<pre>In [2]: import numpy as np import matplotlib.pyplot as plt def fx(n, r, x0, x1): y = [x0] y1 = [x1] a = [1] def f(x, r): b = 4 * r * x * (1 - x) return b for i in range(n-1): a.append(i+2) y.append(f(y[-1], r)) y1.append(f(y1[-1], r)) plt.plot(a, y, 'r', a, y1, 'g') plt.show</pre>
In [3]: fx(100, 0.892486418, 0.5, 0.51) 0.9 0.8 0.7 0.6 0.5 0.4 0.20 40 60 80 100
In [4]: fx(200, 0.91, 0.5, 0.501) 0.9 0.8 0.7 0.6 0.5 0.4 0.3 0 25 50 75 100 125 150 175 200
In [40]: fx(100, 0.85, 0.5, 0.51) 0.8 0.7 0.6 0.5 0.7 0.6 0.7 0.7 0.7 0.8 0.8 0.9 0.9 0.9 0.9 0.9 0.9
<pre>Im [41]: import numpy as np import matplotlib.pyplot as plt def fx(n, r, x0, x1): y = [x0] y1 = [x1] a = [1] def f(x, r): b = 4 * r * x * (1 - x) b = b/10 b = b*10 return b for i in range(n-1): a.append(f(y[-1], r)) y[-1] = y[-1]/10 y[-1] = y[-1]/10 y1.append(f(y[-1], r)) y1[-1] = y1[-1]/10 y1[-1] = y1[-1]/10</pre> plt.plot(a, y, 'r', a, y1, 'g') plt.show
In [42]: fx(100, 0.892486418, 0.5, 0.51) 0.9 0.7 0.6 0.5 0.4 0.7 0.6 0.5 0.7 0.6 0.7 0.7 0.8 0.9 0.9 0.9 0.9 0.9 0.9 0.9
In [43]: fx(200, 0.91, 0.5, 0.51) 0.9 0.7 0.6 0.5 0.4 0.3 0 25 50 75 100 125 150 175 200
<pre>In [6]: import matplotlib.pyplot as plt def fx(n, r, x0, l): y = [x0] y1 = [x0] a = [1] def f(x, r): b = 4 * r * x * (1 - x) return b for i in range(n-1): a.append(i+2) y.append(f(y[-1], r)) k = f(y1[-1], r)/1 k = k *1 y1.append(k) plt.plot(a, y, 'r', a, y1, 'g') plt.show</pre>
In [9]: fx(200,0.92,0.2,64) 0.9 0.8 0.7 0.6 0.5 0.4 0.3 0.2 0.5 0.5 0.4 0.5 0.5 0.6 0.7 0.7 0.6 0.7 0.7 0.7 0.8 0.9 0.9 0.9 0.9 0.9 0.9 0.9
In [85]: fx(200,0.2,0.2,10) 0.200 0.175 0.150 0.125 0.000 0.075 0.050 0.025 0.000 0.25 50 75 100 125 150 175 200
In [86]: fx(200,0.2,0.2,16) 0.200 0.175 0.150 0.100 0.075 0.050 0.025 0.000 0.25 50 75 100 125 150 175 200
In [96]: fx(200,0.2,0.2,20) 0.200 0.175 0.150 0.125 0.000 0.075 0.050 0.025 0.000 0.25 50 75 100 125 150 175 200
In [10]: fx(200,0.958,0.8,10) 0.9 0.8 0.7 0.6 0.5 0.4 0.3 0.2 0.5 50 75 100 125 150 175 200
<pre>Im [91]: import numpy as np import matplotlib.pyplot as plt def fx(n, r, x0, x1): y = [x0] y1 = [x1] a = [1] def f(x, r): b = 4 * r * x * (1 - x) b = b * 10 return b for i in range(n-1): a.append(i+2) y.append(f(y[-1], r)) y[-1] = y[-1]/10 y[-1] = y[-1]/10 y1.append(f(y[-1], r)) y1[-1] = y1[-1]/10 y1[-1] = y1[-1]/10 y1[-1] = y1[-1]/10 p1t.plot(a, y, 'r', a, y1, 'g') plt.show</pre>
In [93]: fx(100, 0.958, 0.5, 0.51) 0.9 - 0.6 - 0.5 - 0.4 - 0.3 - 0.2 - 0.4 - 0.3 - 0.2 - 0.4 - 0.3 - 0.2 - 0.4 - 0.3 - 0.2 - 0.5 - 0.4 - 0.5 - 0.5 - 0.4 - 0.5 -