

Metody numeryczne - N11

DAMIAN PORADYŁO

N11 *Zadanie numeryczne*

Narysować zbiór $\{x_n : n > 100\}$ (atraktor) w zależności od parametru $k \in [2, 4]$ dla odwzorowania logistycznego.

$$x_{n+1} = kx_n(1 - x_n). \quad (12)$$

http://pl.wikipedia.org/wiki/Odwzorowanie_logistyczne

1 OPIS ZADANIA

Program został zaimplementowany zgodnie z artykułem na Wikipedii, do którego prowadzi link w treści zadania.

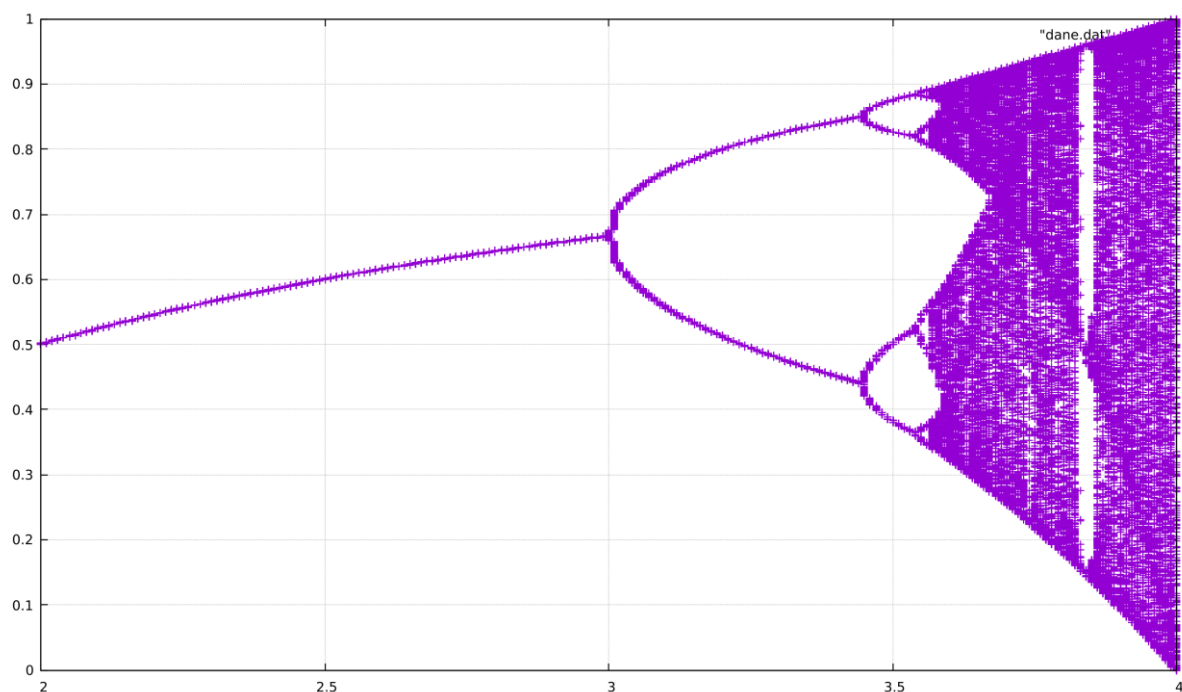
Co nas powinno zaniepokoić w wykresie to to, że:

Przy wartości $k = 3$, pojawiają się dwa nowe punkty przyciągania. Rozdwojenie to, nazywa się **bifurkacją**.

Dla wartości $k \sim 3.828427$ występuje tzw. ostatnie okno stabilności. W tym oknie nasze odwzorowanie logistyczne przyjmuje stabilny atraktor trzypunktowy, który wraz ze wzrostem parametru k podwaja się i przechodzi w chaos.

Przy parametrze $k = 4$, widzimy, że nasz układ zachowuje się chaotycznie. Atraktor pokrywa cały dostępny przedział $(0;1)$

2 WYKRES



Do stworzenia wykresu posłużyłem się plikiem wejściowym, stworzonym przez program *dane.dat* a także programem *gnuplot*.

3 KOD PROGRAMU

```
#include <iostream>
#include <cmath>
#include <vector>
#include <iomanip>
#include <map>

int main(int argc, char const *argv[]) {
    std::map<double, std::vector<double>> mapWithValues;
    std::vector<double> values;

    double r = 2.0;
    double x = 0.5;
    int end_step = 500;

    while(r<=4) {
        values.clear();
        for(int j=100; j<end_step; j++) {
            x = (r*x*(1-x));
            values.push_back(x);
        }
        mapWithValues[r] = values;
        r+=0.005;
    }

    for(auto const& iterator1: mapWithValues) {
        for(auto const& iterator2: iterator1.second) {
            std::cout << std::setprecision(5) << iterator1.first << "\t" <<
std::setprecision(15)<< iterator2 << std::endl;
        }
    }

    return 0;
}
```

4 ZAWARTOŚĆ

>>make run

Zawartość:

- Opracowanie
- Plik programu
- Wykres
- Makefile