

Blog Entry © Wednesday, August 27, 2025, by James Pate Williams, Jr. and the CACM
ALGOL60 Algorithms for Calculating the Chebyshev, Hermite, Laguerre, and Legendre
Orthogonal Polynomials

```
#pragma once
class OrthoPolys
{
public:
    //https://algot60.org/acm/010.pdf
    static double Chebyshev(double x, int n);
    // https://algot60.org/acm/011.pdf
    static double Hermite(double x, int n);
    // https://algot60.org/acm/012.pdf
    static double Laguerre(double x, int n);
    // https://algot60.org/acm/013.pdf
    static double Legendre(double x, int n);
};
```

```
#include "pch.h"
#include "OrthoPolys.h"
```

```
double OrthoPolys::Chebyshev(double x, int n)
{
    double a = 0.0, b = 0.0, c = 0.0;

    a = 1.0;
    b = x;

    if (n == 0)
    {
        c = a;
    }

    else if (n == 1)
    {
        c = b;
    }

    else
    {
        for (int i = 2; i <= n; i++)
        {
            c = 2.0 * x * b - a;
            a = b;
            b = c;
        }
    }
}
```

```

        return c;
    }

double OrthoPolys::Hermite(double x, int n)
{
    double a = 0.0, b = 0.0, c = 0.0;

    a = 1;
    b = 2.0 * x;

    if (n == 0)
    {
        c = a;
    }

    else if (n == 1)
    {
        c = b;
    }

    else
    {
        for (int i = 1; i <= n - 1; i++)
        {
            c = 2.0 * x * b - 2.0 * i * a;
            a = b;
            b = c;
        }
    }

    return c;
}

double OrthoPolys::Laguerre(double x, int n)
{
    double a = 0.0, b = 0.0, c = 0;

    a = 1.0;
    b = 1.0 - x;

    if (n == 0)
    {
        c = a;
    }

    else if (n == 1)

```

```

    {
        c = b;
    }

    else
    {
        for (int i = 1; i <= n - 1; i++)
        {
            c = (1.0 + 2.0 * i - x) * b - i * i * a;
            a = b;
            b = c;
        }
    }

    return c;
}

double OrthoPolys::Legendre(double x, int n)
{
    double a = 0.0, b = 0.0, c = 0.0;

    a = 1.0;
    b = x;

    if (n == 0)
    {
        c = a;
    }

    else if (n == 1)
    {
        c = b;
    }

    else
    {
        for (int i = 1; i <= n - 1; i++)
        {
            c = b * x + i * (x * b - a) / (i + 1);
            a = b;
            b = c;
        }
    }

    return c;
}

```

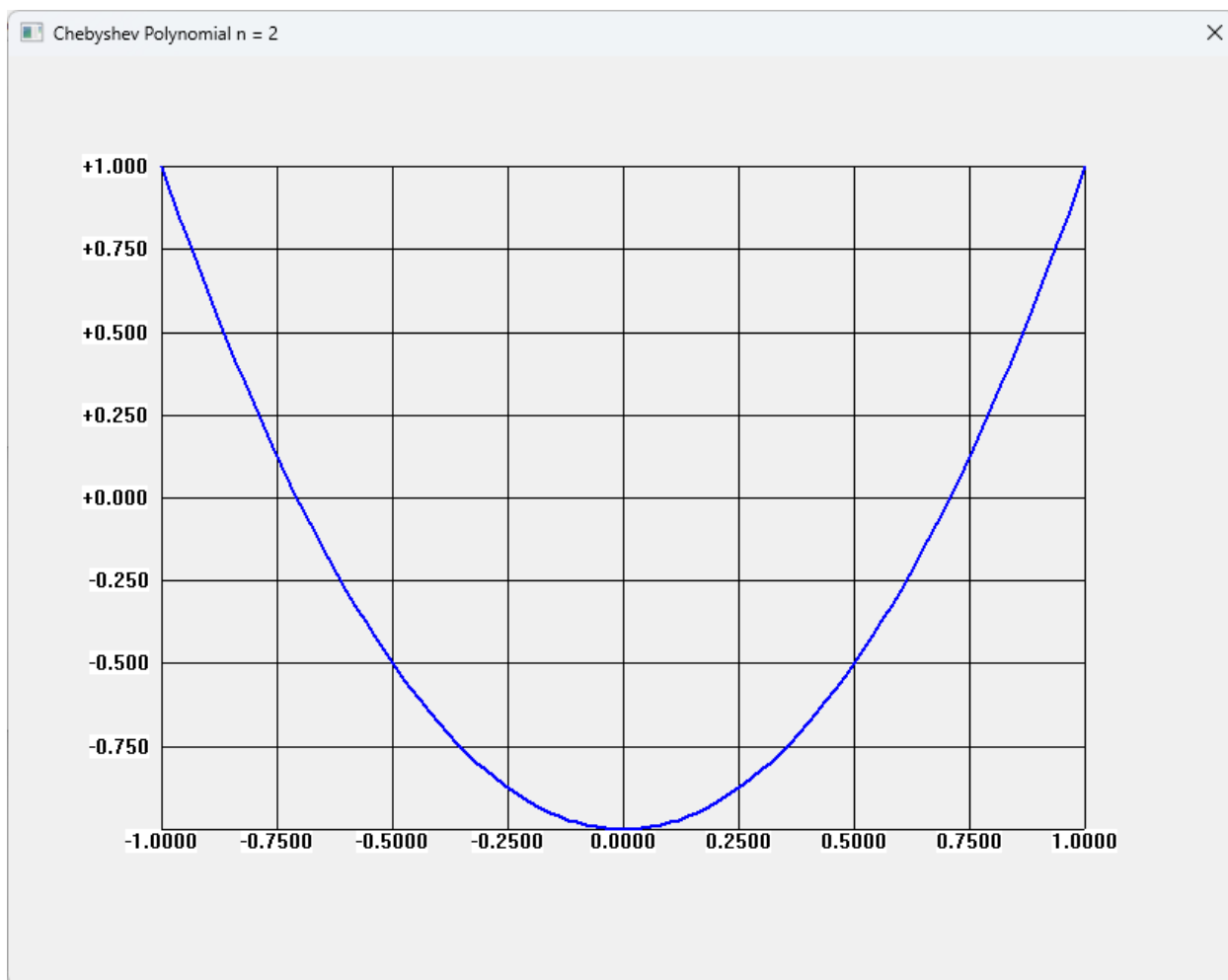
Input Dialog

n

Polynomial Type Choice

- ☒ Chebyshev
- ☐ Hermite
- ☐ Laguerre
- ☐ Legendre

Draw Table OK Cancel



Input Dialog

n

Polynomial Type Choice

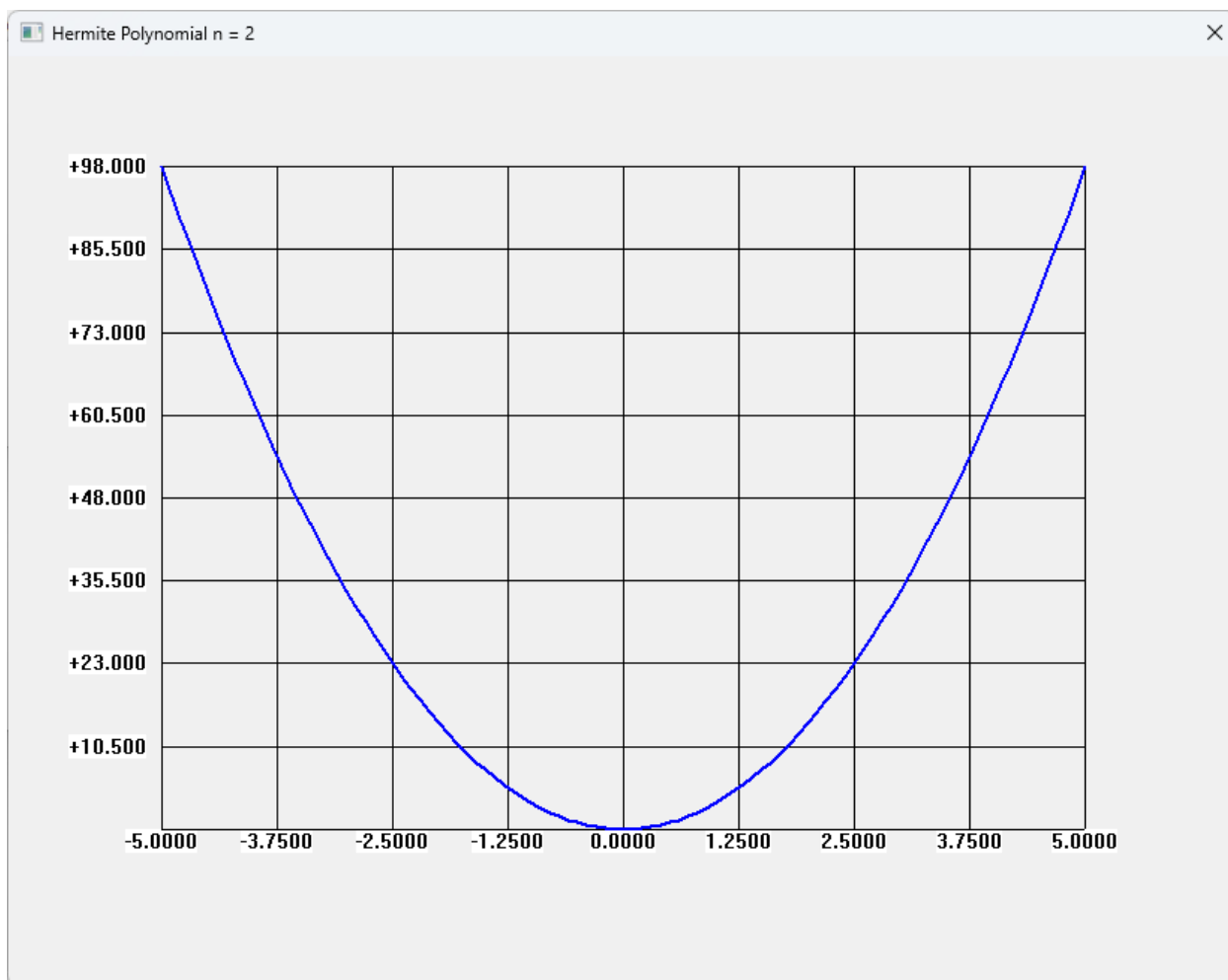
☐ Chebyshev

☒ Hermite

☐ Laguerre

☐ Legendre

Draw Table OK Cancel



Input Dialog

n

Polynomial Type Choice

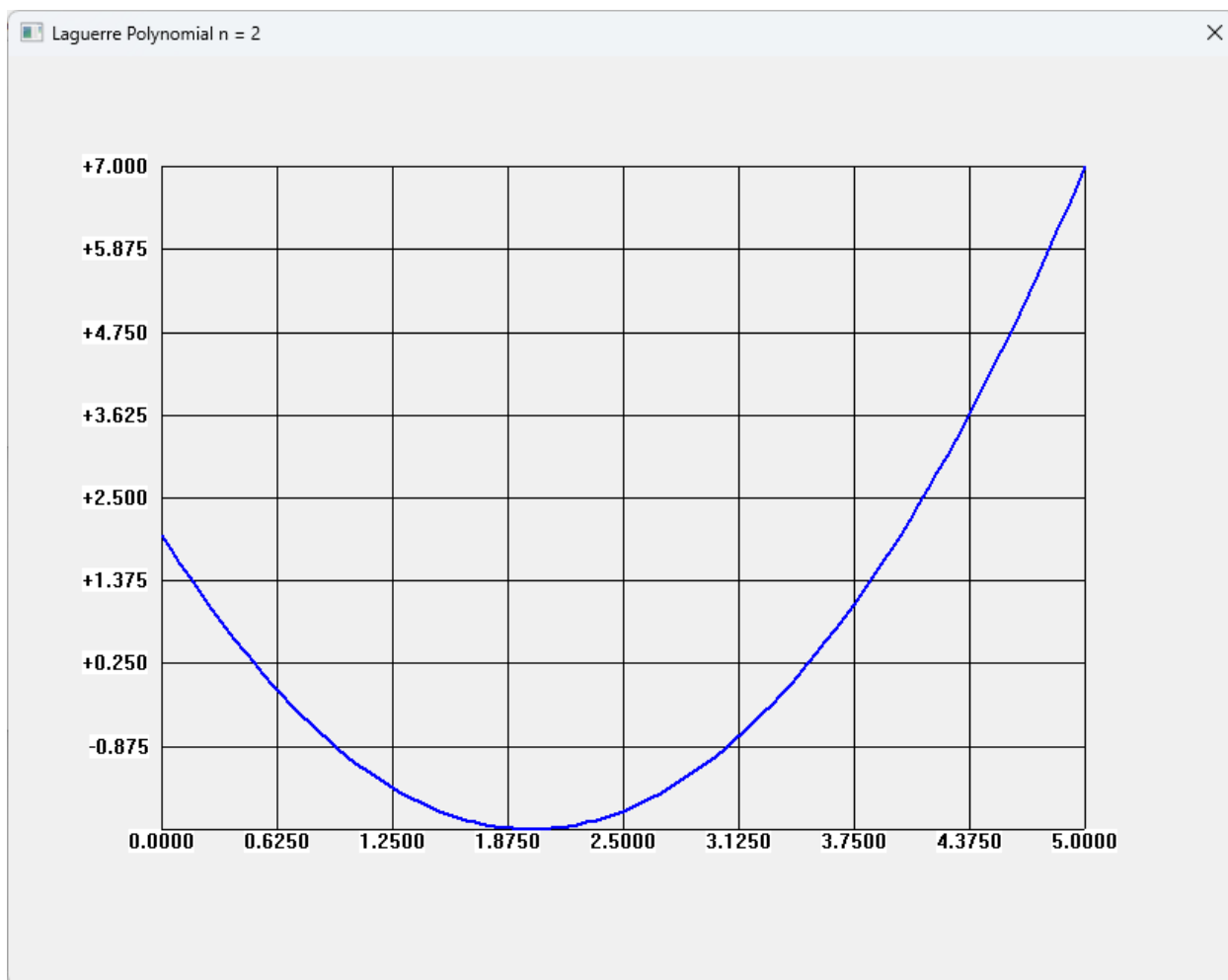
☐ Chebyshev

☐ Hermite

☒ Laguerre

☐ Legendre

Draw Table OK Cancel



Input Dialog

n

Polynomial Type Choice

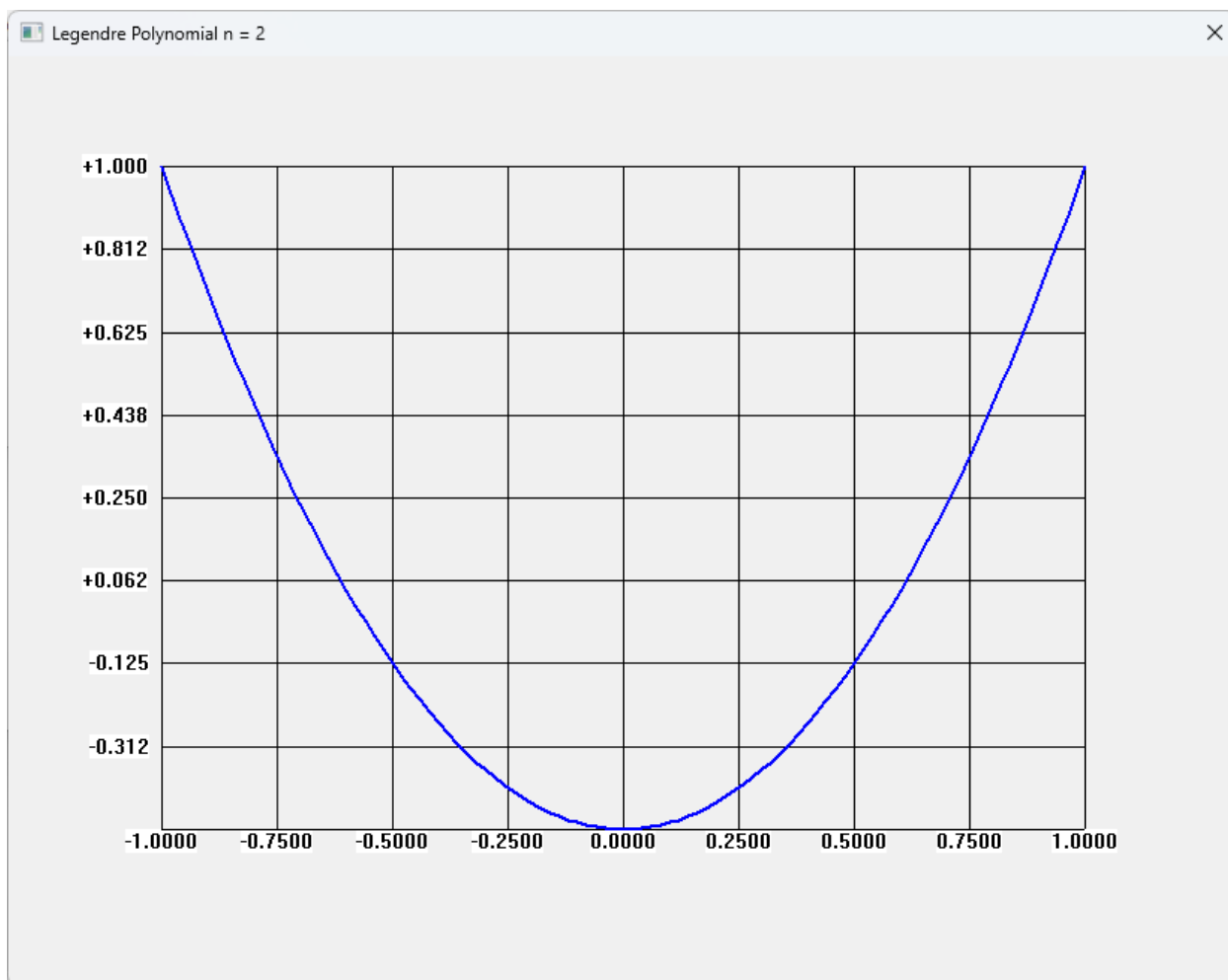
☐ Chebyshev

☐ Hermite

☐ Laguerre

☒ Legendre

Draw Table OK Cancel



Input Dialog

n

Polynomial Type Choice

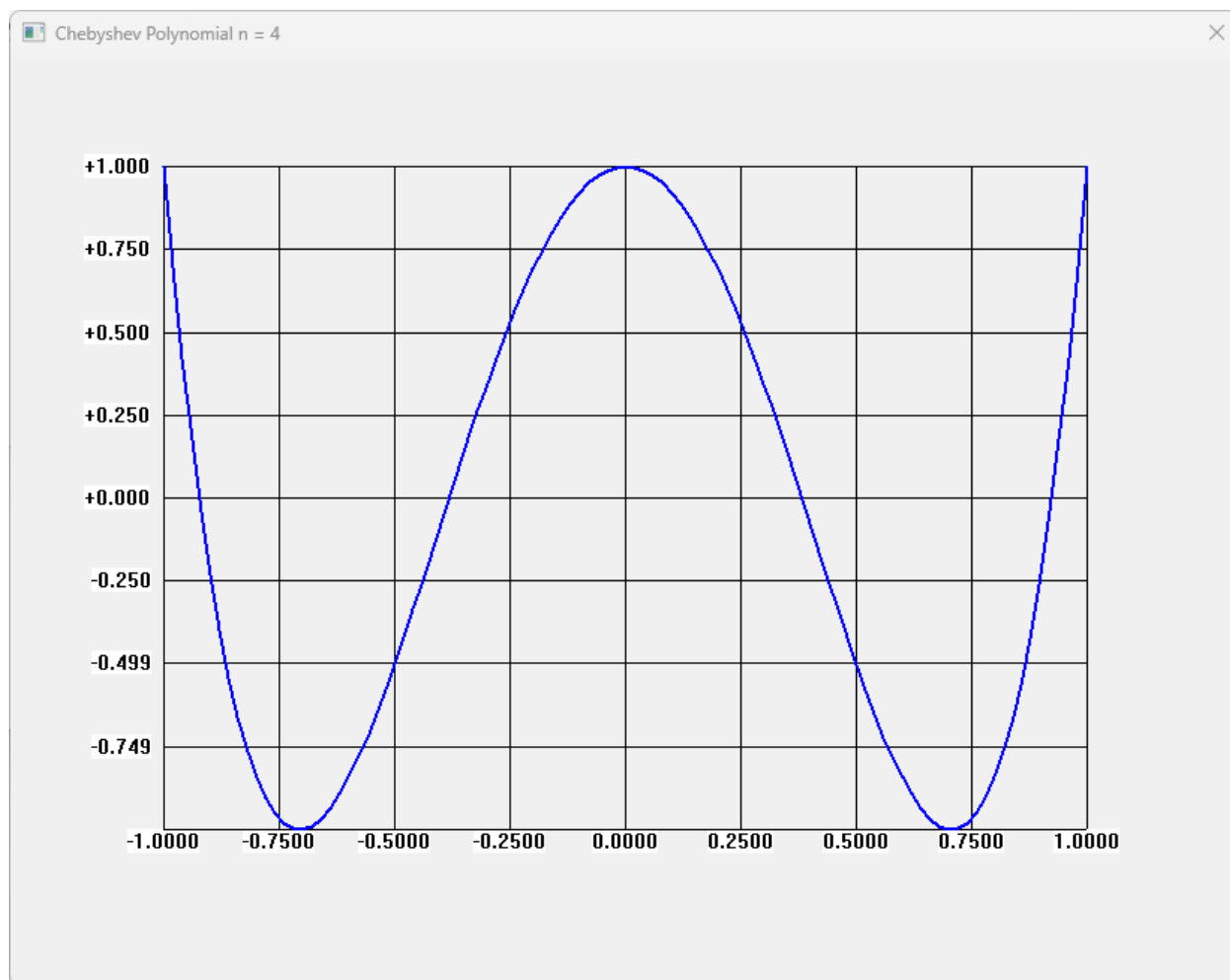
☒ Chebyshev

☐ Hermite

☐ Laguerre

☐ Legendre

Draw Table OK Cancel



Input Dialog

n

Polynomial Type Choice

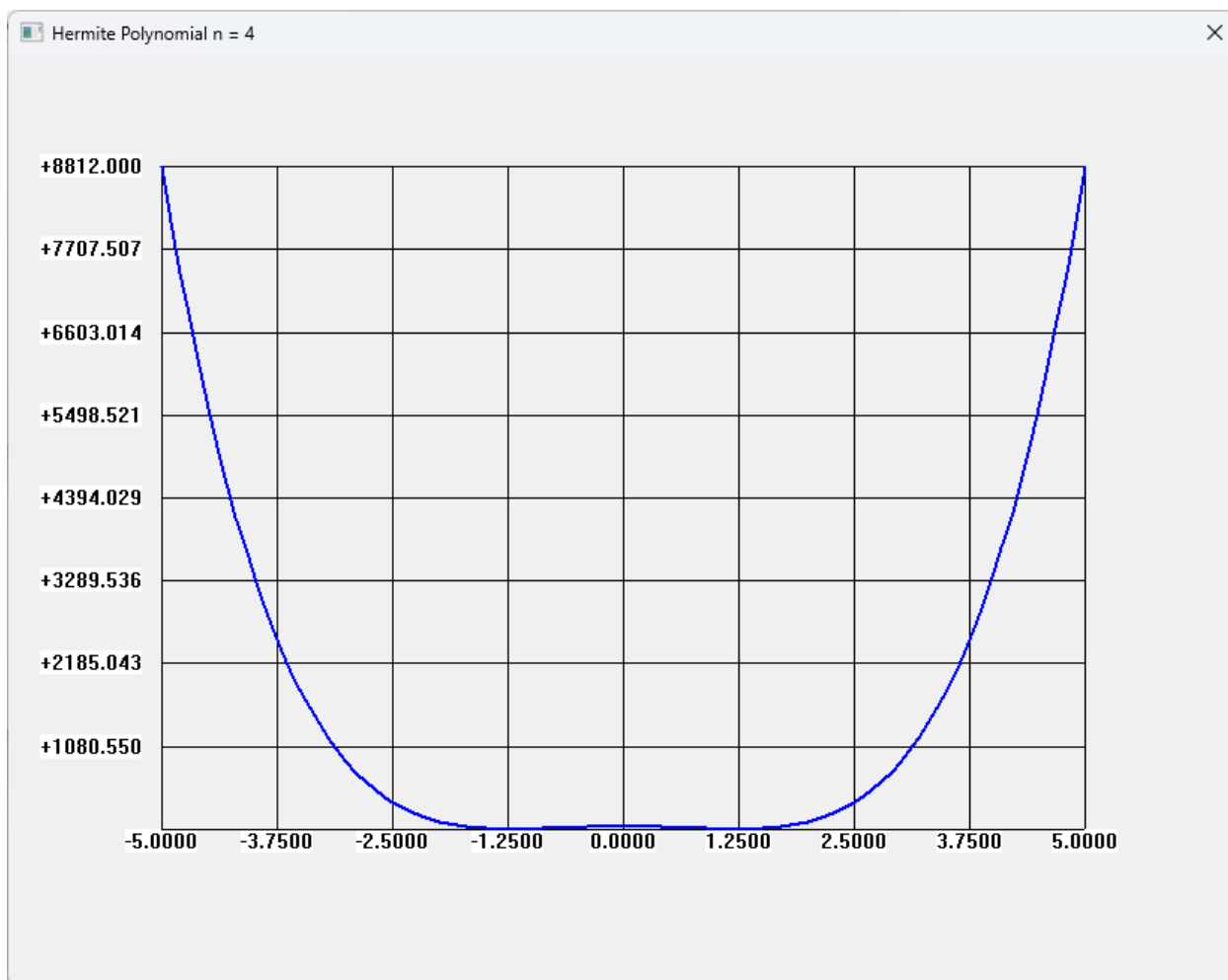
☐ Chebyshev

☒ Hermite

☐ Laguerre

☐ Legendre

Draw Table OK Cancel



Input Dialog

n

Polynomial Type Choice

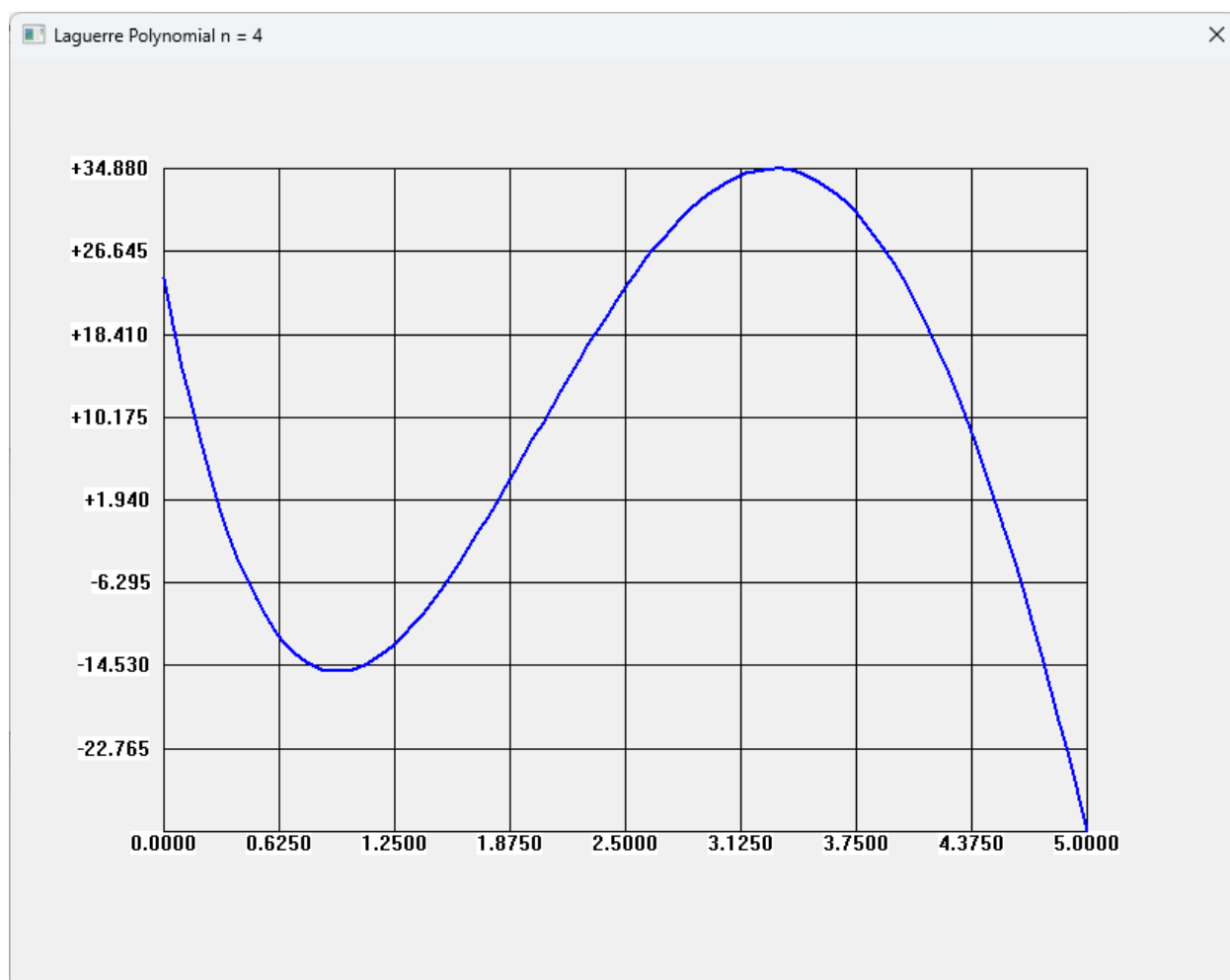
☐ Chebyshev

☐ Hermite

☒ Laguerre

☐ Legendre

Draw Table OK Cancel



Input Dialog

n

Polynomial Type Choice

☐ Chebyshev

☐ Hermite

☐ Laguerre

☒ Legendre

Draw Table OK Cancel

