Bisection Method

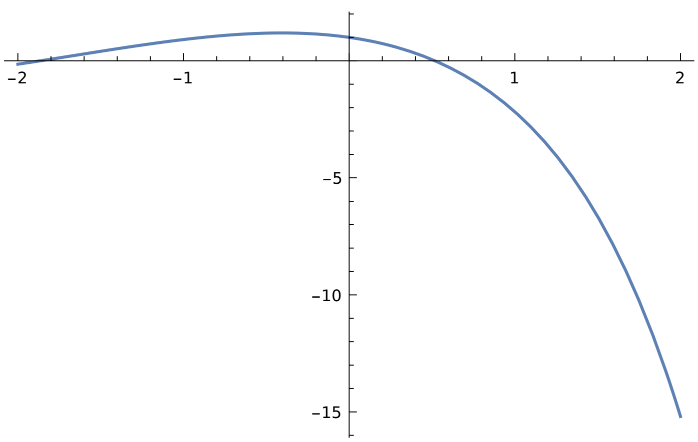
Q1 = Perform five iteration and find root of f(x)= cos(x)-xe^x

In[1]:=

```
f[x_] = Cos[x] - x * E ^ x
Plot[f[x], {x, -2, 2}]
a = 0;
b = 2;
If[f[a] * f[b] > 0, Print["Bisection  method can not be applied"]]
m = (a + b) / 2.0
For[i = 1, i ≤ 5, i++,
 {If[f[a] * f[m] < 0, {a = a, b = m}, {a = m, b = b}],
  Print[i, "   ", a, "   ", b];
  m = (a + b) / 2.0}]
Print["root=", m]
```

Out[1]= $-e^{x}\,x + \mathrm{Cos}[x]$

Out[2]=



Out[6]= 1.

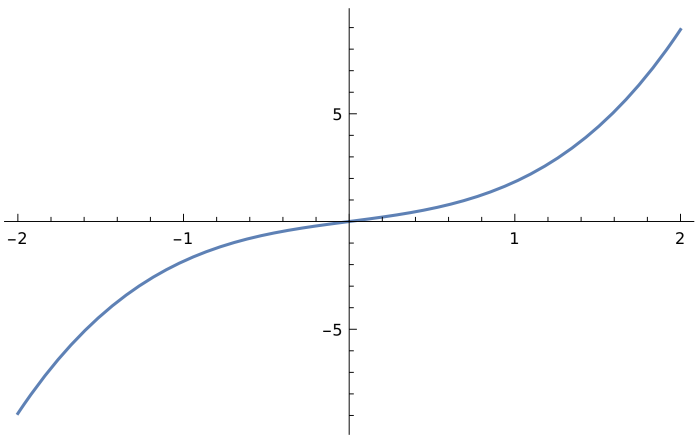
```
1    0    1.
2    0.5    1.
3    0.5    0.75
4    0.5    0.625
5    0.5    0.5625
root=0.53125
```

Q2 = Find root of f(x)= x^3 +Sin(x)

In[63]:=
```
f[x_] = x^3 + Sin[x]
Plot[f[x], {x, -2, 2}]
a = -1;
b = 3;
If[f[a] * f[b] > 0, Print["Bisection  method can not be applied"]]
m = (a + b) / 2.0
For[i = 1, i ≤ 10, i++,
 {If[f[a] * f[m] < 0, {a = a, b = m}, {a = m, b = b}],
  Print[i, " ", a, " ", b];
  m = (a + b) / 2.0}]
Print["root=", m
```

Out[63]= x^3 + Sin[x]

Out[64]=



Out[68]= 1.

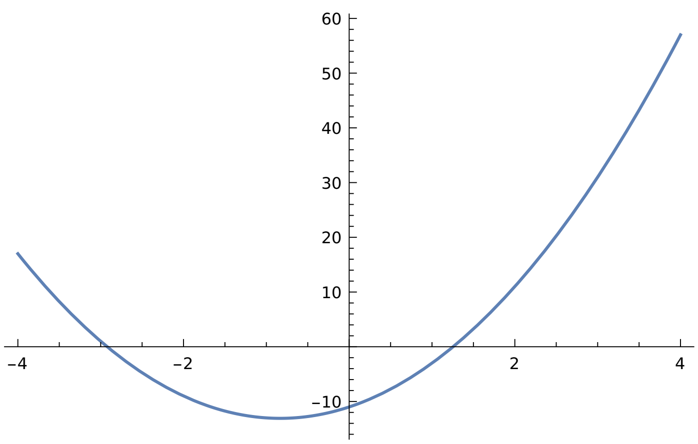
```
1 -1 1.
2 0.  1.
3 0.5  1.
4 0.75  1.
5 0.875  1.
6 0.9375  1.
7 0.96875  1.
8 0.984375  1.
9 0.992188  1.
10  0.996094  1.
```

Q3 = Perform five iteration and find root of f(x)= $3x^2 + 5x - 11$

In[70]:=
```
f[x_] = 3*x^2 + 5*x - 11
Plot[f[x], {x, -4, 4}]
a = 2;
b = 3;
If[f[a]*f[b] > 0, Print["Bisection  method can not be applied"]]
m = (a + b) / 2.0
For[i = 1, i ≤ 6, i++,
 {If[f[a]*f[m] < 0, {a = a, b = m}, {a = m, b = b}],
  Print[i, " ", a, " ", b];
  m = (a + b) / 2.0}]
Print["root=", m]
```

Out[70]= $-11 + 5x + 3x^2$

Out[71]=




```
Bisection  method can not be applied
```

Out[75]= 2.5

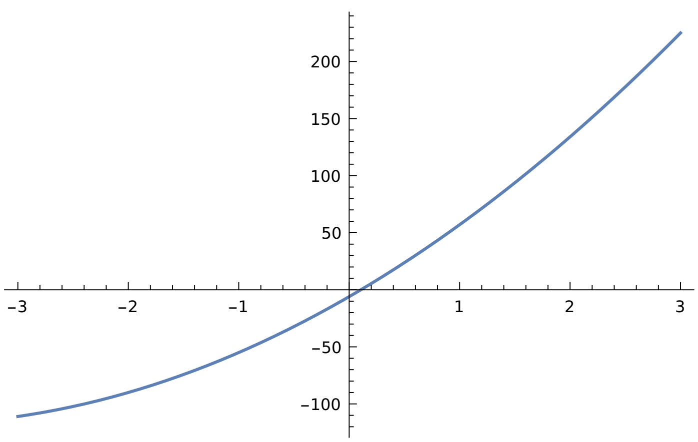
```
1 2.5 3
2 2.75 3
3 2.875 3
4 2.9375 3
5 2.96875 3
6 2.98438 3
root=2.99219
```

Secant Method

Q1= Find root of f(x) = $7 x^2 + 56 x - 6$ **using secant method upto 3 decimal**

In[ ]:=
```
secant[f_, x0_, x1_, n_] := Module[{}, p0 = N[x0]; p1 = N[x1];
  If[f[p0] * f[p1] > 0, Print["secant method can not be applied"];
   Return[]];
  i = 1;
  While[i ≤ n,
   p2 = N[(p0 * f[p1] - p1 * f[p0]) / (f[p1] - f[p0])];
   Print[i, " ", p0, " ", p1];
   i++;
   p0 = p1;
   p1 = p2];
  Print["root=", p2]]
f[x_] := 7 x^2 + 56 x - 6;
Plot[f[x], {x, -3, 3}]
secant[f, -1, 1, 7]
```

Out[ ]=



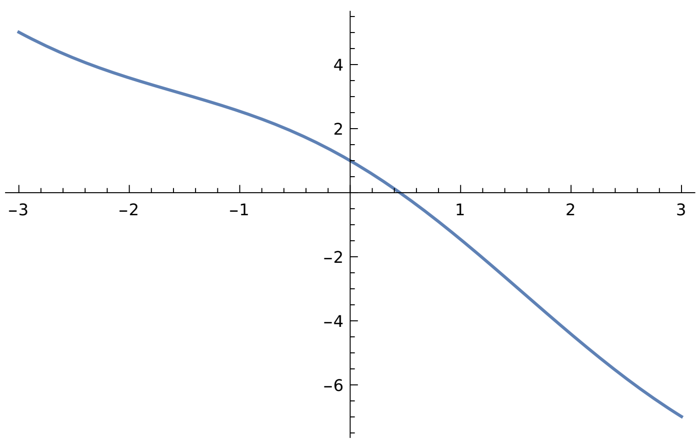

```
1 -1. 1.
2 1. -0.0178571
3 -0.0178571  0.0934394
4 0.0934394  0.105933
5 0.105933  0.105745
6 0.105745  0.105745
7 0.105745  0.105745
root=0.105745
```

Q2= Find root of f(x) = Cos(x)-2x using secant method by performing 5 iteration

In[ ]:=
```
secant[f_, x0_, x1_, n_] := Module[{}, p0 = N[x0]; p1 = N[x1];
  If[f[p0] * f[p1] > 0, Print["secant method can not be applied"];
   Return[]];
  i = 1;
  While[i ≤ n,
   p2 = N[(p0 * f[p1] - p1 * f[p0]) / (f[p1] - f[p0])];
   Print[i, " ", p0, " ", p1];
   i++;
   p0 = p1;
   p1 = p2];
  Print["root=", p2]]
f[x_] = Cos[x] - 2 x
Plot[f[x], {x, -3, 3}]
secant[f, -0, 1, 5]
```

Out[ ]= $-2x + \text{Cos}[x]$

Out[ ]=



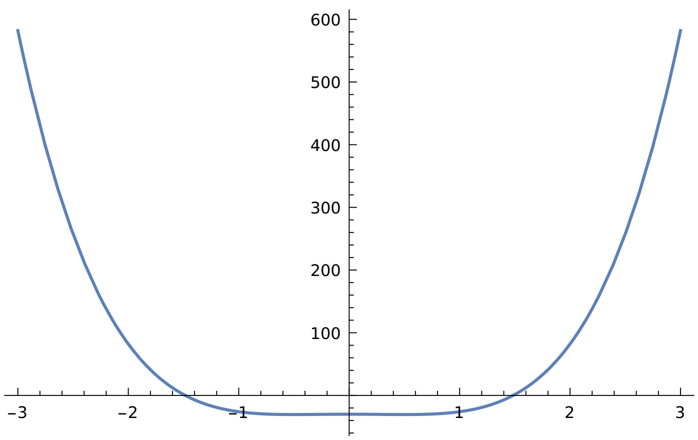

```
1 0. 1.
2 1. 0.406554
3 0.406554  0.446512
4 0.446512  0.450214
5 0.450214  0.450184
root=0.450184
```

Q3= Find root of f(x) =8 $x^4 - 4x^2$-30 using secant method

In[ ]:=
```
secant[f_, x0_, x1_, n_] := Module[{}, p0 = N[x0]; p1 = N[x1];
  If[f[p0] * f[p1] > 0, Print["secant method can not be applied"];
   Return[]];
  i = 1;
  While[i ≤ n,
   p2 = N[(p0 * f[p1] - p1 * f[p0]) / (f[p1] - f[p0])];
   Print[i, " ", p0, " ", p1];
   i++;
   p0 = p1;
   p1 = p2];
  Print["root=", p2]]
f[x_] = 8 x^4 - 4 x^2 - 30
Plot[f[x], {x, -3, 3}]
secant[f, 2, 3, 5]
```

Out[ ]= $-30 - 4 x^2 + 8 x^4$

Out[ ]=



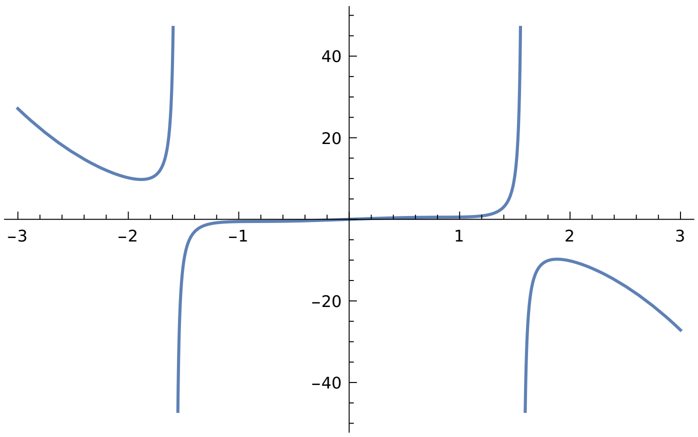

```
secant method can not be applied
```

Regula Falsi Method

Q1= Find root of f(x) = Tan $(x) - x^3$ using regula falsi

In[ ]:=
```
f[x_] = Tan[x] - x ^ 3;
Plot[f[x], {x, -3, 3}]
regulafalsi[a0_, b0_, m_] := Module[{}, a = N[a0]; b = N[b0];
   c = (a * f[b] - b * f[a]) / (f[b] - f[a]); k = 0;
   While[k < m, If[Sign[f[b]] == Sign[f[c]], b = c, a = c;];
    c = (a * f[b] - b * f[a]) / (f[b] - f[a]); k = k + 1;];
   Print["c= ", NumberForm[c, 5]];
   Print["f[c]= ", NumberForm[f[c], 5]];]
regulafalsi[-1, -2, 15]
```

Out[ ]=



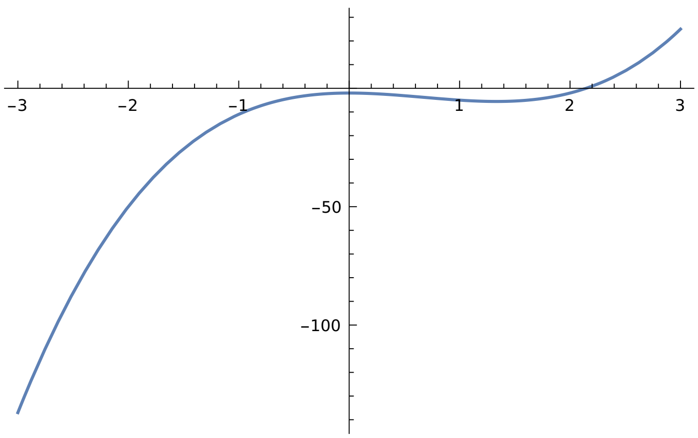

```
c= -1.5681

f[c]= -373.04
```

Q2= Find root of f(x) = 3 x^3 – 6 x^2-2 using regula falsi

In[ ]:=
```
f[x_] = 3 x^3 - 6 x^2 - 2
Plot[f[x], {x, -3, 3}]
regulafalsi[a0_, b0_, m_] := Module[{}, a = N[a0]; b = N[b0];
   c = (a * f[b] - b * f[a]) / (f[b] - f[a]); k = 0;
   While[k < m, If[Sign[f[b]] == Sign[f[c]], b = c, a = c;];
    c = (a * f[b] - b * f[a]) / (f[b] - f[a]); k = k + 1;];
   Print["c= ", NumberForm[c, 5]];
   Print["f[c]= ", NumberForm[f[c], 5]];]
regulafalsi[1, 2, 7]
```

Out[ ]= $-2 - 6 x^2 + 3 x^3$

Out[ ]=

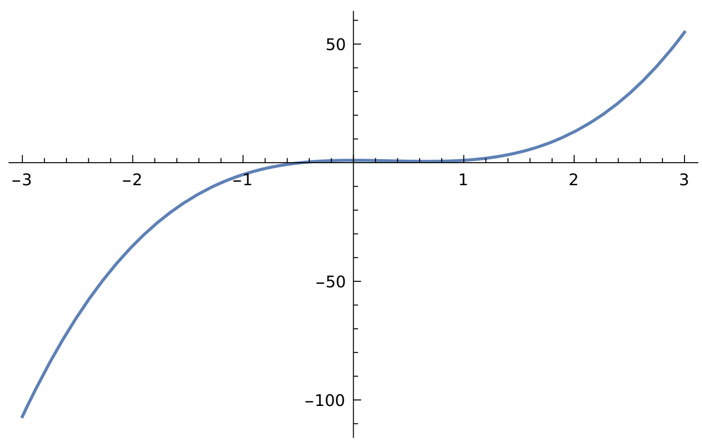



```
c= 2.1448
f[c]= -0.0010938
```

Q3= Find root of f(x) = $3x^3 - 3x^2 + 1$ using regula falsi

In[ ]:=
```
f[x_] = 3 x^3 - 3 x^2 + 1
Plot[f[x], {x, -3, 3}]
regulafalsi[a0_, b0_, m_] := Module[{}, a = N[a0]; b = N[b0];
   c = (a * f[b] - b * f[a]) / (f[b] - f[a]); k = 0;
   While[k < m, If[Sign[f[b]] == Sign[f[c]], b = c, a = c;];
    c = (a * f[b] - b * f[a]) / (f[b] - f[a]); k = k + 1;];
   Print["c= ", NumberForm[c, 5]];
   Print["f[c]= ", NumberForm[f[c], 5]];]
regulafalsi[1, 1.5, 5]
```

Out[ ]= $1 - 3 x^2 + 3 x^3$

Out[ ]=


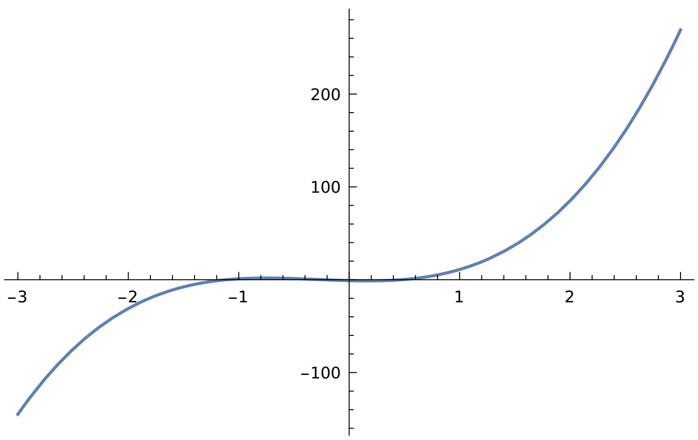

```
c= -0.14269
f[c]= 0.9302
```

Newton Raphson Method

Q1= find root of f(x) =8 x^3 + 7 x^2-3x-1 using newton rap. method by taking x0 = 1.5

In[ ]:=
```
nr[f_, x0_, n_] := Module[{}, a = N[x0];
  i = 1;
  df[x_] = D[f[x], x];
  While[i ≤ n,
   b = N[a] - N[f[a] / N[df[a]]];
   Print[i, "     ", b];
   i++;
   a = b];
  Print["Root =", b]]
f[x_] := 8 x^3 + 7 x^2 - 3 * x - 1;
Plot[f[x], {x, -3, 3}]
nr[f, 1.5, 5]
```

Out[ ]=

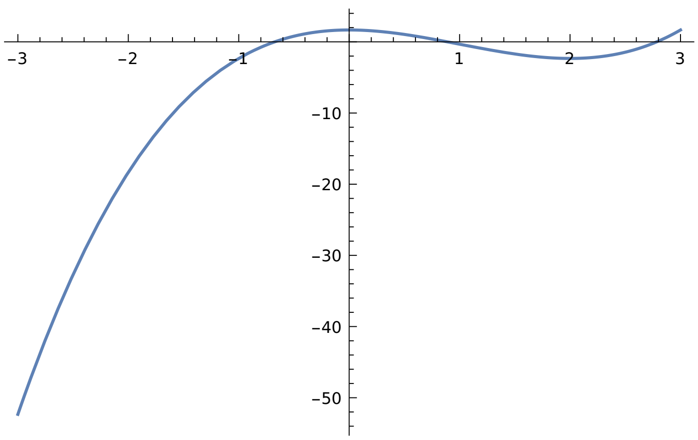



```
1     0.982639
2     0.676084
3     0.52446
4     0.478216
5     0.473741
Root =0.473741
```

Q2= find root of f(x) = $x^3 - 3\,x^2 + 5/3$ using newton rap. method by taking x0 = 0.5

In[ ]:=
```
nr[f_, x0_, n_] := Module[{}, a = N[x0];
  i = 1;
  df[x_] = D[f[x], x];
  While[i ≤ n,
   b = N[a] - N[f[a] / N[df[a]]];
   Print[i, "     ", b];
   i++;
   a = b];
  Print["Root =", b]]
f[x_] := x^3 - 3 x^2 + 5 / 3;
Plot[f[x], {x, -3, 3}]
nr[f, 0.5, 3]
```

Out[ ]=

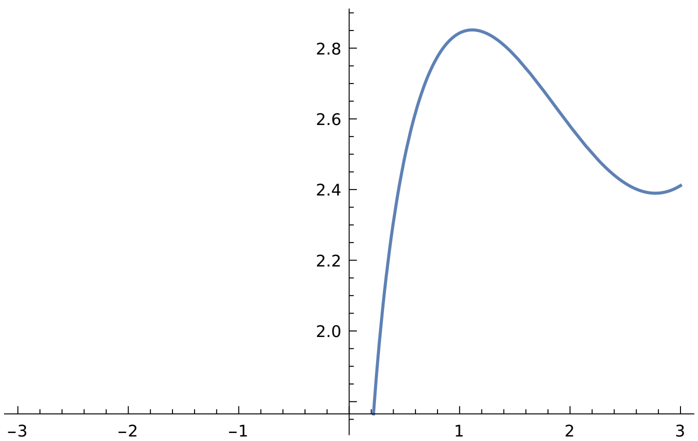



```
1     0.962963
2     0.88877
3     0.888426
Root =0.888426
```

Q3= find root of f(x) = log(10x)+cos(x) using newton rap. method by taking x0 = 1.3

In[ ]:=
```
nr[f_, x0_, n_] := Module[{}, a = N[x0];
  i = 1;
  df[x_] = D[f[x], x];
  While[i ≤ n,
   b = N[a] - N[f[a] / N[df[a]]];
   Print[i, "     ", b];
   i++;
   a = b];
  Print["Root =", b]]
f[x_] := Log[10 x] + Cos[x];
Plot[f[x], {x, -3, 3}]
nr[f, 1.3, 3]
```

Out[ ]=




```
1     15.8756
2     -1.87802
3     -8.13041 - 7.46752 i
Root =-8.13041 - 7.46752 i
```

Gauss Elimination Method

Q1= Solve system of linear equation usning gauss elim.

2y+8z = 1;

3x+4y+7z = 2;

5x+2y+z = -1;

In[ ]:=
```
m = {{0, 2, 8}, {3, 4, 7}, {5, 2, 1}}; x = {{x1}, {x2}, {x3}}; b = {{1}, {2}, {-1}}; m.x == b
ArrayFlatten[{{m, b}}] // MatrixForm
RowReduce[%] // MatrixForm
LinearSolve[m, b]
```

$$m = \begin{pmatrix} 0 & 2 & 8 \\ 3 & 4 & 7 \\ 5 & 2 & 1 \end{pmatrix};\; x = \begin{pmatrix} x1 \\ x2 \\ x3 \end{pmatrix};\; b = \begin{pmatrix} 1 \\ 2 \\ -1 \end{pmatrix};\; m.x == b$$

Out[ ]= {{2 x2 + 8 x3}, {3 x1 + 4 x2 + 7 x3}, {5 x1 + 2 x2 + x3}} == {{1}, {2}, {-1}}

Out[ ]//MatrixForm=

$$\begin{pmatrix} 0 & 2 & 8 & 1 \\ 3 & 4 & 7 & 2 \\ 5 & 2 & 1 & -1 \end{pmatrix}$$

Out[ ]//MatrixForm=

$$\begin{pmatrix} 1 & 0 & 0 & -\frac{3}{4} \\ 0 & 1 & 0 & \frac{3}{2} \\ 0 & 0 & 1 & -\frac{1}{4} \end{pmatrix}$$

Out[ ]= $\left\{\left\{-\frac{3}{4}\right\}, \left\{\frac{3}{2}\right\}, \left\{-\frac{1}{4}\right\}\right\}$

Q2= Solve system of linear equation usning gauss elim.

2y+3z = 1;

7x+2y+4z = 2;

x+2y+z = 3;

In[ ]:=
```
m = {{0, 2, 3}, {7, 2, 4}, {1, 2, 1}}; x = {{x1}, {x2}, {x3}}; b = {{1}, {2}, {3}}; m.x == b
ArrayFlatten[{{m, b}}] // MatrixForm
RowReduce[%] // MatrixForm
LinearSolve[m, b]
```

$$m = \begin{pmatrix} 0 & 2 & 3 \\ 7 & 2 & 4 \\ 1 & 2 & 1 \end{pmatrix};\ x = \begin{pmatrix} x1 \\ x2 \\ x3 \end{pmatrix};\ b = \begin{pmatrix} 1 \\ 2 \\ 3 \end{pmatrix};\ m.x == b$$

Out[ ]= {{2 x2 + 3 x3}, {7 x1 + 2 x2 + 4 x3}, {x1 + 2 x2 + x3}} == {{1}, {2}, {3}}

Out[ ]//MatrixForm=

$$\begin{pmatrix} 0 & 2 & 3 & 1 \\ 7 & 2 & 4 & 2 \\ 1 & 2 & 1 & 3 \end{pmatrix}$$

Out[ ]//MatrixForm=

$$\begin{pmatrix} 1 & 0 & 0 & \frac{4}{15} \\ 0 & 1 & 0 & \frac{9}{5} \\ 0 & 0 & 1 & -\frac{13}{15} \end{pmatrix}$$

Out[ ]= $\left\{\left\{\frac{4}{15}\right\}, \left\{\frac{9}{5}\right\}, \left\{-\frac{13}{15}\right\}\right\}$

Q3= Solve system of linear equation usning gauss elim.

x+2y+z = 7;

2x+3y+z = 10;

3x-y+4z = 13;

In[ ]:=
```
m = {{1, 2, 1}, {2, 3, 1}, {3, -1, 4}}; x = {{x1}, {x2}, {x3}}; b = {{3}, {10}, {13}}; m.x == b
ArrayFlatten[{{m, b}}] // MatrixForm
RowReduce[%] // MatrixForm
LinearSolve[m, b]
```

$$m = \begin{pmatrix} 1 & 2 & 1 \\ 2 & 3 & 1 \\ 3 & -1 & 4 \end{pmatrix};\ x = \begin{pmatrix} x1 \\ x2 \\ x3 \end{pmatrix};\ b = \begin{pmatrix} 3 \\ 10 \\ 13 \end{pmatrix};\ m.x == b$$

Out[ ]= {{x1 + 2 x2 + x3}, {2 x1 + 3 x2 + x3}, {3 x1 − x2 + 4 x3}} == {{3}, {10}, {13}}

Out[ ]//MatrixForm=

$$\begin{pmatrix} 1 & 2 & 1 & 3 \\ 2 & 3 & 1 & 10 \\ 3 & -1 & 4 & 13 \end{pmatrix}$$

Out[ ]//MatrixForm=

$$\begin{pmatrix} 1 & 0 & 0 & 8 \\ 0 & 1 & 0 & -1 \\ 0 & 0 & 1 & -3 \end{pmatrix}$$

Out[ ]= {{8}, {−1}, {−3}}

Gauss Jacobi Method

Q1= Solve the following equation using jacobi method

27x + 6y - z = 65

6x + y + 2z = 72

x + y + 5z = 110

Taking initial approx. =[0,0,0]

In[ ]:=
```
Jacobi[A0_, B0_, P0_, max_] := Module[{A = N[A0], B = N[B0],
   i, j, k = 0, n = Length[P0], P = P0, Pold = P0}, Print[" P "_0, " = ", P];
  While[k < max,
   For[i = 1, i ≤ n, i++,
    P[[i]] = 1/A[[i, i]] (B[[i]] + A[[i, i]] × Pold[[i]] - Sum[A[[i, j]] × Pold[[j]], {j, 1, n}])];
   Print["P "_(k+1), " = ", P];
   Pold = P;
   k = k + 1;];
  Return[P];];
A = {{27, 6, -1}, {6, 1, 2}, {1, 1, 5}};
B = {65, 72, 110};
vars = {"x1", "x2", "x3"};
Print["Solve the system"];
Print[MatrixForm[A], MatrixForm[vars], " = ", MatrixForm[B]]
P = {0, 0, 0};
X = Jacobi[A, B, P, 5];
```

$$P[[i]] = \frac{1}{A[[i,i]]}\left(B[[i]] + A[[i,i]] \times Pold[[i]] - \sum_{j=1}^{n} A[[i,j]] \times Pold[[j]]\right)$$

$$A = \begin{pmatrix} 27 & 6 & -1 \\ 6 & 1 & 2 \\ 1 & 1 & 5 \end{pmatrix};$$

```
Solve the system
```

$$\begin{pmatrix} 27 & 6 & -1 \\ 6 & 1 & 2 \\ 1 & 1 & 5 \end{pmatrix}\begin{pmatrix} x1 \\ x2 \\ x3 \end{pmatrix} = \begin{pmatrix} 65 \\ 72 \\ 110 \end{pmatrix}$$

```
 P 0 = {0, 0, 0}
P 1 = {2.40741 , 72. , 22.}
P 2 = {-12.7778 , 13.5556 , 7.11852 }
P 3 = {-0.341289 , 134.43 , 21.8444 }
P 4 = {-26.6568 , 30.3588 , -4.81767 }
P 5 = {-4.51744 , 241.576 , 21.2596 }
```

Q2= Solve the following equation using jacobi method

x + y + 3z = 5

x + y - 3z = -7

-2x - y - 4z = 48

Taking initial approx. =[0.5,0.5,0.5]

In[ ]:=

```
Jacobi[A0_, B0_, P0_, max_] := Module[{A = N[A0], B = N[B0],
    i, j, k = 0, n = Length[P0], P = P0, Pold = P0}, Print[" P "0, " = ", P];
   While[k < max,
    For[i = 1, i ≤ n, i++,
     P[[i]] = 1/A[[i, i]] (B[[i]] + A[[i, i]] × Pold[[i]] - Sum[A[[i, j]] × Pold[[j]], {j, 1, n}])];
    Print["P "k+1, " = ", P];
    Pold = P;
    k = k + 1;];
   Return[P];];
A = {{1, 1, 3}, {1, 1, -3}, {-2, -1, -4}};
B = {5, -7, 48};
vars = {"x1", "x2", "x3"};
Print["Solve the system"];
Print[MatrixForm[A], MatrixForm[vars], " = ", MatrixForm[B]]
P = {0.5, 0.5, 0.5};
X = Jacobi[A, B, P, 5];
```

Solve the system

$$\begin{pmatrix} 1 & 1 & 3 \\ 1 & 1 & -3 \\ -2 & -1 & -4 \end{pmatrix} \begin{pmatrix} x1 \\ x2 \\ x3 \end{pmatrix} = \begin{pmatrix} 5 \\ -7 \\ 48 \end{pmatrix}$$

$P_0 = \{0.5, 0.5, 0.5\}$

$P_1 = \{3., -6., -12.375\}$

$P_2 = \{48.125, -47.125, -12.\}$

$P_3 = \{88.125, -91.125, -24.2813\}$

$P_4 = \{168.969, -167.969, -33.2813\}$

$P_5 = \{272.813, -275.813, -54.4922\}$

Q3= Solve the following equation using jacobi method

x + y + z +3w = 8

2x + 3y + 8z - 7w= 20

4x + 3y + 2z +w = 6

-7x + 3y+3z + 4w = 4

Taking initial approx. =[1,1,1,1]

In[ ]:=
```
Jacobi[A0_, B0_, P0_, max_] := Module[{A = N[A0], B = N[B0],
    i, j, k = 0, n = Length[P0], P = P0, Pold = P0}, Print[" P "_0, " = ", P];
   While[k < max,
    For[i = 1, i ≤ n, i++,
     P[[i]] = 1/A[[i, i]] (B[[i]] + A[[i, i]] × Pold[[i]] - Sum[A[[i, j]] × Pold[[j]], {j, 1, n}])];
    Print["P "_(k+1), " = ", P];
    Pold = P;
    k = k + 1;];
   Return[P];];
A = {{1, 1, 1, 3}, {2, 3, 8, -7}, {4, 3, 2, 1}, {-7, 3, 3, 4}};
B = {8, 20, 6, 4};
vars = {"x1", "x2", "x3", "x4"};
Print["Solve the system"];
Print[MatrixForm[A], MatrixForm[vars], " = ", MatrixForm[B]]
P = {1, 1, 1, 1};
X = Jacobi[A, B, P, 5];
```

Solve the system

$$\begin{pmatrix} 1 & 1 & 1 & 3 \\ 2 & 3 & 8 & -7 \\ 4 & 3 & 2 & 1 \\ -7 & 3 & 3 & 4 \end{pmatrix}\begin{pmatrix} x1 \\ x2 \\ x3 \\ x4 \end{pmatrix} = \begin{pmatrix} 8 \\ 20 \\ 6 \\ 4 \end{pmatrix}$$

P_0 = {1, 1, 1, 1}

P_1 = {3., 5.66667, -1., 1.25}

P_2 = {-0.416667, 10.25, -12.125, 2.75}

P_3 = {1.625, 45.6944, -12.9167, 1.67708}

P_4 = {-29.809, 43.941, -69.6302, -20.7396}

P_5 = {95.908, 163.828, 7.07639, -31.8989}

Lagrange’s interpolation Method

Q1 = The function y = f(x) is given at the point (7,3) , (8,1) , (9,1) , (10,9)

Find f(9.5) using Lagrange Interpolation

```
In[ ]:= No = 4; sum = 0;
lagrange[No_, n_] :=
  Product[If[Equal[k, n], 1, (x - x[k]) / (x[n] - x[k])], {k, 1, No}];
For[i = 1, i ≤ No, i++, sum += (f[x[i]] * lagrange[No, i])];
Print[sum]
sum = 0;
points = {{7, 3}, {8, 1}, {9, 1}, {10, 9}};
No = Length[points]
y = points[[All, 1]]
f = points[[All, 2]]
lagrange[No_, n_] :=
 Product[If[Equal[k, n], 1, (x - y[[k]]) / (y[[n]] - y[[k]])], {k, 1, No}]
For[i = 1, i ≤ No, i++, sum += (f[[i]] * lagrange[No, i])]
Expand[sum]
sum /. x → 9.5
```

```
{{((x1 - {{x1}, {x2}, {x3}}[2]) (x1 - {{x1}, {x2}, {x3}}[3]) (x1 - {{x1}, {x2}, {x3}}[4]) {2, 1, 1, 9}[{{x1}, {x2}, {x3}}[1]]) /
     (({{x1}, {x2}, {x3}}[1] - {{x1}, {x2}, {x3}}[2])
       ({{x1}, {x2}, {x3}}[1] - {{x1}, {x2}, {x3}}[3]) ({{x1}, {x2}, {x3}}[1] - {{x1}, {x2}, {x3}}[4])) +
   ((x1 - {{x1}, {x2}, {x3}}[1]) (x1 - {{x1}, {x2}, {x3}}[3]) (x1 - {{x1}, {x2}, {x3}}[4])
       {2, 1, 1, 9}[{{x1}, {x2}, {x3}}[2]]) / ((-{{x1}, {x2}, {x3}}[1] + {{x1}, {x2}, {x3}}[2])
       ({{x1}, {x2}, {x3}}[2] - {{x1}, {x2}, {x3}}[3]) ({{x1}, {x2}, {x3}}[2] - {{x1}, {x2}, {x3}}[4])) +
   ((x1 - {{x1}, {x2}, {x3}}[1]) (x1 - {{x1}, {x2}, {x3}}[2]) (x1 - {{x1}, {x2}, {x3}}[4])
       {2, 1, 1, 9}[{{x1}, {x2}, {x3}}[3]]) / ((-{{x1}, {x2}, {x3}}[1] + {{x1}, {x2}, {x3}}[3])
       (-{{x1}, {x2}, {x3}}[2] + {{x1}, {x2}, {x3}}[3]) ({{x1}, {x2}, {x3}}[3] - {{x1}, {x2}, {x3}}[4])) +
   ((x1 - {{x1}, {x2}, {x3}}[1]) (x1 - {{x1}, {x2}, {x3}}[2]) (x1 - {{x1}, {x2}, {x3}}[3])
       {2, 1, 1, 9}[{{x1}, {x2}, {x3}}[4]]) / ((-{{x1}, {x2}, {x3}}[1] + {{x1}, {x2}, {x3}}[4])
       (-{{x1}, {x2}, {x3}}[2] + {{x1}, {x2}, {x3}}[4]) (-{{x1}, {x2}, {x3}}[3] + {{x1}, {x2}, {x3}}[4]))},
 {((x2 - {{x1}, {x2}, {x3}}[2]) (x2 - {{x1}, {x2}, {x3}}[3]) (x2 - {{x1}, {x2}, {x3}}[4])
       {2, 1, 1, 9}[{{x1}, {x2}, {x3}}[1]]) / (({{x1}, {x2}, {x3}}[1] - {{x1}, {x2}, {x3}}[2])
       ({{x1}, {x2}, {x3}}[1] - {{x1}, {x2}, {x3}}[3]) ({{x1}, {x2}, {x3}}[1] - {{x1}, {x2}, {x3}}[4])) +
   ((x2 - {{x1}, {x2}, {x3}}[1]) (x2 - {{x1}, {x2}, {x3}}[3]) (x2 - {{x1}, {x2}, {x3}}[4])
       {2, 1, 1, 9}[{{x1}, {x2}, {x3}}[2]]) / ((-{{x1}, {x2}, {x3}}[1] + {{x1}, {x2}, {x3}}[2])
       ({{x1}, {x2}, {x3}}[2] - {{x1}, {x2}, {x3}}[3]) ({{x1}, {x2}, {x3}}[2] - {{x1}, {x2}, {x3}}[4])) +
   ((x2 - {{x1}, {x2}, {x3}}[1]) (x2 - {{x1}, {x2}, {x3}}[2]) (x2 - {{x1}, {x2}, {x3}}[4])
       {2, 1, 1, 9}[{{x1}, {x2}, {x3}}[3]]) / ((-{{x1}, {x2}, {x3}}[1] + {{x1}, {x2}, {x3}}[3])
       (-{{x1}, {x2}, {x3}}[2] + {{x1}, {x2}, {x3}}[3]) ({{x1}, {x2}, {x3}}[3] - {{x1}, {x2}, {x3}}[4])) +
   ((x2 - {{x1}, {x2}, {x3}}[1]) (x2 - {{x1}, {x2}, {x3}}[2]) (x2 - {{x1}, {x2}, {x3}}[3])
       {2, 1, 1, 9}[{{x1}, {x2}, {x3}}[4]]) / ((-{{x1}, {x2}, {x3}}[1] + {{x1}, {x2}, {x3}}[4])
       (-{{x1}, {x2}, {x3}}[2] + {{x1}, {x2}, {x3}}[4]) (-{{x1}, {x2}, {x3}}[3] + {{x1}, {x2}, {x3}}[4]))},
 {((x3 - {{x1}, {x2}, {x3}}[2]) (x3 - {{x1}, {x2}, {x3}}[3]) (x3 - {{x1}, {x2}, {x3}}[4])
       {2, 1, 1, 9}[{{x1}, {x2}, {x3}}[1]]) / (({{x1}, {x2}, {x3}}[1] - {{x1}, {x2}, {x3}}[2])
       ({{x1}, {x2}, {x3}}[1] - {{x1}, {x2}, {x3}}[3]) ({{x1}, {x2}, {x3}}[1] - {{x1}, {x2}, {x3}}[4])) +
   ((x3 - {{x1}, {x2}, {x3}}[1]) (x3 - {{x1}, {x2}, {x3}}[3]) (x3 - {{x1}, {x2}, {x3}}[4])
       {2, 1, 1, 9}[{{x1}, {x2}, {x3}}[2]]) / ((-{{x1}, {x2}, {x3}}[1] + {{x1}, {x2}, {x3}}[2])
       ({{x1}, {x2}, {x3}}[2] - {{x1}, {x2}, {x3}}[3]) ({{x1}, {x2}, {x3}}[2] - {{x1}, {x2}, {x3}}[4])) +
   ((x3 - {{x1}, {x2}, {x3}}[1]) (x3 - {{x1}, {x2}, {x3}}[2]) (x3 - {{x1}, {x2}, {x3}}[4])
       {2, 1, 1, 9}[{{x1}, {x2}, {x3}}[3]]) / ((-{{x1}, {x2}, {x3}}[1] + {{x1}, {x2}, {x3}}[3])
       (-{{x1}, {x2}, {x3}}[2] + {{x1}, {x2}, {x3}}[3]) ({{x1}, {x2}, {x3}}[3] - {{x1}, {x2}, {x3}}[4])) +
   ((x3 - {{x1}, {x2}, {x3}}[1]) (x3 - {{x1}, {x2}, {x3}}[2]) (x3 - {{x1}, {x2}, {x3}}[3])
       {2, 1, 1, 9}[{{x1}, {x2}, {x3}}[4]]) / ((-{{x1}, {x2}, {x3}}[1] + {{x1}, {x2}, {x3}}[4])
       (-{{x1}, {x2}, {x3}}[2] + {{x1}, {x2}, {x3}}[4]) (-{{x1}, {x2}, {x3}}[3] + {{x1}, {x2}, {x3}}[4]))}}
```

Out[ ]= 4

Out[ ]= {7, 8, 9, 10}

Out[ ]= {3, 1, 1, 9}

Out[ ]= $\{\{-431 + 174\, x1 - 23\, x1^2 + x1^3\}, \{-431 + 174\, x2 - 23\, x2^2 + x2^3\}, \{-431 + 174\, x3 - 23\, x3^2 + x3^3\}\}$

Out[ ]=
$$\left\{\left\{\frac{1}{2}(8-x1)(9-x1)(10-x1)+\frac{1}{2}(9-x1)(10-x1)(-7+x1)+\frac{1}{2}(10-x1)(-8+x1)(-7+x1)+\frac{3}{2}(-9+x1)(-8+x1)(-7+x1)\right\},\left\{\frac{1}{2}(8-x2)(9-x2)(10-x2)+\frac{1}{2}(9-x2)(10-x2)(-7+x2)+\frac{1}{2}(10-x2)(-8+x2)(-7+x2)+\frac{3}{2}(-9+x2)(-8+x2)(-7+x2)\right\},\left\{\frac{1}{2}(8-x3)(9-x3)(10-x3)+\frac{1}{2}(9-x3)(10-x3)(-7+x3)+\frac{1}{2}(10-x3)(-8+x3)(-7+x3)+\frac{3}{2}(-9+x3)(-8+x3)(-7+x3)\right\}\right\}$$

Q 2= Use Lagrange’s interpolation to find the unique polynomial p(x) of degree 2 such that

p(1) = 1 , p(3) = 27 , p(4) = 64

In[ ]:=
```
No = 2; sum = 0;
lagrange[No_, n_] :=
  Product[If[Equal[k, n], 1, (x - x[k]) / (x[n] - x[k])], {k, 1, No}];
For[i = 1, i ≤ No, i++, sum += (f[x[i]] * lagrange[No, i])];
Print[sum]
sum = 0;
points = {{1, 1}, {3, 27}, {4, 64}};
No = Length[points]
y = points[[All, 1]]
f = points[[All, 2]]
lagrange[No_, n_] :=
 Product[If[Equal[k, n], 1, (x - y[[k]]) / (y[[n]] - y[[k]])], {k, 1, No}]
For[i = 1, i ≤ No, i++, sum += (f[[i]] * lagrange[No, i])]
Expand[sum]
```

$$\frac{(x-x[2])\,\{1,\ 27,\ 64\}[x[1]]}{x[1]-x[2]}+\frac{(x-x[1])\,\{1,\ 27,\ 64\}[x[2]]}{-x[1]+x[2]}$$

Out[ ]= 3

Out[ ]= {1, 3, 4}

Out[ ]= {1, 27, 64}

Out[ ]= $12-19\,x+8\,x^2$

Q 3= Use Lagrange’s interpolation to find the unique polynomial p(x) of degree 2 such that

p(6) = 0.175 , p(6.1) = -0.199 , p(6.2) = -0.222

```
In[ ]:= No = 3; sum = 0;
lagrange[No_, n_] :=
  Product[If[Equal[k, n], 1, (x - x[k]) / (x[n] - x[k])], {k, 1, No}];
For[i = 1, i ≤ No, i++, sum += (f[x[i]] * lagrange[No, i])];
Print[sum]
sum = 0;
points = {{6, 0.175}, {6.1, -0.199}, {6.2, -0.222}};
No = Length[points]
y = points[[All, 1]]
f = points[[All, 2]]
lagrange[No_, n_] :=
 Product[If[Equal[k, n], 1, (x - y[[k]]) / (y[[n]] - y[[k]])], {k, 1, No}]
For[i = 1, i ≤ No, i++, sum += (f[[i]] * lagrange[No, i])]
Expand[sum]
```

$$\frac{(x-x[2])\,(x-x[3])\,\{1,27,64\}[x[1]]}{(x[1]-x[2])\,(x[1]-x[3])}+\frac{(x-x[1])\,(x-x[3])\,\{1,27,64\}[x[2]]}{(-x[1]+x[2])\,(x[2]-x[3])}+\frac{(x-x[1])\,(x-x[2])\,\{1,27,64\}[x[3]]}{(-x[1]+x[3])\,(-x[2]+x[3])}$$

```
Out[ ]= 3
Out[ ]= {6, 6.1, 6.2}
Out[ ]= {0.175, -0.199, -0.222}
```

Out[ ]= $664.945 - 216.095\, x + 17.55\, x^2$

Newton Interpolation

Q 1 = Use Newton’s interpolation to find the unique polyno-

mial p(x) such that

p(1) = 1 , p(3) = 27 , p(4) = 64 and Find p(3.5)

In[ ]:=
```
sum = 0;
points = {{1, 1}, {3, 27}, {4, 64}};
n = Length[points]
y = points[[All, 1]]
f = points[[All, 2]]
dd[k_] :=
 Sum[(f[[i]] / Product[If[Equal[j, i], 1, (y[[i]] - y[[j]])], {j, 1, k}]), {i, 1, k}]
p[x_] = Sum[(dd[i] * Product[If[i ≤ j, 1, x - y[[j]]], {j, 1, i - 1}]), {i, 1, n}]
Simplify[p[x]]
Evaluate[p[3.5]]
```

Out[ ]= 3

Out[ ]= {1, 3, 4}

Out[ ]= {1, 27, 64}

Out[ ]= $1 + 13 (-1 + x) + 8 (-3 + x) (-1 + x)$

Out[ ]= $12 - 19 x + 8 x^2$

Out[ ]= 43.5

Q 2 = Use Newton’s interpolation to find the unique polynomial p(x) such that

p(0) = -1.5 , p(0.1) = -1.27 , p(0.2) = -0.98 , p(0.3) = -0.63 , p(0.4) = -0.22 and Find p(0.15)

```
sum = 0;
points = {{0, -1.5}, {0.1, -1.27}, {0.2, -0.98}, {0.3, -0.63}, {0.4, -0.22}};
n = Length[points]
y = points[[All, 1]] x^4 - x^2 - 15
f = points[[All, 2]]
dd[k_] :=
  Sum[(f[[i]] / Product[If[Equal[j, i], 1, (y[[i]] - y[[j]])], {j, 1, k}]), {i, 1, k}]
p[x_] = Sum[(dd[i] * Product[If[i ≤ j, 1, x - y[[j]]], {j, 1, i - 1}]), {i, 1, n}]
Simplify[p[x]]
Evaluate[p[0.15]]
```

Out[ ]= 5

Out[ ]= {0, 0.1, 0.2, 0.3, 0.4}

Out[ ]= {-1.5, -1.27, -0.98, -0.63, -0.22}

Out[ ]= $-1.5 + 2.3\, x + 3.\, (-0.1 + x)\, x - 4.26326 \times 10^{-14}\, (-0.2 + x)\, (-0.1 + x)\, x - 5.54223 \times 10^{-13}\, (-0.3 + x)\, (-0.2 + x)\, (-0.1 + x)\, x$

Out[ ]= $-1.5 + 2.\, x + 3.\, x^2 + 2.89901 \times 10^{-13}\, x^3 - 5.54223 \times 10^{-13}\, x^4$

Out[ ]= -1.1325

Q 3 = Use Newton’s interpolation to find the unique polynomial p(x) osuch that

p(0) = 1 , p(1) = 3 , p(3) = 5.5 and Find p(2.5)

In[ ]:=

```
sum = 0;
points = {{0, 1}, {1, 3}, {3, 5.5}};
n = Length[points]
y = points[[All, 1]]
f = points[[All, 2]]
dd[k_] :=
  Sum[(f[[i]] / Product[If[Equal[j, i], 1, (y[[i]] - y[[j]])], {j, 1, k}]), {i, 1, k}]
p[x_] = Sum[(dd[i] * Product[If[i ≤ j, 1, x - y[[j]]], {j, 1, i - 1}]), {i, 1, n}]
Simplify[p[x]]
Evaluate[p[2.5]]
```

Out[ ]= 3

Out[ ]= {0, 1, 3}

Out[ ]= {1, 3, 5.5}

Out[ ]= $1 + 2\,x - 0.25\,(-1 + x)\,x$

Out[ ]= $1 + 2.25\,x - 0.25\,x^2$

Out[ ]= 5.0625

Trapezoidal Rule

Q 1 = Use Trapezoidal rule to calculate an approx value of integral $\int_0^{\pi/2} \sin x \, dx$

In[ ]:=
```
ClearAll[n, x, f]
a = Input["Enter the left end point : "]
b = Input["Enter the right end point : "]
n = Input["Enter the number of sub interval to be formed : "]
sum = 0
h = (b - a) / n
f[x] = Sin[x]
For[i = 1, i ≤ n - 1, i++, sum += N[f[x] /. x → (a + i * h)]]
sum = N[(2 * sum + f[x] /. x → b) * h / 2]
```

Out[ ]= 0

Out[ ]= $\frac{\pi}{2}$

Out[ ]= 10

Out[ ]= 0

Out[ ]= $\frac{\pi}{20}$

Out[ ]= Sin[x]

Out[ ]= 0.997943

Q 2 = Use Trapezoidal rule to calculate an approx value of integral $\int_{1/2}^{1} \frac{1}{x}\, dx$

In[ ]:=
```
ClearAll[n, x, f]
a = Input["Enter the left end point : "]
b = Input["Enter the right end point : "]
n = Input["Enter the number of sub interval to be formed : "]
sum = 0
h = (b - a) / n
f[x] = 1/x
For[i = 1, i ≤ n - 1, i++, sum += N[f[x] /. x → (a + i * h)]]
sum = N[(2 * sum + f[x] /. x → b) * h / 2]
```

Out[ ]= $\frac{1}{2}$

Out[ ]= 1

Out[ ]= 4

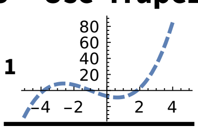
Out[ ]= 0

Out[ ]= $\frac{1}{8}$

Out[ ]= $\frac{1}{x}$

Out[ ]= 0.572024

Q 3 = Use Trapezoidal rule to calculate an approx value of integral



$$\int_0^1 \frac{1}{x^2+6x+10} d x$$

In[ ]:=
```
ClearAll[n, x, f]
a = Input["Enter the left end point : "]
b = Input["Enter the right end point : "]
n = Input["Enter the number of sub interval to be formed : "]
sum = 0
h = (b - a) / n
f[x] = 1/(x^2 + 6 x + 10)
For[i = 1, i ≤ n - 1, i++, sum += N[f[x] /. x → (a + i * h)]]
sum = N[(2 * sum + f[x] /. x → b) * h / 2]
```

Out[ ]= 0

Out[ ]= 1

Out[ ]= 4

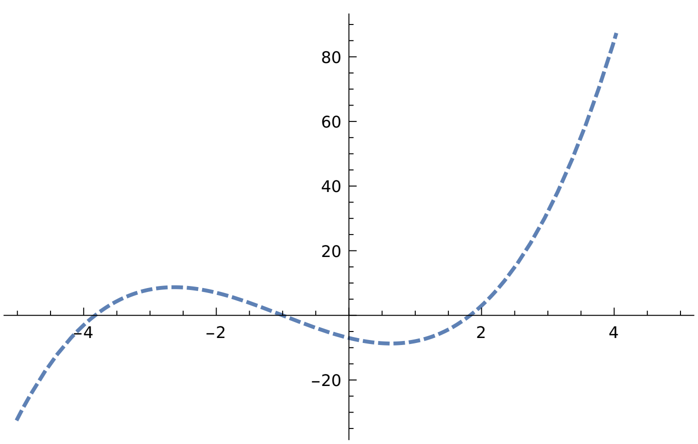
Out[ ]= 0

Out[ ]= $\frac{1}{4}$

Out[ ]= $\frac{1}{10 + 6 x + x^2}$

Out[ ]= 0.06444

Out[ ]=

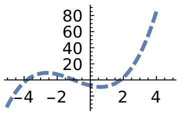


Simpson Rule

Q 1 = Evaluate $\int_1^2 \frac{1}{5+3x} \, dx$, using simpson rule with 8 subinterval and find the

absolute error in the solution

```
a = Input["Enter the left end point : "];
b = Input["Enter the right end point : "];
n = Input["Enter the number of sub interval to be formed : "];
```




```
h = (b - a) / n;
y = Table[a + i * h , {i, 1, n}];
f[x] := 1 / (5 + 3 x);
sumodd = 0;
sumeven = 0;
For[i = 1, i < n, i += 2, sumodd += 4 * f[x] /. x → y[[i]]];
For[i = 2, i < n, i += 2, sumeven += 2 * f[x] /. x → y[[i]]];
Sn = (h / 3) * ((f[x] /. x → a) + N[sumodd] + N[sumeven] + (f[x] /. x → b));
Print["For n =  ", n, " , Simpson estimate is : ", Sn]
in = Integrate[1 / (5 + 3 x) , {x, 1, 2}]
Print["True value is ", in]
Print["absolute error is ", Abs[Sn - in]]
```

For n = 8 , Simpson estimate is : 0.106151

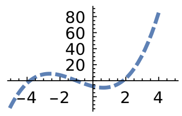
Out[ ]= $\frac{1}{3}\text{Log}\left[\frac{11}{8}\right]$

True value is $\frac{1}{3}\text{Log}\left[\frac{11}{8}\right]$

absolute error is 3.83999×10^{-8}

Q 2 = Evaluate $\int_0^1 \frac{1}{x^2+6x+10}\,dx$, using simpson rule with 2 subinterval and find the absolute error in the solution

```
a = Input["Enter the left end point : "];
b = Input["Enter the right end point : "];
n = Input["Enter the number of sub interval to be formed : "];
h = (b - a) / n;
```




```
y = Table[a + i * h , {i, 1, n}];
f[x] := 1 / (x^2 + 6 x + 10);
sumodd = 0;
sumeven = 0;
For[i = 1, i < n, i += 2, sumodd += 4 * f[x] /. x → y[[i]]];
For[i = 2, i < n, i += 2, sumeven += 2 * f[x] /. x → y[[i]]];
Sn = (h / 3) * ((f[x] /. x → a) + N[sumodd] + N[sumeven] + (f[x] /. x → b));
Print["For n =  ", n, " , Simpson estimate is : ", Sn]
in = Integrate[1 / (x^2 + 6 x + 10), {x, 0, 1}]
Print["True value is ", in]
Print["absolute error is ", Abs[Sn - in]]
```

```
For n =  2 , Simpson estimate is : 0.0767851
```

Out[ ]= -ArcTan[3] + ArcTan[4]

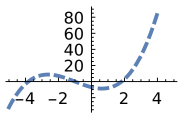
```
True value is -ArcTan[3] + ArcTan[4]
```

```
x ^ 4 - x ^ 2 - 15
```

```
absolute error is 0.0000131624
```

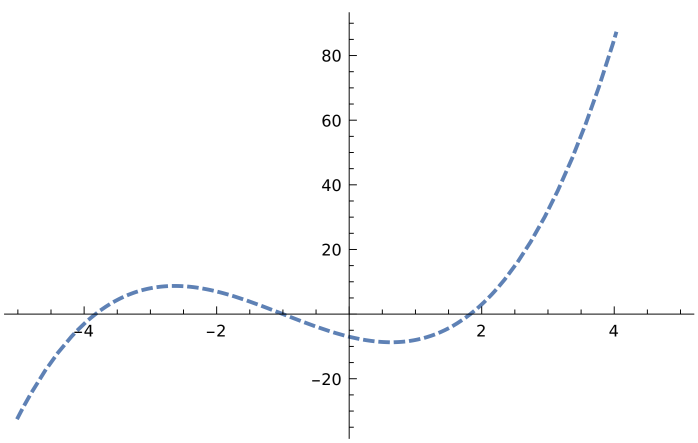
Q 3 = Evaluate $\int_1^2 \frac{1}{\mathrm{Sin}(x) - x^2} \, dx$, using simpson rule with 4 subinterval and find the absolute error in the solution

```
a = Input["Enter the left end point : "];
b = Input["Enter the right end point : "];
n = Input["Enter the number of sub interval to be formed : "];
h = (b - a) / n;
y = Table[a + i * h , {i, 1, n}];
f[x] := 1 / (Sin[x] - x^2);
sumodd = 0;
sumeven = 0;
```




```
For[i = 1, i < n, i += 2, sumodd += 4 * f[x] /. x → y[[i]]];
For[i = 2, i < n, i += 2, sumeven += 2 * f[x] /. x → y[[i]]];
Sn = (h / 3) * ((f[x] /. x → a) + N[sumodd] + N[sumeven] + (f[x] /. x → b));
Print["For n =  ", n, " , Simpson estimate is : ", Sn]
in = Integrate[1 / (Sin[x] - x^2) , {x, 1, 2}]
Print["True value is ", in]
Print["absolute error is ", Abs[Sn - in]]
```

Out[ ]=

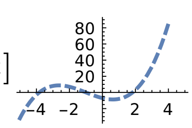


For n = 4 , Simpson estimate is : -1.38938

Out[ ]= $\int_1^2 \frac{1}{-x^2 + \mathrm{Sin}[x]} \, dx$

True value is $\int_1^2 \frac{1}{-x^2 + \mathrm{Sin}[x]} \, dx$

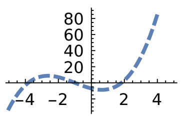
absolute error is $\mathrm{Abs}\left[-1.38938 - \int_1^2 \frac{1}{-x^2 + \mathrm{Sin}[x]} \, dx\right]$



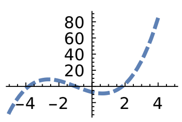
Euler Method

Q 1 = Use Euler’s Method with h = 0.1 to find the solution of $\frac{dy}{dx} = x^2 + y^2$, y(0) = 0

```
Euler[x_, y_] := y + h (x ^ 2 + y ^ 2)
h = 0.1; y = 0
```




```
y = Euler[0, y]
y = Euler[0.1, y]
y = Euler[0.2, y]
y = Euler[0.3, y]
y = Euler[0.4, y]
y = Euler[0.5, y]
y = Euler[0.6, y]
y = Euler[0.7, y]
y = Euler[0.8, y
```




```
    y = Euler[0.9, y]
```

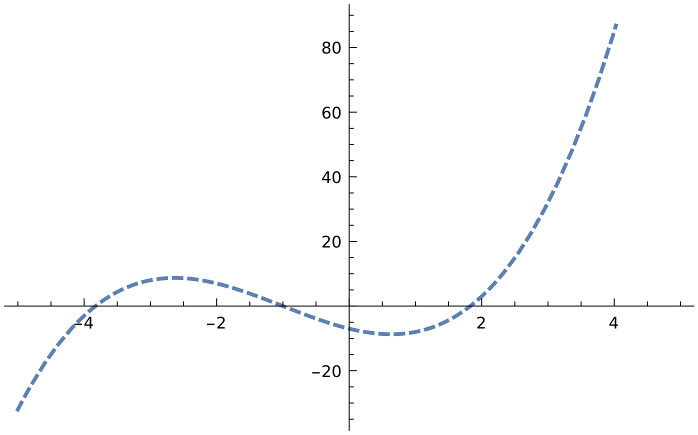
Out[ ]= 0

Out[ ]= 0.

Out[ ]= 0.001

Out[ ]= 0.0050001

Out[ ]= 0.0140026

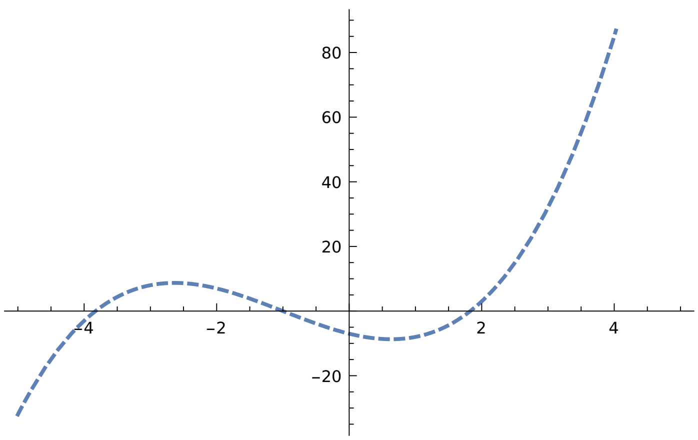
Out[ ]= 

Out[ ]= 0.0300222

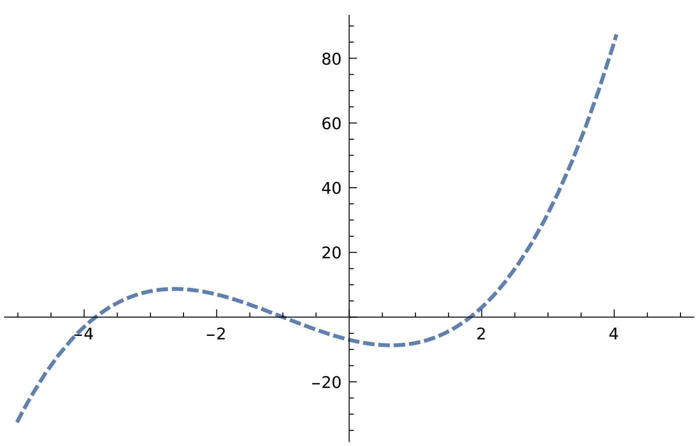
Out[ ]= 0.0551123

Out[ ]= 0.0914161

Out[ ]= 0.141252

Out[ ]= 

Out[ ]= 0.207247

Out[ ]= 

Out[ ]= 0.292542

Q 2 = Use Euler’s Method with h = 0.2 to find the solution of $\frac{dy}{dx} = -2\,xy^2$

In[ ]:=
```
Euler[x_, y_] := y + h (-2 xy^2)
h = 0.2; y = 1
y = Euler[0, y]
y = Euler[0.1, y]
y = Euler[0.2, y]
y = Euler[0.3, y]
y = Euler[0.4, y]
y = Euler[0.5, y]
y = Euler[0.6, y]
y = Euler[0.7, y]
y = Euler[0.8, y]
y = Euler[0.9, y]
```

Out[ ]= 1

Out[ ]= $1 - 0.4\,xy^2$

Out[ ]= $1 - 0.8\,xy^2$

Out[ ]= $1 - 1.2\,xy^2$

Out[ ]= $1 - 1.6\,xy^2$

Out[ ]= $1 - 2.\,xy^2$

Out[ ]= $1 - 2.4\,xy^2$

Out[ ]= $1 - 2.8\,xy^2$

Out[ ]= $1 - 3.2\,xy^2$

Out[ ]= $1 - 3.6\,xy^2$

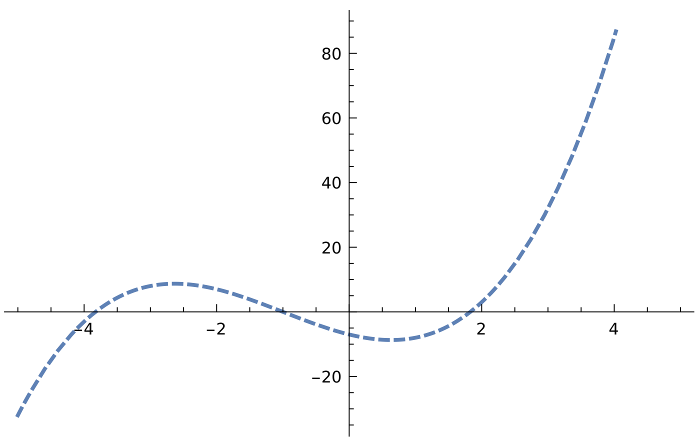
Out[ ]= $1 - 4.\,xy^2$

Q 3 = Use Euler’s Method with h = 0.1 to find the solution of $\frac{dy}{dx} = x + y$

```
Euler[x_, y_] := y + h (x + y)
h = 0.1; y = 1
y = Euler[0, y]
y = Euler[0.1, y]
y = Euler[0.2, y]
y = Euler[0.3, y]
y = Euler[0.4, y] x ^ 4 - x ^ 2 - 15
y = Euler[0.5, y]
y = Euler[0.6, y]
y = Euler[0.7, y]
y = Euler[0.8, y]
y = Euler[0.9, y]
```

Out[ ]= 1

Out[ ]=



Out[ ]= 1.1

Out[ ]= 1.22

Out[ ]= 1.362

Out[ ]= 1.5282

Out[ ]= 1.72102

Out[ ]= 1.94312

Out[ ]= 2.19743

Out[ ]= 2.48718

Out[ ]= 2.8159

Out[ ]= 3.18748