

Transmisja Danych – Lab 01

Krystian Bartosik 213A, nr 44266

Kod źródłowy:

```
// Krystian Bartosik
// bk44266@zut.edu.pl
// FEDCBA
#define _USE_MATH_DEFINES

#include <iostream>
#include <fstream>
#include <math.h>

using namespace std;

double x(double t)
{
    return pow(6 * t, 2) + 6 * t + 2;
}

double y(double t)
{
    return 2 * pow(x(t), 2) + 12 * cos(t);
}

double z(double t)
{
    return sin(2*M_PI*7*t)*x(t)-0.2*log10(abs(y(t))+M_PI);
}

double u(double t)
{
    return sqrt(abs(y(t) * y(t) * z(t))) - 1.8 * sin(0.4 * t * z(t) * x(t));
}

double v(double t)
{
    if ((0.22 > t) && (t >= 0))
    {
        return (1 - 7 * t) * sin((2 * M_PI * t * 10) / (t + 0.04));
    }

    if ((0.22 <= t) && (t < 0.7))
    {
        return 0.63 * t * sin((2 * M_PI * t * 10) / (t + 0.04));
    }

    if ((1 >= t) && (t >= 0.7))
    {
        return pow(t, -0.662) + 0.77 * sin(8 * t);
    }
}
```

```

double p(double t, int N)
{
    double SUM = 0;

    for (int n = 1; n <= N; n++)
    {
        SUM = SUM + ((cos(12 * t * pow(n, 2)) + cos(16 * t * n)) / (pow(n, 2)));
    }

    return SUM;
}

int main()
{
    fstream File;
    File.open("C:/Users/Qrystian/Desktop/results.txt", ios::out);

    // Zadanie 1
    for (double t = -10; t <= 10; t = t + 1.0/100.0)
    {
        File << t << " " << x(t) << endl;
    }

    double Delta = pow(6, 2) - 4 * 6 * 2;
    if (Delta < 0)
    {
        cout << "Delta < 0, brak miejsc zerowych" << endl;
        return 1;
    }

    if (Delta == 0)
    {
        double t0 = -6 / (2 * 2);
        cout << "Miejsce zerowe: " << t0 << endl;
        cout << "Delta: " << Delta << endl;
        return 1;
    }

    if (Delta > 0)
    {
        double t1 = -6 + sqrt(Delta) / (2 * 2);
        double t2 = -6 - sqrt(Delta) / (2 * 2);
        cout << "Miejsca zerowe: " << t1 << " " << t2 << endl;
        cout << "Delta: " << Delta << endl;
        return 1;
    }

    // Zadanie 2
    for (double t = 0; t <= 1; t = t + 1.0 / 22050.0)
    {
        //File << t << " " << y(t) << endl;
        //File << t << " " << z(t) << endl;
        //File << t << " " << u(t) << endl;
        //File << t << " " << v(t) << endl;
        //File << t << " " << p(t,2) << endl;
        //File << t << " " << p(t, 4) << endl;
        File << t << " " << p(t, 66) << endl;
    }

    File.close();
}

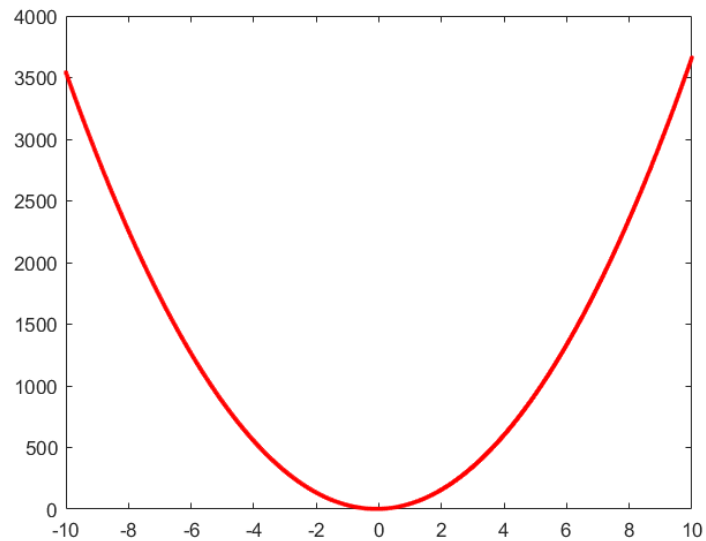
```

Opis kodu:

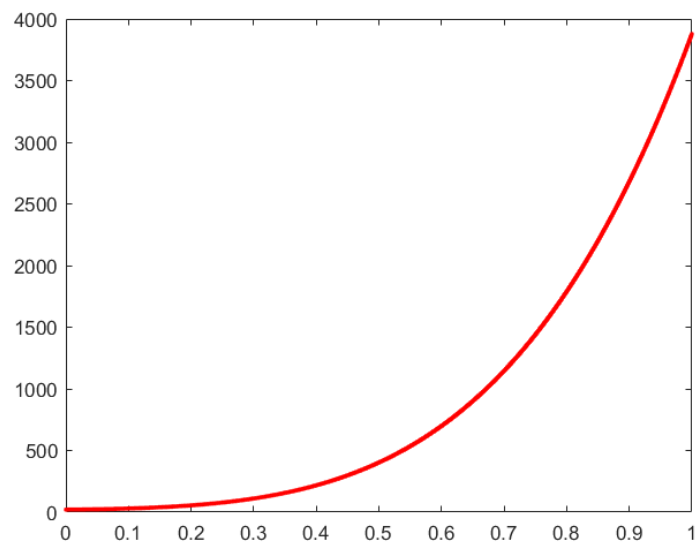
- **Zadanie 1**
- **Zadanie 2**

Generowanie wykresów: wyniki operacji zostały zapisane do pliku tekstowego, zaimportowane w programie MATLAB następnie odpowiednio narysowane.

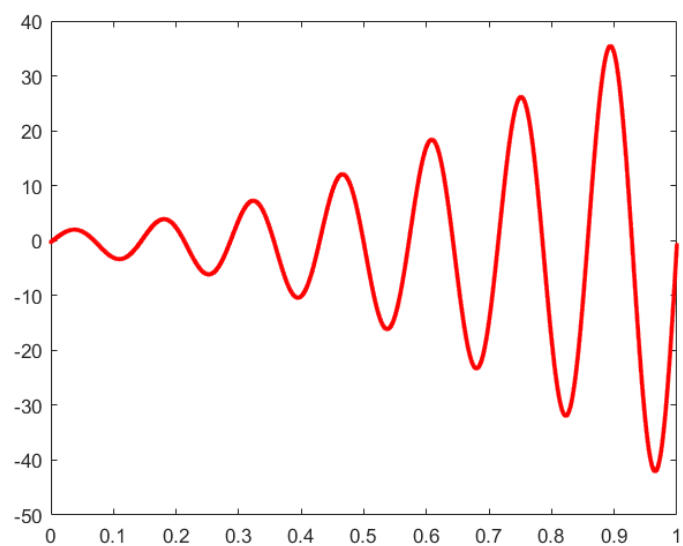
Wygenerowane wykresy:



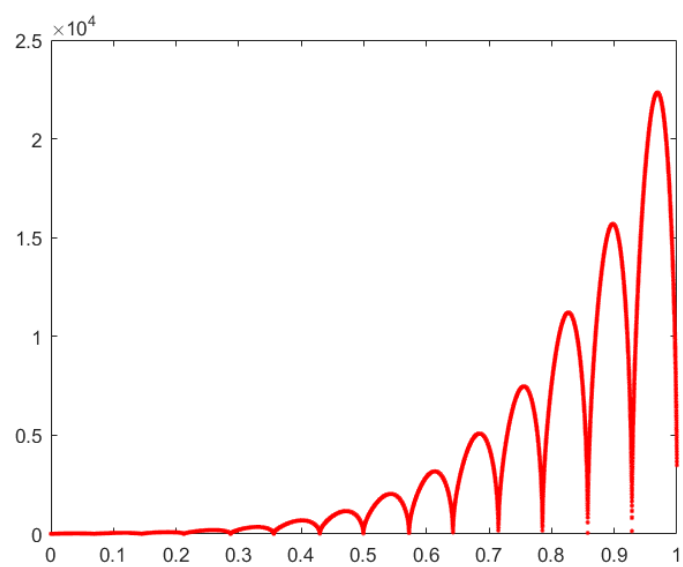
Wykres 1



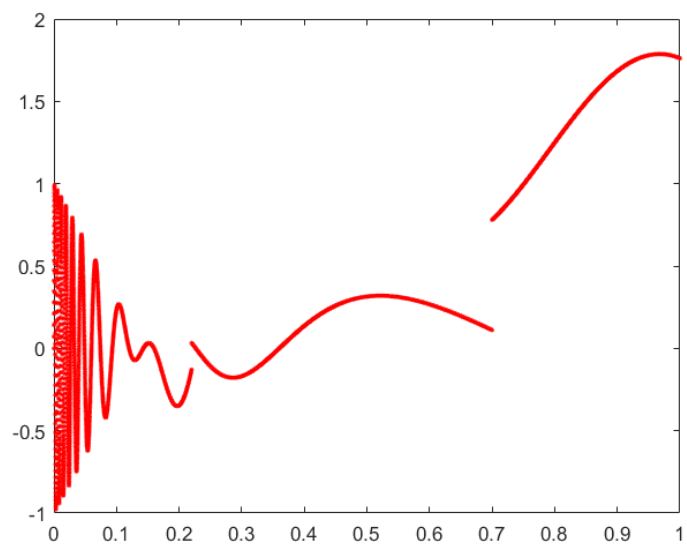
Wykres 2



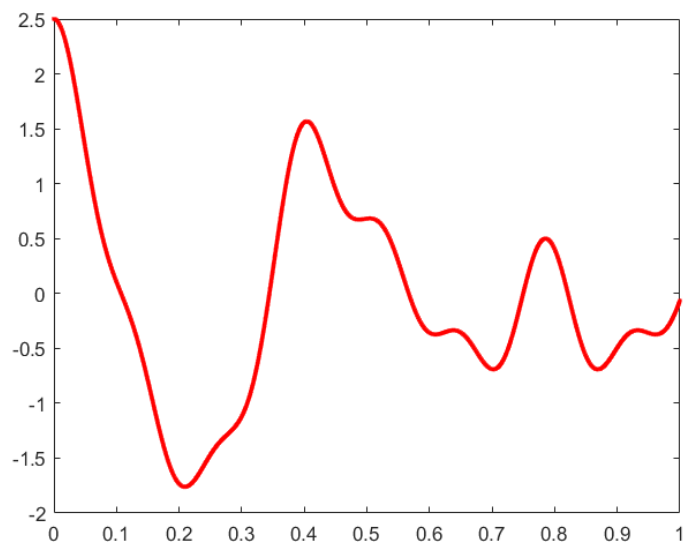
Wykres 3



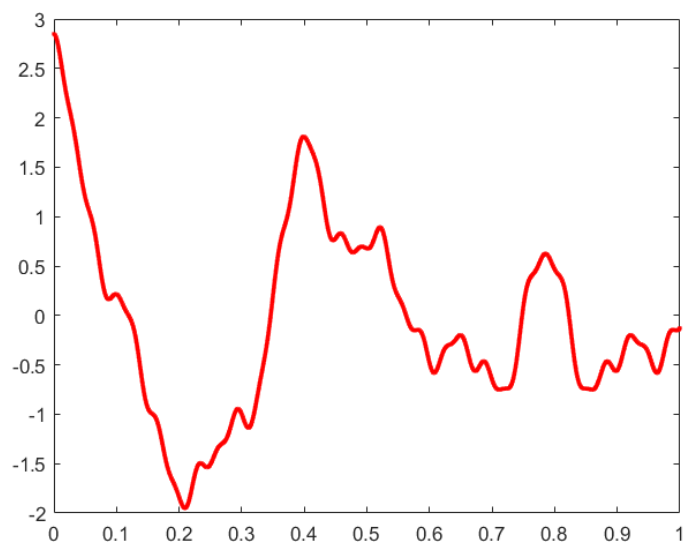
Wykres 4



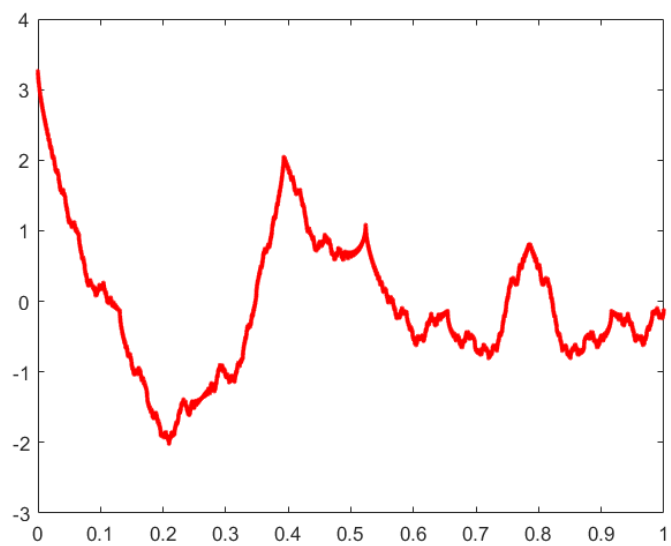
Wykres 5



Wykres 6



Wykres 7



Wykres 8