# BIFURCATION DIAGRAM OF THE LOGISTIC MAP
# Programming Assignment 4 (20 points)

Math 248 Computers and Numerical Algorithms
Fall 2008–Pruett

**DATE ASSIGNED**: Wednesday, 1 October, 2008 **DATE DUE**: Tuesday, 14 October, 2008

**BACKGROUND**: Recall that the logistic map $x_{i+1} \leftarrow \alpha x_i(1.0-x_i)$ can yield convergent, periodic, chaotic, or unbounded orbits, depending on the value of the parameter $\alpha$. Instead of appealing to individual graphs for each parameter value, as we did in the last lab, we can cleverly assemble all of the information into a single "bifurcation diagram" with the parameter value $\alpha$ along the horizontal axis and values of the iterates $x_i$ along the vertical axis. Recall that, when the parameter value is relatively low, the orbits are convergent. Thus, we need plot only $\alpha$ vs. its corresponding fixed-point $p(\alpha)$. At some threshold value of $\alpha$, call it $\alpha_2$, the behavior of the iteration changes qualitatively, and the orbit becomes periodic with a period of 2. The graph is said to bifurcate; that is, the single curve splits into two curves, each representing one of the two periodic iterates. Because the graph resembles a pitchfork, this is sometimes referred to as a "pitchfork bifurcation." Amazingly, as the parameter is increased beyond a second threshold, call it $\alpha_4$, the period doubles. That is, the orbit becomes periodic with period 4! Graphically, each branch of the pitchfork experiences a second pitchfork bifurcation. The pattern is repeated; at some value $\alpha_8$, the period 4 orbit bifurcates to a period 8 orbit, an so on, to infinity! Eventually, beyond a certain finite value $\alpha_\infty$, the orbits become *aperiodic* (chaotic) in that they never settle down into any recognizable pattern. More remarkably, there are narrow bands of $\alpha$ values, buried within the chaotic region of $\alpha$, that yield periodic orbits with periods 3, 6, 12,..., or 7, 14, 28..., etc. Most remarkably, the entire bifurcation diagram is *fractal*-like. And the mysteries go on and on, enough to fill books and to keep a small army of mathematicians busy for decades. For clarification or further reading, see *Chaos, Making a New Science* by James Gleick.

**ASSIGNMENT: (B-Level)** Write a Fortran 90 program that 1) computes $n$ iterates of the logistic map for $m$ evenly spaced values $\alpha_0 < \alpha < 4$; 2) writes out to file LOG.DAT the iterates as a two-column table with $m \times n$ pairs ($\alpha$,$x_i$). While developing your program, plot the values using the MATLAB x-y plotting program found on Blackboard (ASSIGNMENTS) that we used for LAB 7. If done correctly, your plot should be the bifurcation diagram of the logistic equation. Submit your program using the auto-submission protocol, which will compile your program, execute your program, AND generate the MATLAB plot automatically. **(A-Level)** In addition to the B-level assignment, look at the ratios $\frac{\alpha_4-\alpha_2}{\alpha_8-\alpha_4}$, $\frac{\alpha_8-\alpha_4}{\alpha_{16}-\alpha_8}$, etc. The limit of this sequence of ratios is a *universal* constant (like $\pi$ or $e$) discovered by Mitchell Feigenbaum at Los Alamos National Laboratory in 1976. Find this value to 2 decimal places, documenting your work in comment lines appended to your program.

**SPECIAL CONSIDERATIONS**: This project may be done alone or with a partner. Things only get really interesting for $\alpha > 3$, so, choose $\alpha_0$ to be, say, 2.5. You do NOT want to output all iterates, only those after the orbit has settled into its *equilibrium state*. So, for each $\alpha$, discard the first $n_1$ iterates before you start writing out the data. In general, your data file will contain $(n-n_1) \times m$ values and can become quite large. Use care in deciding how big $m$ and $n$ should be. Otherwise, your disk will fill up quickly! For practical reasons, limit the total number of lines in your output file to roughly 10,000.

*As always, you are allowed to discuss programming issues with other students, but the actual coding of your program should be accomplished individually (or with your partner) and your program should be unique.*