

Sprawozdanie z LAB 01

Zadaniem zajęć było znaleźć rozwiązania podanych funkcji oraz sporządzić ich wykresy.

Pierwszy wzór był funkcją kwadratową, natomiast kolejne to złożenia następnych funkcji. Oto kod źródłowy rysujący funkcję:

```

std::pair<double, double> wyroznik_trojmiannu_wykres(double a, double b, double c, double const start, double const koniec, double dok)
{
    int delta = pow(b, 2) - (4 * (long long)a * c);

    std::fstream delta_wykres, mz_wykres;
    delta_wykres.open("wykresy\\function.dat", std::ios::out | std::ios::trunc);

    long long l_dok = (koniec - start) / dok;
    double actual = start;
    for (int i = 0; i < l_dok; i++)
    {
        delta_wykres << actual << " " << (pow(actual, 2) * a + actual * b + c) << std::endl;
        actual += dok;
    }

    delta_wykres.close();

    std::pair<double, double> wynik;
    std::string option;

    mz_wykres.open("wyroznik_trojmiannu\\points.dat", std::ios::out | std::ios::trunc);
    if (delta < 0)
    {
        wynik = std::pair<double, double>(NaN, NaN);
        option = "set label 'brak miejsc zerowych' at graph 0.5,0.5 center font 'Arial, 24'; ";
    }
    else if (delta == 0)
    {
        wynik = std::pair<double, double>(((((-1) * b) / (2 * a)), NaN);
        mz_wykres << wynik.first << " " << (pow(wynik.first, 2) * a + wynik.first * b + c) << std::endl;
        option = "set label 'jedno miejsce zerowe' at graph 0.5,0.5 center font 'Arial, 24'; ";
    }
    else
    {
        wynik = std::pair<double, double>((sqrt(delta) - b) / (2 * a), (-sqrt(delta) - b) / (2 * a));
        mz_wykres << wynik.first << " " << (pow(wynik.first, 2) * a + wynik.first * b + c) << std::endl;
        mz_wykres << wynik.second << " " << (pow(wynik.second, 2) * a + wynik.second * b + c) << std::endl;
        option = "set label 'dwa miejsca zerowe' at graph 0.5,0.5 center font 'Arial, 24'; ";
    }
    mz_wykres.close();

    std::vector<std::string> files{ "function.dat", "points.dat" };
    std::vector<std::string> plot_option{ "pointtype 6 pointsize 0.5 lc rgb 'red' title 'funkcja " + to_string((int)a) + "*x^2 + " + to_string((int)b) + "*x + " + to_string((int)c) + "' "
    , "pointtype 3 pointsize 3 lc rgb 'black' title 'miejsca zerowe'" };

    //print("C:/Users/GSzwia/source/repos/TD_2020_44522/wykresy", "wyroznik trojmiannu", files, plot_option, "wyróżnik trójkątnu kwadratowego", option);
    return wynik;
}

```

Funkcje zapisałem w następujący sposób:

```
double fun_x(double const& t)
{
    return (pow(t, 2) * 2 + t * 2 + 4);
}

double fun_y(double const& t)
{
    return (2 * pow(fun_x(t), 2) + 12 * cos(t));
}

double fun_z(double const& t)
{
    return (sin(2 * M_PI * 7 * t) * fun_x(t) - 0.2 * log10(abs(fun_y(t)) + M_PI));
}

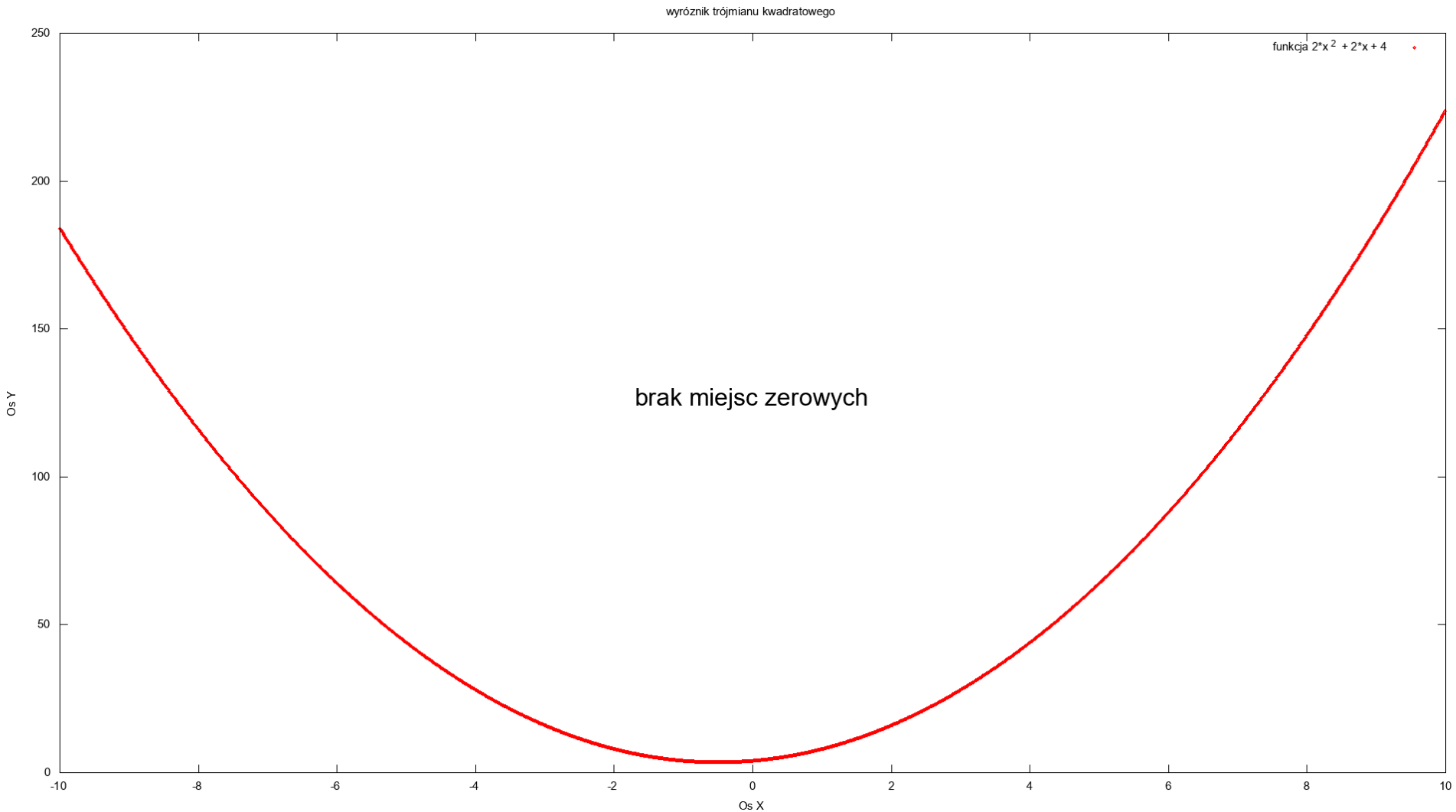
double fun_u(double const& t)
{
    return (sqrt(abs(fun_y(t) * fun_y(t) * fun_z(t))) - 1.8 * sin(0.4 * t * fun_z(t) * fun_x(t)));
}

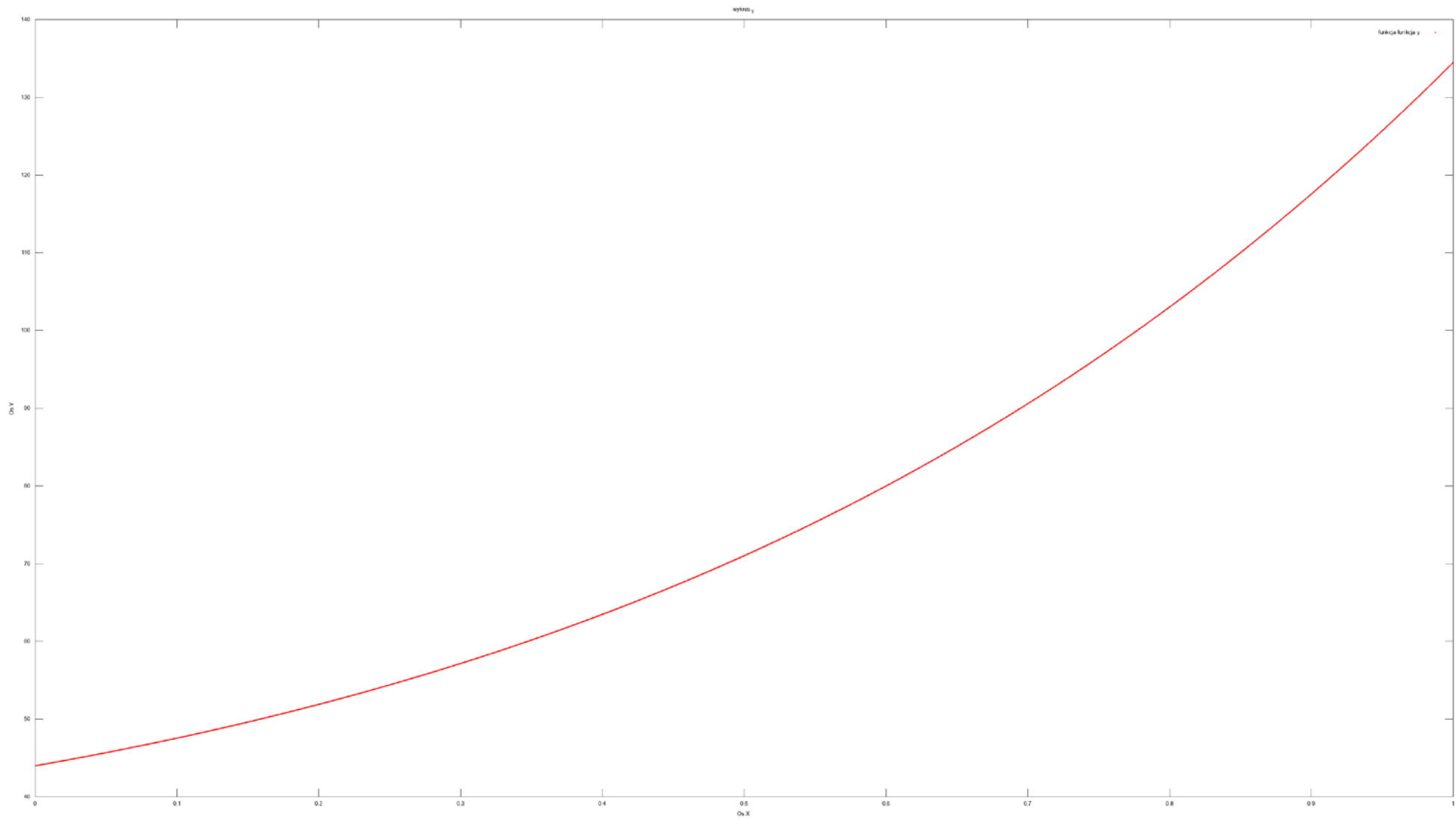
double fun_v(double const& t)
{
    if (t < 0.22)
        return ((1 - 7 * t) * sin((2 * M_PI * t * 10) / (t + 0.04)));
    else if (t < 0.7)
        return (0.63 * t * sin(125 * t));
    else
        return (pow(t, -0.662) + 0.77 * sin(8 * t));
}

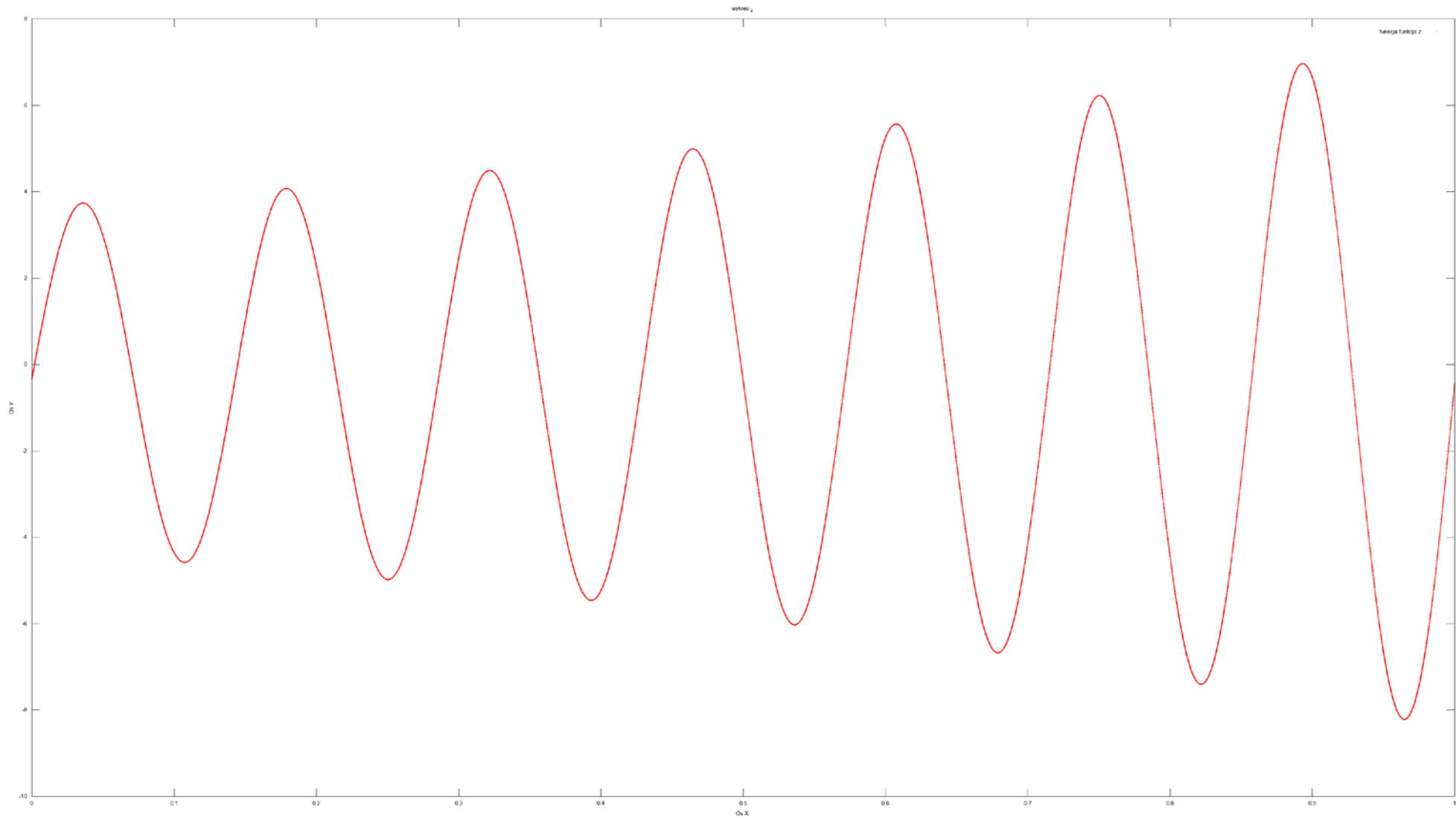
int N;

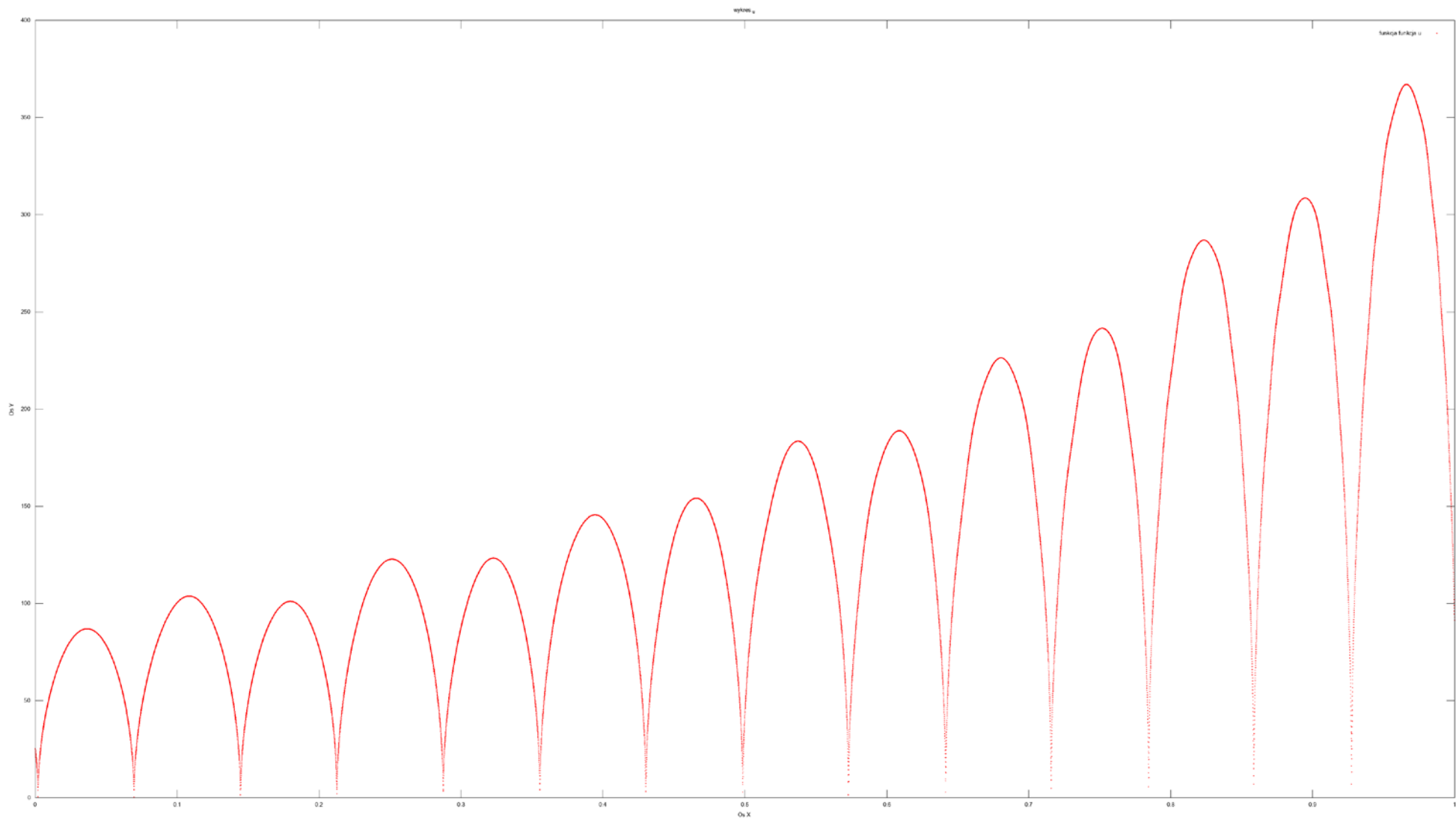
double fun_p(double t)
{
    double sum = 0;
    for (int i = 0; i < N; i++)
    {
        sum += ((cos(12 * t * pow(N, 2)) + cos(16 * t * N)) / ((pow(N, 2) == 0.0 ? 1 : pow(N, 2))));
    }
    return sum;
}
```

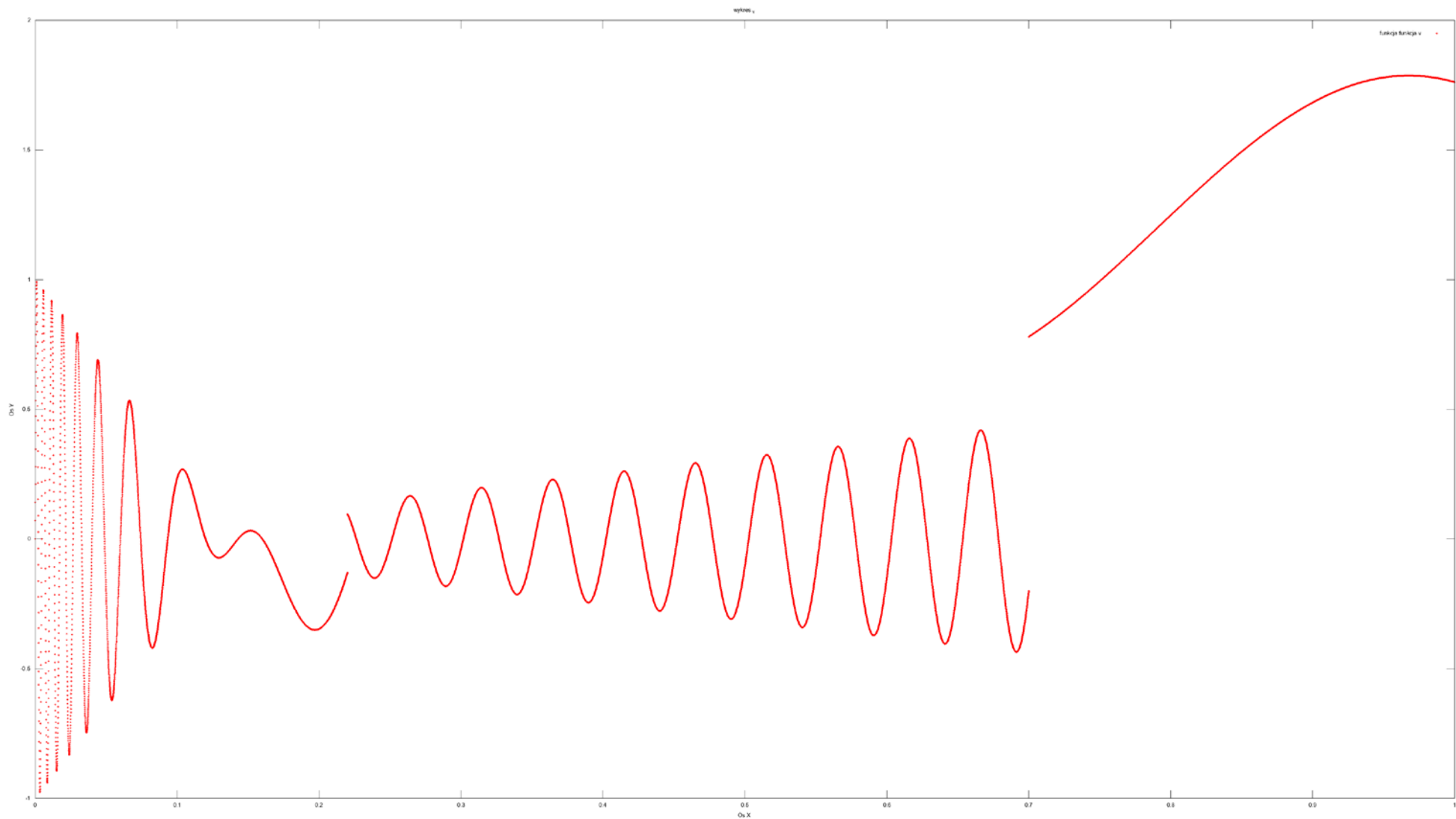
Podczas zajęć sporządziłem 8 wykresów.

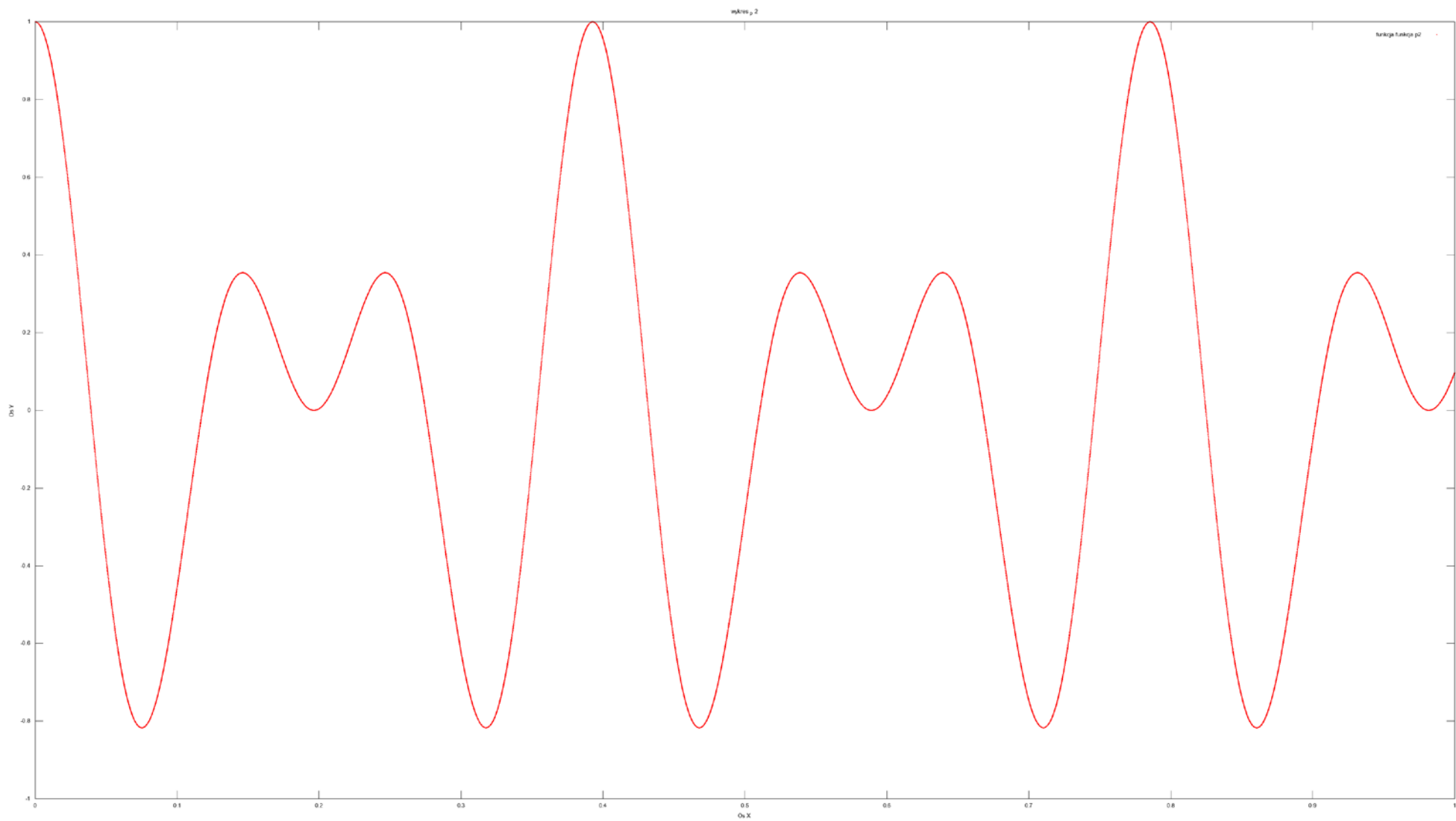


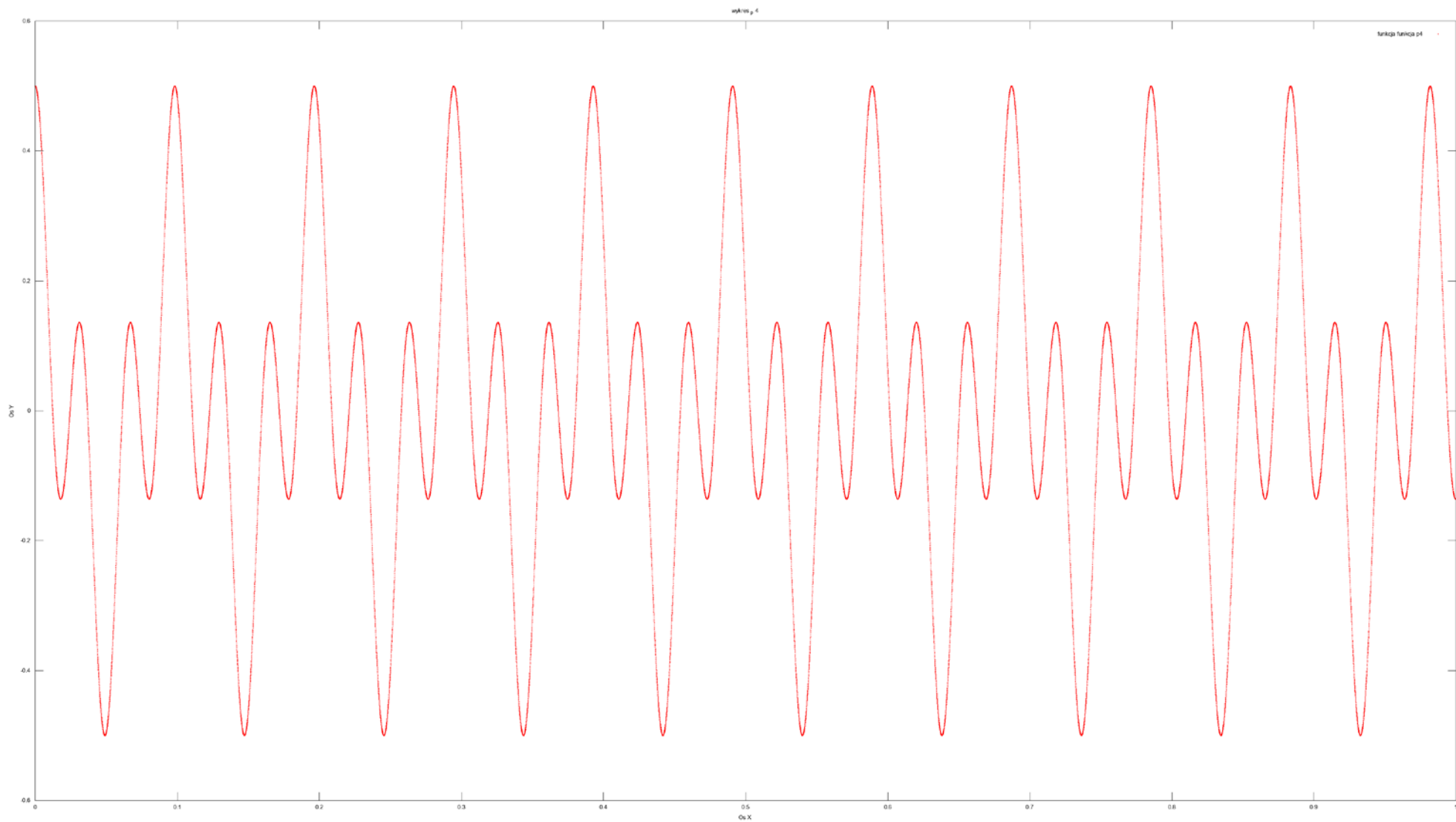


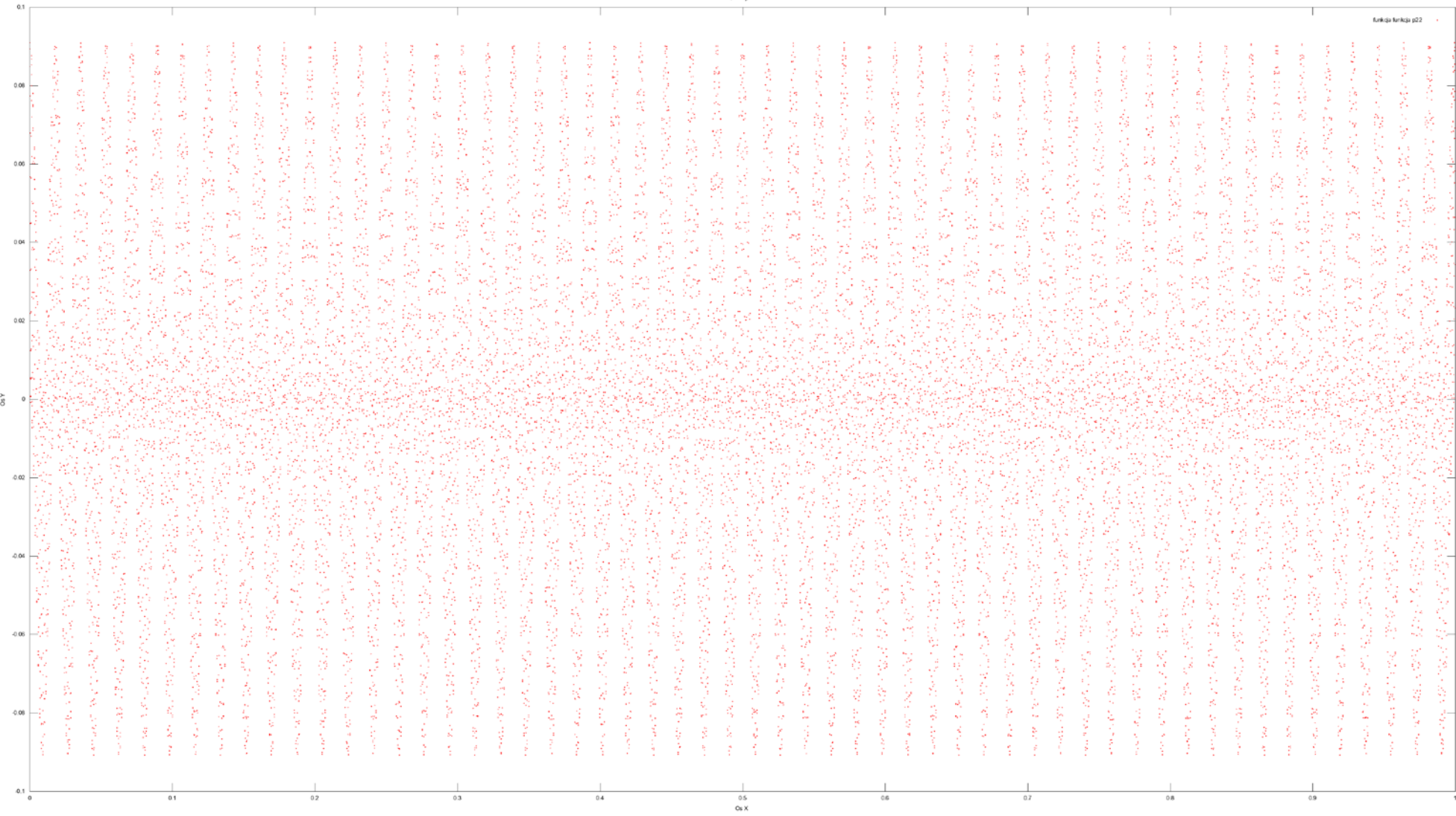












Podsumowanie

Dzięki tym laboratoriom nauczyłem się sporządzać wykresy zadanych funkcji za pomocą różnych środowisk oraz obliczać je i implementować w c++.

Wykresy w głównej mierze przypominają różne sygnały odbierane ze świata realnego.

Również zapoznaliśmy się z systemem oceniania oraz sposobie udostępniania prac.

Wykonał Szwarc Grzegorz