth curve in pink. Plot the vector field for the model given with \( N \) ranging between 0 and 150 and \( P \) ranging between 0 and 120.

```python
var("N,P")
t = srange(0,100,0.1)
sol=desolve_odeint([0.5*N - 0.01*N*P, 0.5*0.01*N*P - 0.2*P], ics=[50,75], dvars=[N,P], times=t)
list_plot(zip(t, sol[:,0])) + list_plot(zip(t,sol[:,1]), color="red")
```

4. (10 points) Write a script that generates the complementary strand (as a list) to a DNA sequence with the variable name `dna_seq` with any number of nucleotides (A,T,C,G) (A and T are complementary, C and G are complementary). For example, if the script is given `dna_seq = "ACTGACTGAC"`, it should print “TGACTGACTG”. The variable `dna_seq` can be a string or list of characters such as `['A','C','T']` with only A,C,T,G. You don’t need to consider the possibility of other characters.