1.0 Definition of Equation with Exact Solution

Definition of Equation

$$P(t) = a_1 \times t + \sin(2\pi f_1 t)$$

Where,

$$a_1 = \frac{[My\ registration\ number]}{40000} = \frac{126829}{40000} W\ and\ f_1 = 2Hz.$$

From the above equation, the exact solution of the integration process is obtained as:

$$E(t) = \int_{t=0}^{t=20} P(t)dt = \left[\frac{126829}{40000 \times 2} t^2 - \frac{\cos(2\pi f_1 t)}{2\pi} + C\right]_0^{20}$$

Substituting the time limit into the exact solution, we obtain energy as:

E(t) = 634.238J

2.0 Plotting the Function P(t)

The Function P(t) is plotted on the unity engine by using the default C# math class ***"org.mariuszgromada.math.mxparser"*** from .NET Framework and Function Plotter Class. The function plot can be seen in Figure 1 with P(t) on Y-axis and t on x-axis.

It should be noted that to plot this point, the variable ***"numberOfFunctionPoints"*** in the C# class was modified to 30.

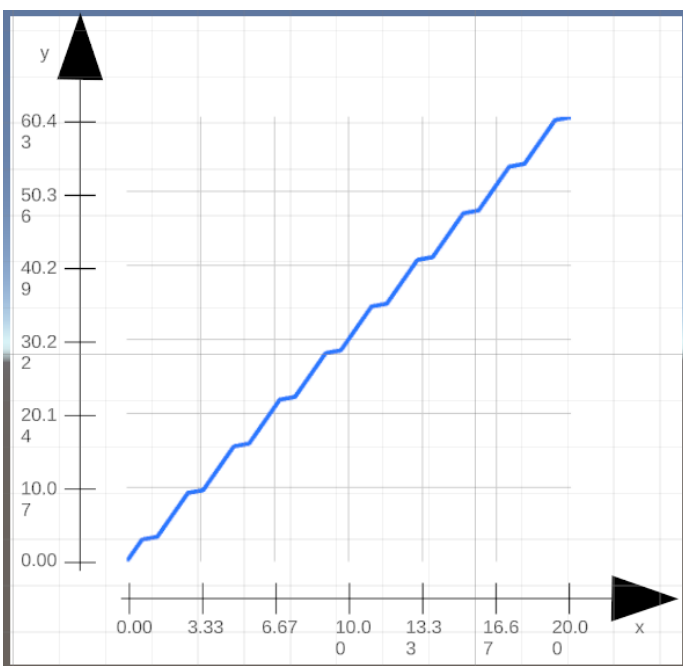

Figure 1: Plot of Function P(t) for 0<= t <= 20

3.0 UML Diagram

The UML diagram used to implement the class Numerical Integral can be seen in figure 2 below.

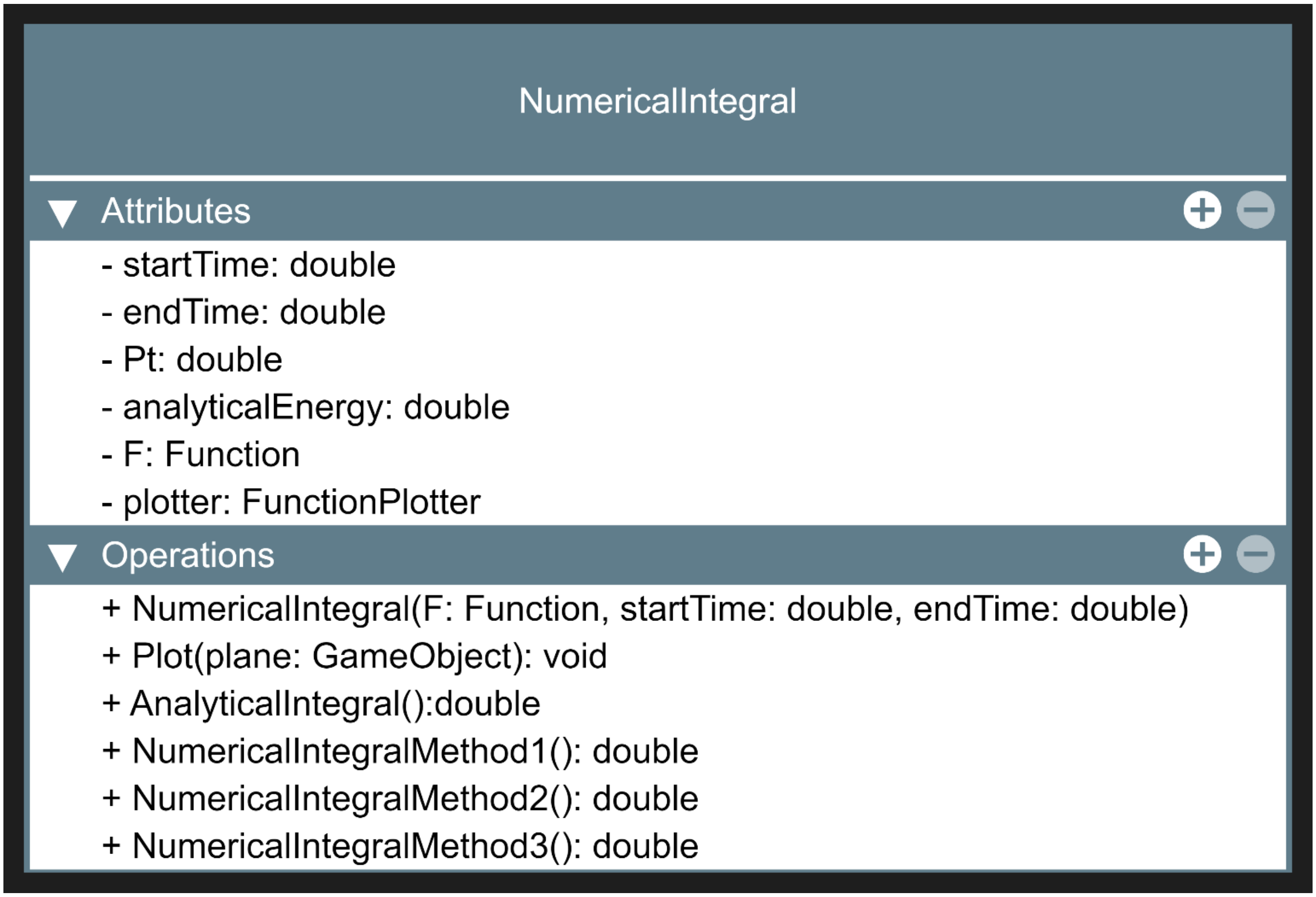

Figure 2: UML Diagram for Implementing the Numerical Integral.

From the UML diagram above, the C# file where the methods are implemented is name **NumericalIntegral,** while the test file is named **PowerCalculations**.

4.0 Nassi-Schneiderman Diagrams

The Nassi-Schneiderman diagrams used in implementing the numerical integral using the three methods are shown below, and a extra one used to obtain the result of the analytical integral is also shown.

Figure 3 shows the Nassi-Schneiderman Diagram for Numerical Integral Method 1.

Figure 4 shows the Nassi-Schneiderman Diagram for Numerical Integral Method 2.

Figure 5 shows the Nassi-Schneiderman Diagram for Numerical Integral Method 3.

Figure 6 shows the Nassi-Schneiderman Diagram for calculating the exact analytical solution.

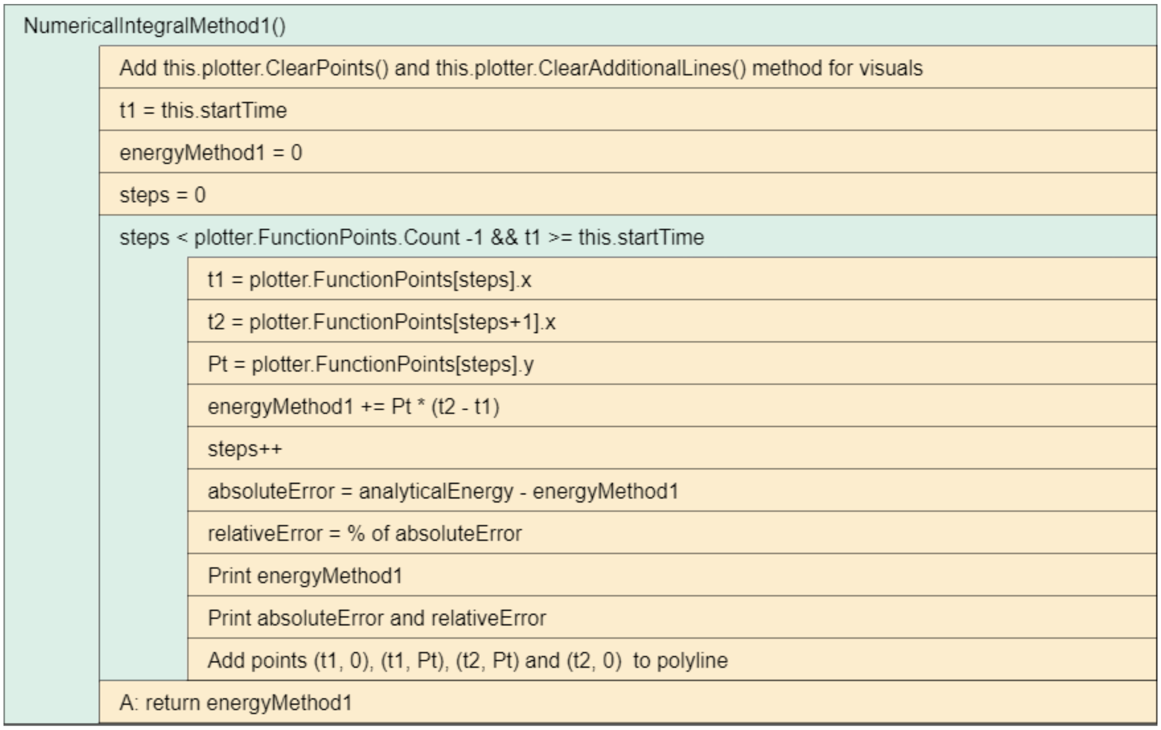

Figure 3: Nassi-Schneiderman's Diagram for Numerical Integral using Method 1

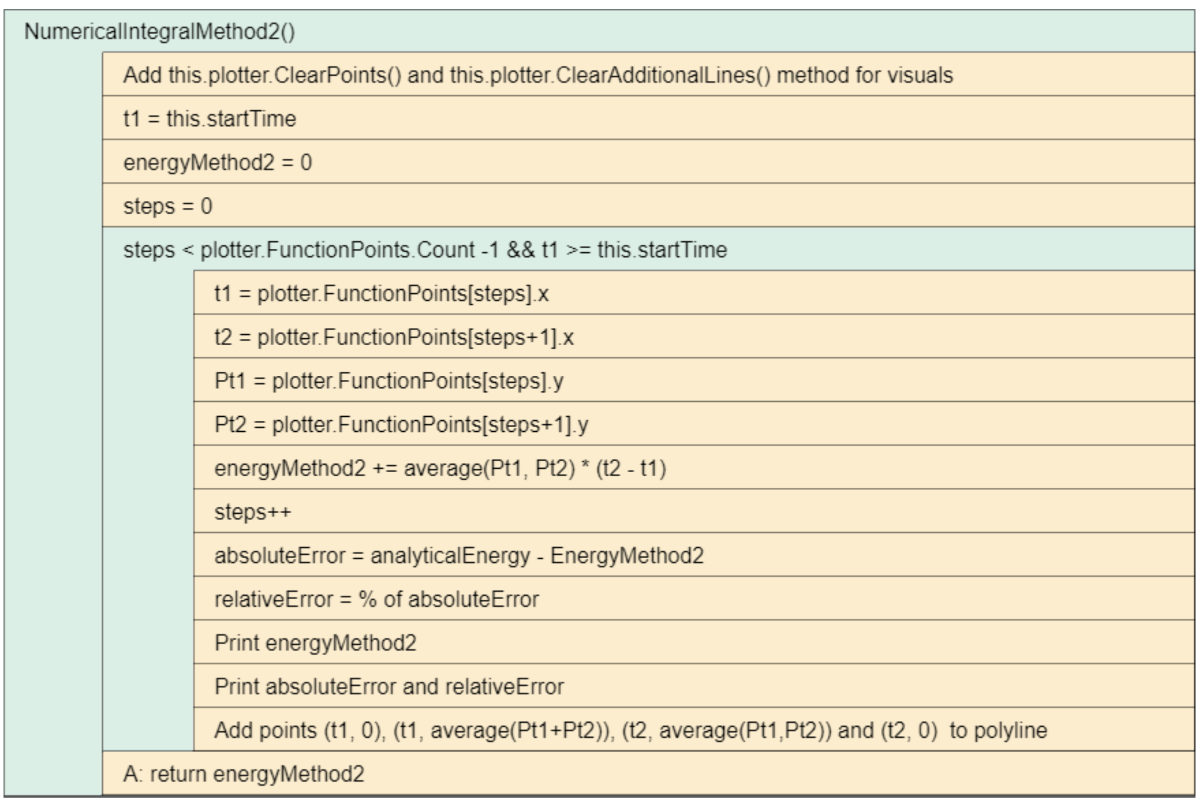

Figure 4: Nassi-Schneiderman's Diagram for Numerical Integral using Method 2

```
NumericalIntegralMethod3()
    Add this.plotter.ClearPoints() and this.plotter.ClearAdditionalLines() method for visuals
    t1 = this.startTime
    energyMethod3 = 0
    steps = 0
    steps < plotter.FunctionPoints.Count -1 && t1 >= this.startTime
        t1 = plotter.FunctionPoints[steps].x
        t2 = plotter.FunctionPoints[steps+1].x
        Pt1 = plotter.FunctionPoints[steps].y
        Pt2 = plotter.FunctionPoints[steps+1].y
        energyMethod3 += (Pt1+Pt2)/2 * (t2 - t1) //Area of a trapezium
        steps++
        absoluteError = analyticalEnergy - EnergyMethod3
        relativeError = % of absoluteError
        Print energyMethod3
        Print absoluteError and relativeError
        Add points (t1, 0), (t1, (Pt1+Pt2)/2), (t2, 0) and (t2, (Pt1+Pt2)/2)  to polyline
    A: return energyMethod3
```

Figure 5: Nassi-Schneiderman's Diagram for Numerical Integral using Method 3

```
AnalyticalIntegral()
    Function E = new Function("E(t) = (126829*t^2/(40000*2)) - (cos(2*pi*2*t)/(pi*2))")
    E1 = E.calculate(this.startTime)
    E2 = E.calculate(this.endTime)
    analyticalEnergy = E2 -E1
    print analyticalEnergy
    A: return analyticalEnergy
```

Figure 6: Nassi-Schneiderman Diagram for Analytical Integral

5.0 Results of Numerical Integral

The solution of the numerical integration are shown below. This includes the plots/visualization indicating the method, the result, absolute and relative error.

- **Numerical Integral Result Method 1:**
  Figure 7 shows the graphical visualization of the numerical integral using method 1, and Figure 8 shows the result:

  Numerical Integral E(t) = 572.72J

  Absolute Error = 61.428J

  Relative Error = 9.69%

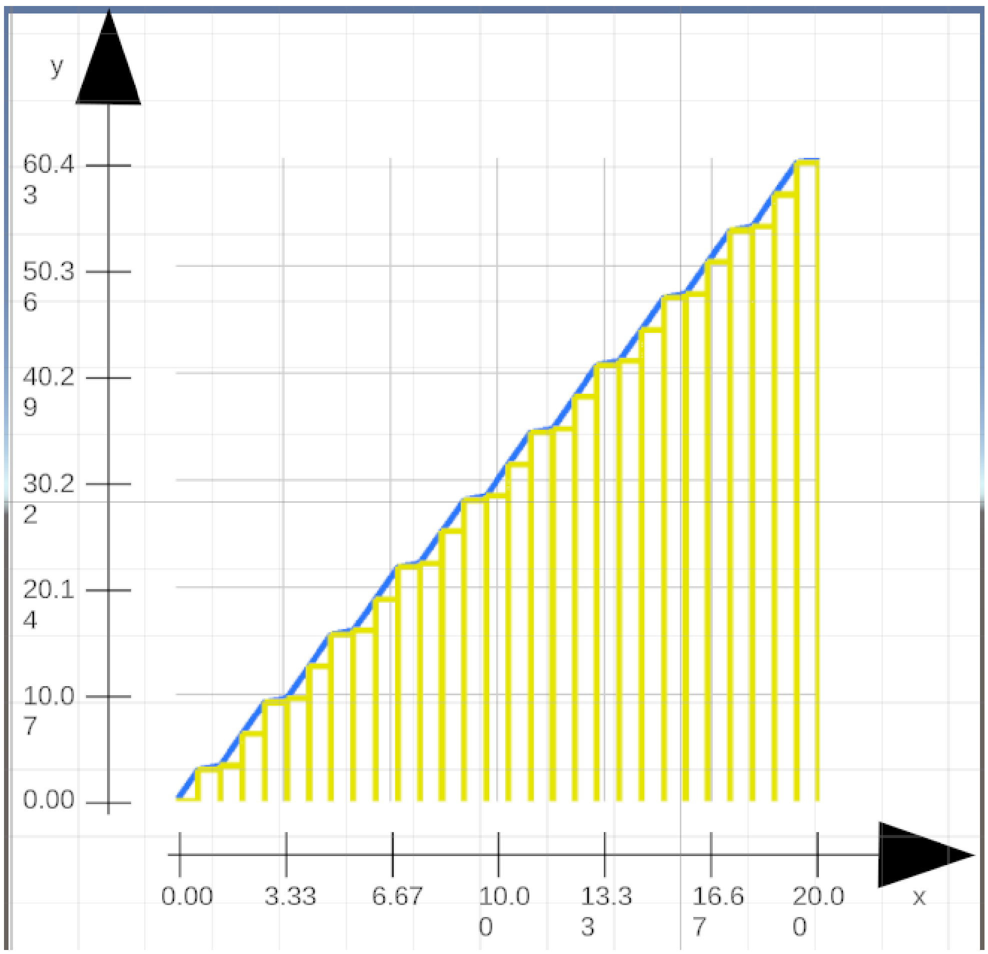

Figure 7: Plot of Numerical Integral using Method 1

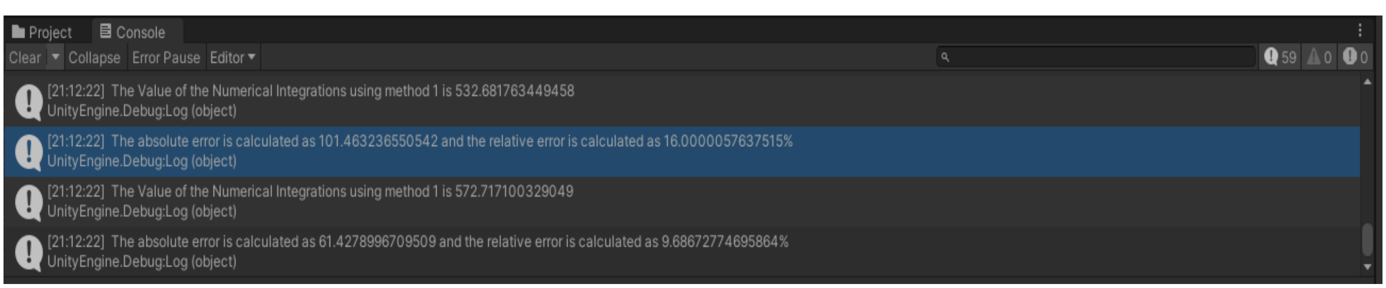

Figure 8: Image showing result of Numerical Integral Method 1, Absolute Error, and Relative Error

- **Numerical Integral Result Method 2:**
  Figure 9 shows the graphical visualization of the numerical integral using method 2, and Figure 10 shows the result:

  Numerical Integral E(t) = 592.862J

  Absolute Error = 41.283J

  Relative Error = 6.51%

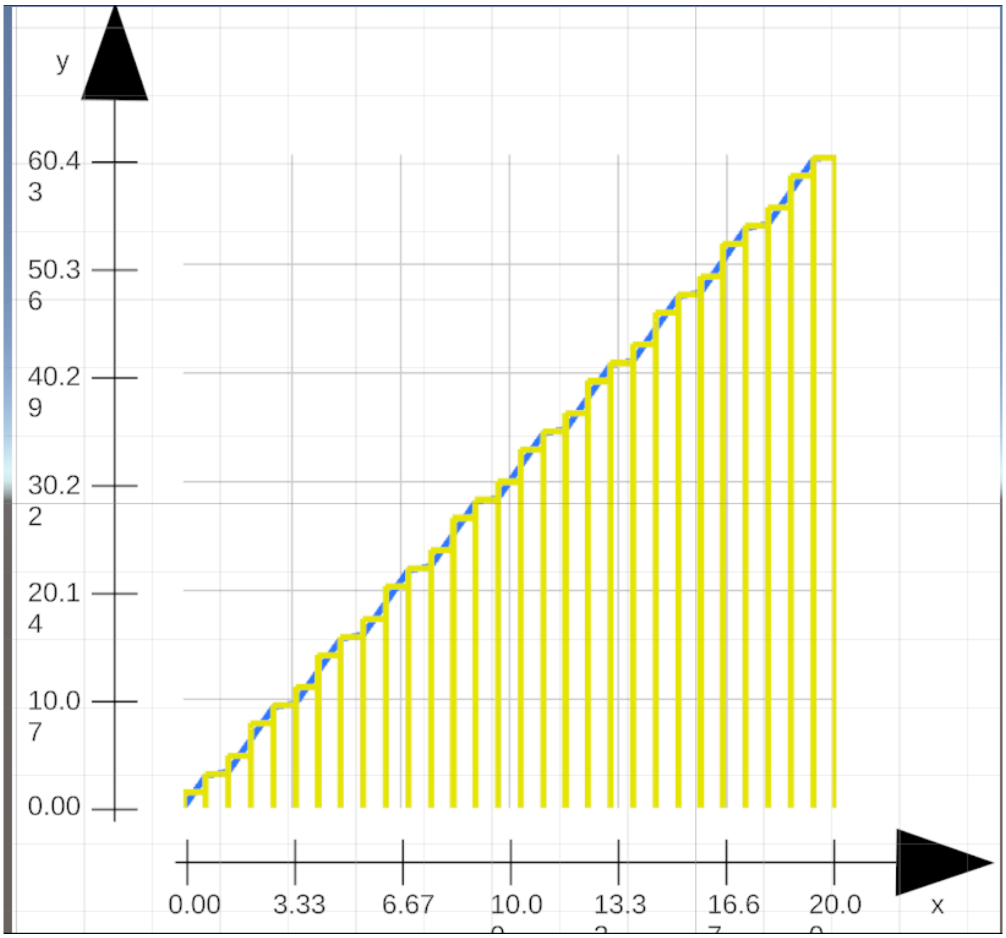

Figure 9: Plot of Numerical Integral using Method 2

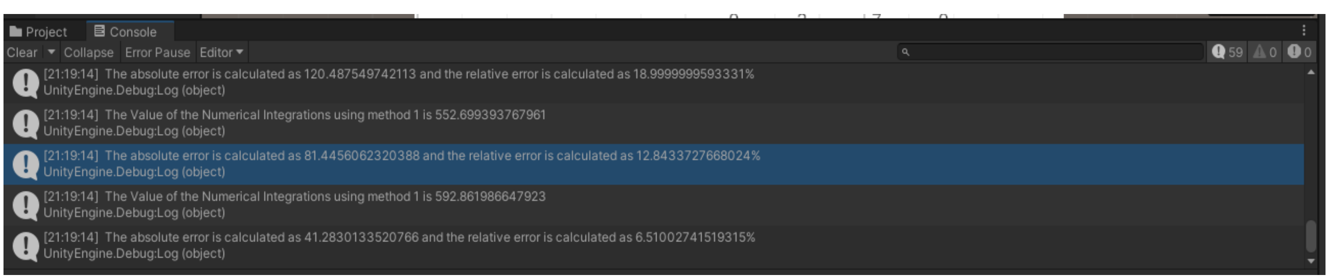

Figure 10: Image showing result of Numerical Integral Method 2, Absolute Error, and Relative Error

- **Numerical Integral Result Method 3:**
  Figure 11 shows the graphical visualization of the numerical integral using method 3, and Figure 12 shows the result:

  Numerical Integral E(t) = 592.862J

  Absolute Error = 41.283J

  Relative Error = 6.51%

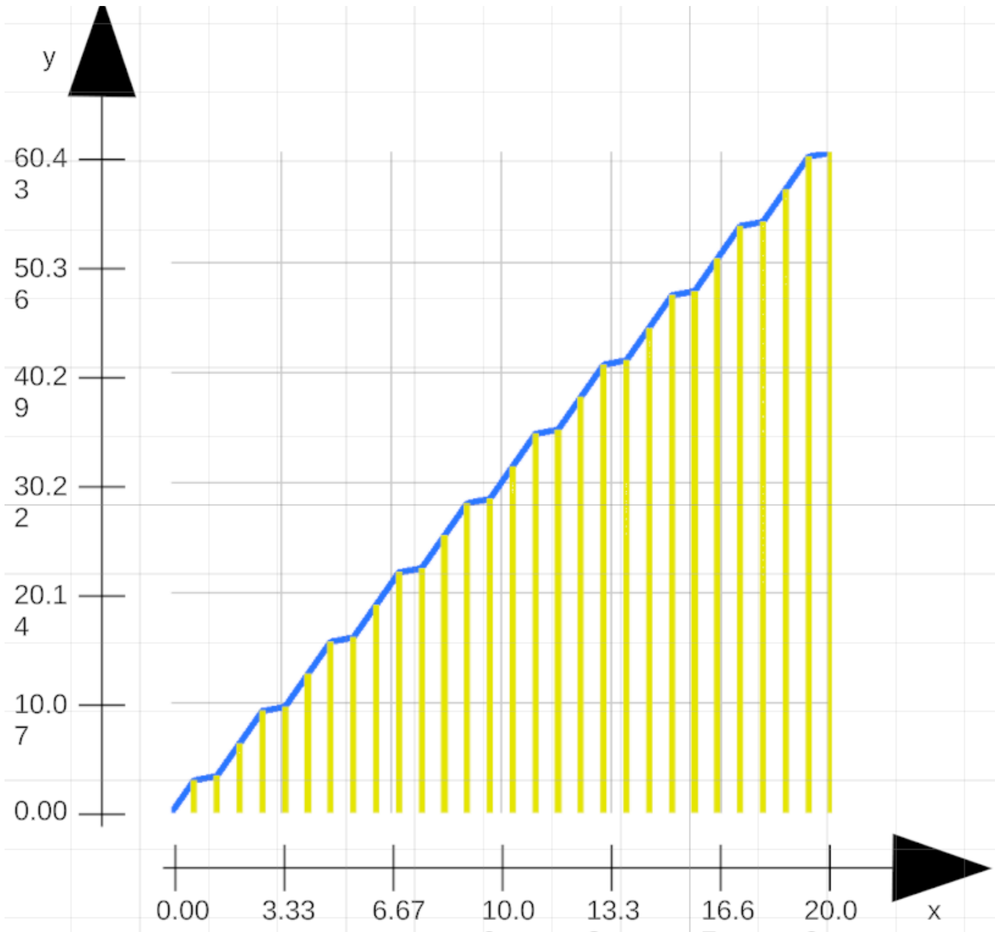

Figure 11: Plot of Numerical Integral using Method 3

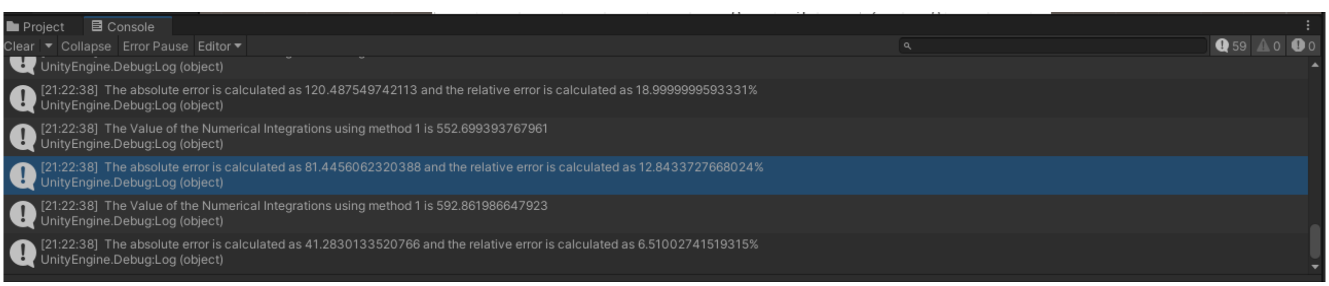

Figure 12: Image showing result of Numerical Integral Method 3, Absolute Error, and Relative Error