Name: PRACHI AGGARWAL

Course: BSC(H) COMPUTER SCIENCE, 2ND YEAR, 4TH SEMESTER

Examination Rollno: 21033570042

College Rollno: 21570015

Unique Paper Code: 32355402

Paper Name: NUMERICAL METHODS

Submitted to: MR. SANJAY KUMAR SIR

Practical file Numerical Methods

1) BISECTION METHOD

a) Find the real root of the equation x^3-5*x+1, by the method of bisection method in 10 iterations.

```
In[ ]:= f[x_] = x^3 - 5 * x + 1
        a = 0
        b = 1
        n = 10
        Print["The given function is: ", f[x]]
Out[ ]=
        1 - 5 x + x^3
Out[ ]=
        0
Out[ ]=
        1
Out[ ]=
        10
        The given function is: 1 - 5 x + x^3
In[ ]:= i = 0
        c = (a + b) / 2
        OutputDetails = {{i, N[a], N[b], N[c], N[f[a]], N[f[b]], N[f[c]]}}
        For[i = 1, i ≤ n, i++,
         If[f[a] * f[c] < 0, b = c, a = c];
         c = N[(a + b) / 2];
         OutputDetails =
          Append[OutputDetails, {i, N[a], N[b], N[c], N[f[a]], N[f[b]], N[f[c]]}]]
        Print[NumberForm[TableForm[OutputDetails,
           TableHeadings → {None, {"i", "a", "b", "c", "f[a]", "f[b]", "f[c]"}}], 8]]
        Print["Root after ", 10, " iterations ", NumberForm[N[c], 8]];
        Print["accuracy= ", N[Abs[(b - a) / 2]]];
        Print["Function value at approximated root f[c]= ", NumberForm[N[f[c]], 8]]
```

```
        0
Out[ ]=
        1
        -
        2
Out[ ]=
        {{0, 0., 1., 0.5, 1., -3., -1.375}}
```

i	a	b	c	f[a]	f[b]	f[c]
0	0.	1.	0.5	1.	-3.	-1.375
1	0.	0.5	0.25	1.	-1.375	-0.234375
2	0.	0.25	0.125	1.	-0.234375	0.37695313
3	0.125	0.25	0.1875	0.37695313	-0.234375	0.069091797
4	0.1875	0.25	0.21875	0.069091797	-0.234375	-0.083282471
5	0.1875	0.21875	0.203125	0.069091797	-0.083282471	-0.0072441101
6	0.1875	0.203125	0.1953125	0.069091797	-0.0072441101	0.030888081
7	0.1953125	0.203125	0.19921875	0.030888081	-0.0072441101	0.011812866
8	0.19921875	0.203125	0.20117188	0.011812866	-0.0072441101	0.0022820756
9	0.20117188	0.203125	0.20214844	0.0022820756	-0.0072441101	-0.0024815956
10	0.20117188	0.20214844	0.20166016	0.0022820756	-0.0024815956	-0.000099904253

Root after 10 iterations 0.20166016

accuracy= 0.000488281

Function value at approximated root f[c]= -0.000099904253

b) Find the real root of the equation x^3-x-4, by the method of bisection method in 10 iterations.

```
In[14]:= f[x_] = x^3 - x - 4
a = 0
b = 1
n = 10
Print["The given function is: ", f[x]]
i = 0
c = (a + b) / 2
OutputDetails = {{i, N[a], N[b], N[c], N[f[a]], N[f[b]], N[f[c]]}}
For[i = 1, i ≤ n, i++,
 If[f[a] * f[c] < 0, b = c, a = c];
 c = N[(a + b) / 2];
 OutputDetails =
  Append[OutputDetails, {i, N[a], N[b], N[c], N[f[a]], N[f[b]], N[f[c]]}]]
Print[NumberForm[TableForm[OutputDetails,
   TableHeadings → {None, {"i", "a", "b", "c", "f[a]", "f[b]", "f[c]"}}], 8]]
Print["Root after ", 10, " iterations ", NumberForm[N[c], 8]];
Print["accuracy= ", N[Abs[(b - a) / 2]]];
Print["Function value at approximated root f[c]= ", NumberForm[N[f[c]], 8]]
```

$-4 - x + x^3$

Out[15]=

0

Out[16]=

1

Out[17]=

10

The given function is: $-4 - x + x^3$

Out[19]=

0

Out[20]=

$\frac{1}{2}$

Out[21]=

{{0, 0., 1., 0.5, -4., -4., -4.375}}

i	a	b	c	f[a]	f[b]	f[c]
0	0.	1.	0.5	-4.	-4.	-4.375
1	0.5	1.	0.75	-4.375	-4.	-4.328125
2	0.75	1.	0.875	-4.328125	-4.	-4.2050781
3	0.875	1.	0.9375	-4.2050781	-4.	-4.1135254
4	0.9375	1.	0.96875	-4.1135254	-4.	-4.0596008
5	0.96875	1.	0.984375	-4.0596008	-4.	-4.0305214
6	0.984375	1.	0.9921875	-4.0305214	-4.	-4.0154424
7	0.9921875	1.	0.99609375	-4.0154424	-4.	-4.0077668
8	0.99609375	1.	0.99804688	-4.0077668	-4.	-4.0038948
9	0.99804688	1.	0.99902344	-4.0038948	-4.	-4.0019503
10	0.99902344	1.	0.99951172	-4.0019503	-4.	-4.0009758

Root after 10 iterations 0.99951172

accuracy= 0.000488281

Function value at approximated root f[c]= -4.0009758

c) Find the real root of the equation x^3 - 2 * x - 5, by the method of bisection method in 10 iterations.

```
In[27]:= f[x_] = x^3 - 2 * x - 5
         a = 0
         b = 1
         n = 10
         Print["The given function is: ", f[x]]
         i = 0
         c = (a + b) / 2
         OutputDetails = {{i, N[a], N[b], N[c], N[f[a]], N[f[b]], N[f[c]]}}
         For[i = 1, i ≤ n, i++,
          If[f[a] * f[c] < 0, b = c, a = c];
          c = N[(a + b) / 2];
          OutputDetails =
           Append[OutputDetails, {i, N[a], N[b], N[c], N[f[a]], N[f[b]], N[f[c]]}]]
         Print[NumberForm[TableForm[OutputDetails,
            TableHeadings → {None, {"i", "a", "b", "c", "f[a]", "f[b]", "f[c]"}}], 8]]
         Print["Root after ", 10, " iterations ", NumberForm[N[c], 8]];
         Print["accuracy= ", N[Abs[(b - a) / 2]]];
         Print["Function value at approximated root f[c]= ", NumberForm[N[f[c]], 8]]
Out[27]=
```

```
Out[34]=
{{0, 0., 1., 0.5, -5., -6., -5.875}}
```

i	a	b	c	f[a]	f[b]	f[c]
0	0.	1.	0.5	-5.	-6.	-5.875
1	0.5	1.	0.75	-5.875	-6.	-6.078125
2	0.75	1.	0.875	-6.078125	-6.	-6.0800781
3	0.875	1.	0.9375	-6.0800781	-6.	-6.0510254
4	0.9375	1.	0.96875	-6.0510254	-6.	-6.0283508
5	0.96875	1.	0.984375	-6.0283508	-6.	-6.0148964
6	0.984375	1.	0.9921875	-6.0148964	-6.	-6.0076299
7	0.9921875	1.	0.99609375	-6.0076299	-6.	-6.0038605
8	0.99609375	1.	0.99804688	-6.0038605	-6.	-6.0019417
9	0.99804688	1.	0.99902344	-6.0019417	-6.	-6.0009737
10	0.99902344	1.	0.99951172	-6.0009737	-6.	-6.0004876

```
Root after 10 iterations 0.99951172

accuracy= 0.000488281

Function value at approximated root f[c]= -6.0004876
```

d) Perform 10 iteration of bisection method to find the root of the function: f(x)= x^3 - x -2 in the interval [1,2].

```
In[1]:= f[x_] = x^3 - x - 2;
a = 1;
b = 2;
n = 10
Print["The given function is: ", f[x]]
i = 0;
c = (a + b) / 2;
OutputDetails = {{i, N[a], N[b], N[c], N[f[a]], N[f[b]], N[f[c]]}};
For[i = 1, i ≤ n, i++,
 If[f[a] * f[c] < 0, b = c, a = c];
 c = N[(a + b) / 2];
 OutputDetails =
 Append[OutputDetails, {i, N[a], N[b], N[c], N[f[a]], N[f[b]], N[f[c]]}]]
Print[NumberForm[TableForm[OutputDetails,
   TableHeadings → {None, {"i", "a", "b", "c", "f[a]", "f[b]", "f[c]"}}], 8]]
Print["Root after ", 10, " iterations ", NumberForm[N[c], 8]];
Print["accuracy= ", N[Abs[(b - a) / 2]]];
Print["Function value at approximated root f[c]= ", NumberForm[N[f[c]], 8]]

Out[4]= 10
```

The given function is: $-2 - x + x^3$

i	a	b	c	f[a]	f[b]	f[c]
0	1.	2.	1.5	-2.	4.	-0.125
1	1.5	2.	1.75	-0.125	4.	1.609375
2	1.5	1.75	1.625	-0.125	1.609375	0.66601563
3	1.5	1.625	1.5625	-0.125	0.66601563	0.25219727
4	1.5	1.5625	1.53125	-0.125	0.25219727	0.059112549
5	1.5	1.53125	1.515625	-0.125	0.059112549	-0.034053802
6	1.515625	1.53125	1.5234375	-0.034053802	0.059112549	0.012250423
7	1.515625	1.5234375	1.5195313	-0.034053802	0.012250423	-0.010971248
8	1.5195313	1.5234375	1.5214844	-0.010971248	0.012250423	0.00062217563
9	1.5195313	1.5214844	1.5205078	-0.010971248	0.00062217563	-0.0051788865
10	1.5205078	1.5214844	1.5209961	-0.0051788865	0.00062217563	-0.0022794433

Root after 10 iterations 1.5209961

accuracy= 0.000488281

Function value at approximated root f[c]= -0.0022794433

e) Perform 10 iteration of bisection method to find the root of the function: f(x)= 2x-root(1+sinx) in the interval [0,1].

```
In[14]:= f[x_] = 2 * x - Sqrt[1 + Sin[x]];
a = 0;
b = 1;
n = 10
Print["The given function is: ", f[x]]
i = 0;
c = (a + b) / 2;
OutputDetails = {{i, N[a], N[b], N[c], N[f[a]], N[f[b]], N[f[c]]}};
For[i = 1, i ≤ n, i++,
 If[f[a] * f[c] < 0, b = c, a = c];
 c = N[(a + b) / 2];
 OutputDetails =
 Append[OutputDetails, {i, N[a], N[b], N[c], N[f[a]], N[f[b]], N[f[c]]}]]
Print[NumberForm[TableForm[OutputDetails,
   TableHeadings → {None, {"i", "a", "b", "c", "f[a]", "f[b]", "f[c]"}}], 8]]
Print["Root after ", 10, " iterations ", NumberForm[N[c], 8]];
Print["accuracy= ", N[Abs[(b - a) / 2]]];
Print["Function value at approximated root f[c]= ", NumberForm[N[f[c]], 8]]

Out[17]= 10

The given function is: 2 x - Sqrt[1 + Sin[x]]
```

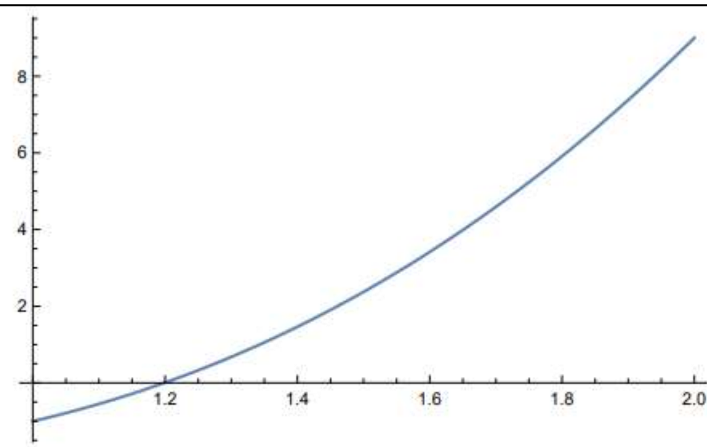
i	a	b	c	f[a]	f[b]	f[c]
0	0.	1.	0.5	-1.	0.6429919	-0.21631638
1	0.5	1.	0.75	-0.21631638	0.6429919	0.20321985
2	0.5	0.75	0.625	-0.21631638	0.20321985	-0.0090064626
3	0.625	0.75	0.6875	-0.0090064626	0.20321985	0.096482468
4	0.625	0.6875	0.65625	-0.0090064626	0.096482468	0.043583109
5	0.625	0.65625	0.640625	-0.0090064626	0.043583109	0.017249749
6	0.625	0.640625	0.6328125	-0.0090064626	0.017249749	0.0041120186
7	0.625	0.6328125	0.62890625	-0.0090064626	0.0041120186	-0.0024496258
8	0.62890625	0.6328125	0.63085938	-0.0024496258	0.0041120186	0.00083059519
9	0.62890625	0.63085938	0.62988281	-0.0024496258	0.00083059519	-0.00080966556
10	0.62988281	0.63085938	0.63037109	-0.00080966556	0.00083059519	0.00001042724

```
Root after 10 iterations 0.63037109
accuracy= 0.000488281
Function value at approximated root f[c]= 0.00001042724
```

2)Secant Method and Regula-Falsi Method

a) Find an interval of unit length that contains the smallest positive root of the function f(x)=x^3+2x^2-3x-1. Perform 15 iterations of method of false position to find the root of the function starting with the resulted interval.

```
f[x_] = x^3 + 2 x^2 - 3 x - 1;
Plot[f[x], {x, 1, 2}]
a = 1
b = 2
n = 15
Print["The given function is: ", f[x]]
```




```
Out[•]=
1
Out[•]=
2
Out[•]=
15
The given function is: -1 - 3 x + 2 x^2 + x^3
```

In[ ]:=
```
i = 0
c = ((a * f[b]) - (b * f[a])) / (f[b] - f[a])
OutputDetails = {{i, N[a], N[b], N[c], N[f[a]], N[f[b]], N[f[c]]}}
For[i = 1, i ≤ n, i++,
 If[f[a] * f[c] < 0, b = c, a = c];
 c = N[((a * f[b]) - (b * f[a])) / (f[b] - f[a])];
 OutputDetails =
  Append[OutputDetails, {i, N[a], N[b], N[c], N[f[a]], N[f[b]], N[f[c]]}]]
Print[NumberForm[TableForm[OutputDetails,
   TableHeadings → {None, {"i", "a", "b", "c", "f[a]", "f[b]", "f[c]"}}], 8]]
Print["Root after ", 15, " iterations ", NumberForm[N[c], 8]];
Print["accuracy= ", N[Abs[(b - a) / 2]]];
Print["Function value at approximated root f[c]= ", NumberForm[N[f[c]], 8]]
```
Out[ ]=

Out[ ]=

0

Out[ ]=

$\frac{11}{10}$

Out[ ]=

{{0, 1., 2., 1.1, -1., 9., -0.549}}

i	a	b	c	f[a]	f[b]	f[c]
0	1.	2.	1.1	-1.	9.	-0.549
1	1.1	2.	1.1517436	-0.549	9.	-0.27440072
2	1.1517436	2.	1.1768409	-0.27440072	9.	-0.13074253
3	1.1768409	2.	1.1886277	-0.13074253	9.	-0.060875863
4	1.1886277	2.	1.1940789	-0.060875863	9.	-0.028040938
5	1.1940789	2.	1.1965821	-0.028040938	9.	-0.01285224
6	1.1965821	2.	1.1977278	-0.01285224	9.	-0.0058772415
7	1.1977278	2.	1.1982513	-0.0058772415	9.	-0.0026848163
8	1.1982513	2.	1.1984904	-0.0026848163	9.	-0.001225881
9	1.1984904	2.	1.1985996	-0.001225881	9.	-0.0005596125
10	1.1985996	2.	1.1986494	-0.0005596125	9.	-0.00025543669
11	1.1986494	2.	1.1986721	-0.00025543669	9.	-0.0001165895
12	1.1986721	2.	1.1986825	-0.0001165895	9.	-0.000053214081
13	1.1986825	2.	1.1986873	-0.000053214081	9.	-0.00002428788
14	1.1986873	2.	1.1986894	-0.00002428788	9.	-0.000011085385
15	1.1986894	2.	1.1986904	-0.000011085385	9.	$-5.0595404 \times 10^{-6}$

Root after 15 iterations 1.1986904

accuracy= 0.400655

Function value at approximated root f[c]= $-5.0595404 \times 10^{-6}$

b) Perform 10 iteration of regular falsi method to find the root of the function: f(x)= x^3 - 5x + 1 in the interval [1,2].

```
In[ ]:= f[x_] = x^3 - 5 * x + 1
        a = 1
        b = 2
        n = 10
        Print["The given function is: ", f[x]]
Out[ ]=
        1 - 5 x + x^3
Out[ ]=
        1
Out[ ]=
        2
Out[ ]=
        10
        The given function is: 1 - 5 x + x^3
        i = 0
        c = ((a * f[b]) - (b * f[a])) / (f[b] - f[a])
Out[ ]=
        0
```

```
In[ ]:= OutputDetails = {{i, N[a], N[b], N[c], N[f[a]], N[f[b]], N[f[c]]}}
        For[i = 1, i ≤ n, i++,
         If[f[a] * f[c] < 0, b = c, a = c];
         c = N[((a * f[b]) - (b * f[a])) / (f[b] - f[a])];
         OutputDetails =
          Append[OutputDetails, {i, N[a], N[b], N[c], N[f[a]], N[f[b]], N[f[c]]}]]
        Print[NumberForm[TableForm[OutputDetails,
           TableHeadings → {None, {"i", "a", "b", "c", "f[a]", "f[b]", "f[c]"}}], 8]]
        Print["Root after ", 10, " iterations ", NumberForm[N[c], 8]];
        Print["accuracy= ", N[Abs[(b - a) / 2]]];
        Print["Function value at approximated root f[c]= ", NumberForm[N[f[c]], 8]]
Out[ ]=
        {{11, 2.12793, 2.13086, 2.12939, -0.00420246, 0.0210015, 0.00838583}}
```

i	a	b	c	f[a]	f[b]	f[c]
11	2.1279297	2.1308594	2.1293945	-0.0042024599	0.02100154	0.0083858325
1	2.1279297	2.1293945	2.1284187	-0.0042024599	0.0083858325	$-3.0472645 \times 10^{-6}$
2	2.1284187	2.1293945	2.1284191	$-3.0472645 \times 10^{-6}$	0.0083858325	$-2.2081839 \times 10^{-9}$
3	2.1284191	2.1293945	2.1284191	$-2.2081839 \times 10^{-9}$	0.0083858325	$-1.5987212 \times 10^{-1}$
4	2.1284191	2.1293945	2.1284191	$-1.5987212 \times 10^{-12}$	0.0083858325	$-5.3290705 \times 10^{-1}$
5	2.1284191	2.1293945	2.1284191	$-5.3290705 \times 10^{-15}$	0.0083858325	$-1.7763568 \times 10^{-1}$
6	2.1284191	2.1293945	2.1284191	$-1.7763568 \times 10^{-15}$	0.0083858325	$-1.7763568 \times 10^{-1}$
7	2.1284191	2.1293945	2.1284191	$-1.7763568 \times 10^{-15}$	0.0083858325	$-1.7763568 \times 10^{-1}$
8	2.1284191	2.1293945	2.1284191	$-1.7763568 \times 10^{-15}$	0.0083858325	$-1.7763568 \times 10^{-1}$
9	2.1284191	2.1293945	2.1284191	$-1.7763568 \times 10^{-15}$	0.0083858325	$-1.7763568 \times 10^{-1}$
10	2.1284191	2.1293945	2.1284191	$-1.7763568 \times 10^{-15}$	0.0083858325	$-1.7763568 \times 10^{-1}$

Root after 10 iterations 2.1284191

accuracy= 0.000487734

Function value at approximated root f[c]= $-1.7763568 \times 10^{-15}$

c) Perform 10 iteration of regular falsi method to find the root of the function: f(x)= x^3 - 2x - 1 in the interval [2,3].

```
In[53]:= f[x_] = x^3 - 2 * x - 6;
a = 2;
b = 3;
n = 10;
Print["The given function is: ", f[x]]
i = 0
c = ((a * f[b]) - (b * f[a])) / (f[b] - f[a])
OutputDetails = {{i, N[a], N[b], N[c], N[f[a]], N[f[b]], N[f[c]]}}
For[i = 1, i ≤ n, i++,
 If[f[a] * f[c] < 0, b = c, a = c];
 c = N[((a * f[b]) - (b * f[a])) / (f[b] - f[a])];
 OutputDetails =
 Append[OutputDetails, {i, N[a], N[b], N[c], N[f[a]], N[f[b]], N[f[c]]}]]
Print[NumberForm[TableForm[OutputDetails,
   TableHeadings → {None, {"i", "a", "b", "c", "f[a]", "f[b]", "f[c]"}}], 8]]
Print["Root after ", 10, " iterations ", NumberForm[N[c], 8]];
Print["accuracy= ", N[Abs[(b - a) / 2]]];
Print["Function value at approximated root f[c]= ", NumberForm[N[f[c]], 8]]
```

The given function is: $-6-2x+x^3$

Out[58]= 0

Out[59]= $\frac{36}{17}$

Out[60]= {{0, 2., 3., 2.11765, -2., 15., -0.738856}}

i	a	b	c	f[a]	f[b]	f[c]
0	2.	3.	2.1176471	-2.	15.	-0.7388561
1	2.1176471	3.	2.1590689	-0.7388561	15.	-0.25346902
2	2.1590689	3.	2.1730427	-0.25346902	15.	-0.084728365
3	2.1730427	3.	2.1776876	-0.084728365	15.	-0.02807642
4	2.1776876	3.	2.1792239	-0.02807642	15.	-0.0092767401
5	2.1792239	3.	2.1797312	-0.0092767401	15.	-0.0030621937
6	2.1797312	3.	2.1798986	-0.0030621937	15.	-0.001010491
7	2.1798986	3.	2.1799539	-0.001010491	15.	-0.00033341634
8	2.1799539	3.	2.1799721	-0.00033341634	15.	-0.00011000852
9	2.1799721	3.	2.1799781	-0.00011000852	15.	-0.000036296173
10	2.1799781	3.	2.1799801	-0.000036296173	15.	-0.000011975501

Root after 10 iterations 2.1799801

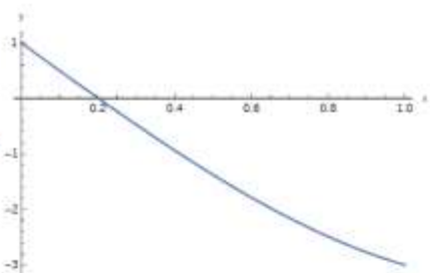
accuracy= 0.410011

Function value at approximated root f[c]= -0.000011975501

d) Find the approximate value of the root of x^3-5x+1 using SECANT METHOD.

In[21]:=
```
f[x_] := x^3 - 5 x + 1;
Plot[f[x], {x, 0, 1}, PlotStyle → {Thick}, AxesLabel → {x, y}]
a = N[0];
b = N[1];
c = (a * f[b] - b * f[a]) / (f[b] - f[a])
i = 1;
output = {{a, b, c, f[a], f[b], f[c]}};
While[i ≤ 6, If[f[c] ≠ 0, {a = b, b = c}]; c = (a * f[b] - b * f[a]) / (f[b] - f[a]);
  i = i + 1;
  output = Append[output, {a, b, c, f[a], f[b], f[c]}]];
Print[NumberForm[TableForm[output,
    TableHeadings → {Automatic, {"a", "b", "c", "f[a]", "f[b]", "f[c]"}}], 8]];
Print["Approximate value of the roots", c];
```

Out[22]=



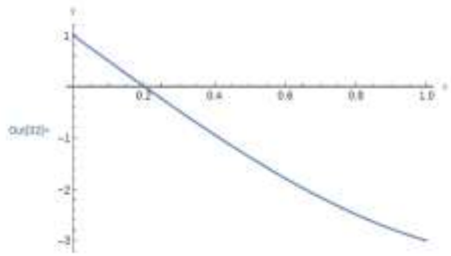
Out[23]= 0.25

	a	b	c	f[a]	f[b]	f[c]
1	0.	1.	0.25	1.	-3.	-0.234375
2	1.	0.25	0.18644068	-3.	-0.234375	0.074277312
3	0.25	0.18644068	0.20173626	-0.234375	0.074277312	-0.00047111617
4	0.18644068	0.20173626	0.20163985	0.074277312	-0.00047111617	-8.642293×10^{-7}
5	0.20173626	0.20163985	0.20163968	-0.00047111617	-8.642293×10^{-7}	$1.0352719 \times 10^{-11}$
6	0.20163985	0.20163968	0.20163968	-8.642293×10^{-7}	$1.0352719 \times 10^{-11}$	$-2.220446 \times 10^{-16}$
7	0.20163968	0.20163968	0.20163968	$1.0352719 \times 10^{-11}$	$-2.220446 \times 10^{-16}$	1.110223×10^{-16}

Approximate value of the roots0.20164

e) Find the approximate value of the root of cos[x]-xe^x using SECANT METHOD.

```
f[x_] := x^3 - 5 x + 1;
Plot[f[x], {x, 0, 1}, PlotStyle → {Thick}, AxesLabel → {x, y}]
a = N[0];
b = N[1];
c = (a * f[b] - b * f[a]) / (f[b] - f[a])
i = 1;
output = {{a, b, c, f[a], f[b], f[c]}};
While[i ≤ 6, If[f[c] ≠ 0, {a = b, b = c}]; c = (a * f[b] - b * f[a]) / (f[b] - f[a]);
  i = i + 1;
  output = Append[output, {a, b, c, f[a], f[b], f[c]}]];
Print[NumberForm[TableForm[output,
    TableHeadings → {Automatic, {"a", "b", "c", "f[a]", "f[b]", "f[c]"}}], 8]];
Print["Approximate value of the roots", c];
```



0.25

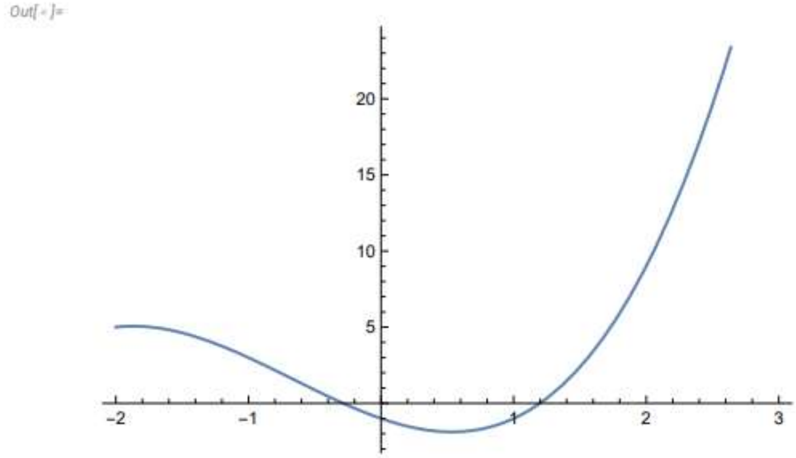
	a	b	c	f[a]	f[b]	f[c]
1	0.	1.	0.25	1.	-3.	-0.234375
2	1.	0.25	0.18644068	-3.	-0.234375	0.074277312
3	0.25	0.18644068	0.20173626	-0.234375	0.074277312	-0.00047111617
4	0.18644068	0.20173626	0.20163985	0.074277312	-0.00047111617	-8.642293×10^{-7}
5	0.20173626	0.20163985	0.20163968	-0.00047111617	-8.642293×10^{-7}	$1.0352719 \times 10^{-11}$
6	0.20163985	0.20163968	0.20163968	-8.642293×10^{-7}	$1.0352719 \times 10^{-11}$	$-2.220446 \times 10^{-16}$
7	0.20163968	0.20163968	0.20163968	$1.0352719 \times 10^{-11}$	$-2.220446 \times 10^{-16}$	1.110223×10^{-16}

Approximate value of the roots0.20164

3) Newton-Raphson Method

a) Perform iteration of Newton Raphson Method to find the root of the functions f(x)=x^3+2x^2-3x-1.

```
In[ ]:= Clear[x, f, a, b, m, n, i]
        f[x_] := x^3 + 2 * x^2 - 3 * x - 1;
        Plot[f[x], {x, -2, 3}]
        IA = 2
        n = 5
        k = N[IA];
        If[f'[k] == 0,
         Print["We cannot continue with the Newton Raphson Method"];
         Abort[]]
        i = 0;
        p = k - f[k] / f'[k];
        OutputDetails = {{i, k, f[k]}};
        While[i < n, p = k - f[k] / f'[k]; k = p; i++;
         OutputDetails = Append[OutputDetails, {i, k, f[k]}];]
        (*Combining the output details with the headings of the table*)
        Print[NumberForm[
          N[TableForm[OutputDetails, TableHeadings → {None, {"i", "ki", "f[ki]"}}]],
          8]] (*Printing Table*)
        Print["Root after ", n, " iteration ", NumberForm[N[k], 8]]
        (*8 is used to give the reslt with 8 digits precision*)
        Print["Function value at approximated root f[x]= ", NumberForm[N[f[k]], 8]];
Out[ ]=
```



Out[=]=

2

Out[=]=

5

i	ki	f[ki]
0.	2.	9.
1.	1.4705882	2.0938327
2.	1.2471327	0.30899704
3.	1.2006987	0.012278977
4.	1.1986949	0.000022485706
5.	1.1986912	7.5904616×10^{-11}

Root after 5 iteration 1.1986912

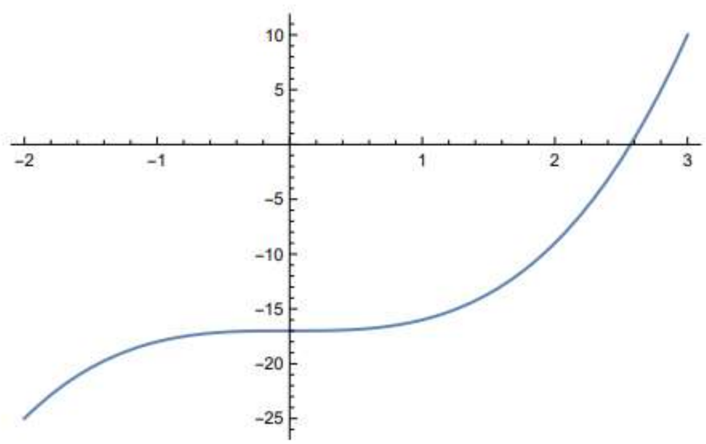
Function value at approximated root f[x] = 7.5904616×10^{-11}

b) Perform 4 iteration of Newton Raphson Method to obtain approximate value of (17)^1/3 starting with the initial approximation 2.

```
In[ ]:= Clear[x, f, a, b, m, n, i]
f[x_] := x^3 - 17;
Plot[f[x], {x, -2, 3}]
IA = 2
n = 5
k = N[IA];

If[f'[k] == 0,
 Print["We cannot continue with the Newton Raphson Method"];
 Abort[]]
i = 0;
p = k - f[k] / f'[k];
OutputDetails = {{i, k, f[k]}};
While[i < n, p = k - f[k] / f'[k]; k = p; i++;
 OutputDetails = Append[OutputDetails, {i, k, f[k]}];]
(*Combining the output details with the headings of the table*)
Print[NumberForm[
  N[TableForm[OutputDetails, TableHeadings → {None, {"i", "ki", "f[ki]"}}]],
  8]] (*Printing Table*)
Print["Root after ", n, " iteration ", NumberForm[N[k], 8]]
(*8 is used to give the reslt with 8 digits precision*)
Print["Function value at approximated root f[x]= ", NumberForm[N[f[k]], 8]];
```

Out[ ]=



Out[ ]=

2

Out[ ]=

5

i	ki	f[ki]
0.	2.	-9.
1.	2.75	3.796875
2.	2.5826446	0.22637726
3.	2.5713315	0.00099018374
4.	2.5712816	1.9223531×10^{-8}
5.	2.5712816	3.5527137×10^{-15}

Root after 5 iteration 2.5712816

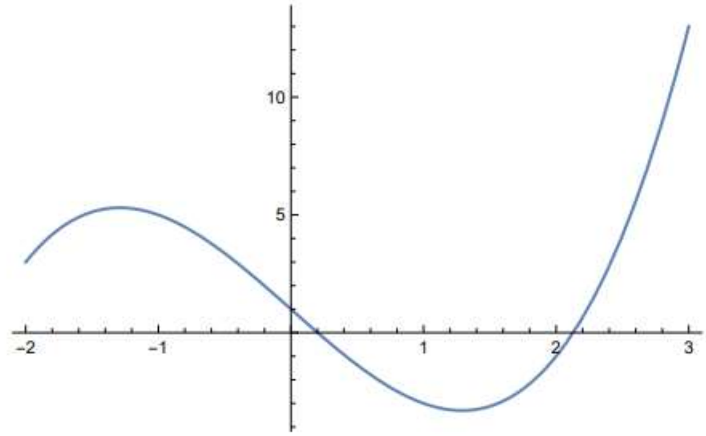
Function value at approximated root f[x]= 3.5527137×10^{-15}

c) Perform 6 iterations of Newton Raphson Method to obtain approximate value of f(x)=x^3-5*x+1 starting with the initial approximation 0.

```
In[ ]:= Clear[x, f, a, b, m, n, i]
        f[x_] := x^3 - 5 * x + 1;
        Plot[f[x], {x, -2, 3}]
        IA = 0
        n = 6
        k = N[IA];

        If[f'[k] == 0,
         Print["We cannot continue with the Newton Raphson Method"];
         Abort[]]
        i = 0;
        p = k - f[k] / f'[k];
        OutputDetails = {{i, k, f[k]}};
        While[i < n, p = k - f[k] / f'[k]; k = p; i++;
         OutputDetails = Append[OutputDetails, {i, k, f[k]}];]
        (*Combining the output details with the headings of the table*)
        Print[NumberForm[
          N[TableForm[OutputDetails, TableHeadings → {None, {"i", "ki", "f[ki]"}}]],
          8]] (*Printing Table*)
        Print["Root after ", n, " iteration ", NumberForm[N[k], 8]]
        (*8 is used to give the reslt with 8 digits precision*)
        Print["Function value at approximated root f[x]= ", NumberForm[N[f[k]], 8]];
Out[ ]=
```

Out[ ]=



Out[ ]=

0

Out[ ]=

6

i	ki	f[ki]
0.	0.	1.
1.	0.2	0.008
2.	0.20163934	1.6168754×10^{-6}
3.	0.20163968	6.6391337×10^{-14}
4.	0.20163968	1.110223×10^{-16}
5.	0.20163968	1.110223×10^{-16}
6.	0.20163968	-2.220446×10^{-16}

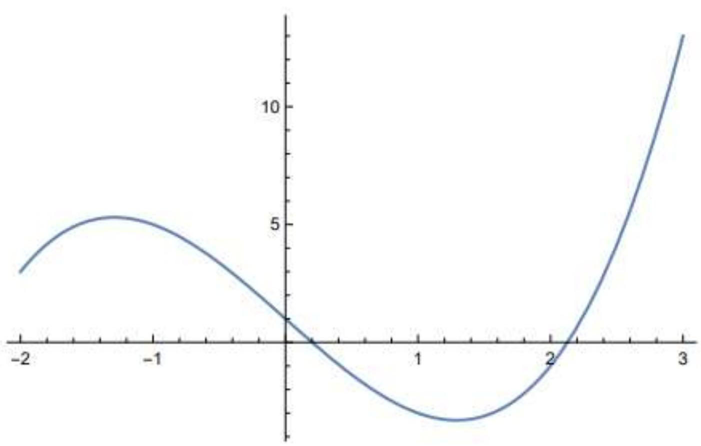
Root after 6 iteration 0.20163968

Function value at approximated root f[x] = -2.220446×10^{-16}

d) Find the root of the function f(x)=x^3-5*x+1. Perform 5 iterations of Newton Raphson Method to find the root of the function starting with initial approximation 0.5,

```
In[ ]:= Clear[x, f, a, b, m, n, i]
f[x_] := x^3 - 5 * x + 1;
Plot[f[x], {x, -2, 3}]
IA = 0.5
n = 5
k = N[IA];
If[f'[k] == 0,
 Print["We cannot continue with the Newton Raphson Method"];
 Abort[]]
i = 0;
p = k - f[k] / f'[k];
OutputDetails = {{i, k, f[k]}};
While[i < n, p = k - f[k] / f'[k]; k = p; i++;
 OutputDetails = Append[OutputDetails, {i, k, f[k]}];]
(*Combining the output details with the headings of the table*)
Print[NumberForm[
  N[TableForm[OutputDetails, TableHeadings → {None, {"i", "ki", "f[ki]"}}]],
  8]] (*Printing Table*)
Print["Root after ", n, " iteration ", NumberForm[N[k], 8]]
(*8 is used to give the reslt with 8 digits precision*)
Print["Function value at approximated root f[x]= ", NumberForm[N[f[k]], 8]];
Out[ ]=
```

Out[ ]=



Out[ ]=

```
0.5
```

Out[ ]=

```
5
```

i	ki	f[ki]
0.	0.5	-1.375
1.	0.17647059	0.12314268
2.	0.20156807	0.0003492764
3.	0.20163968	3.1004843×10^{-9}
4.	0.20163968	1.110223×10^{-16}
5.	0.20163968	1.110223×10^{-16}

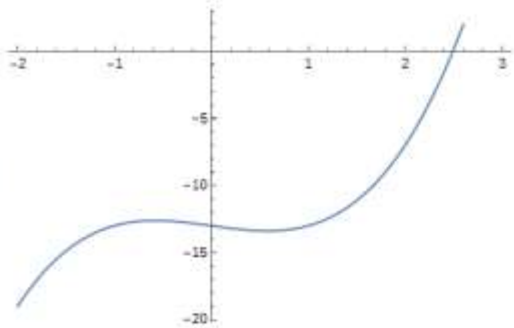
Root after 5 iteration 0.20163968

Function value at approximated root f[x] = 1.110223×10^{-16}

d) Find the root of the function f(x)=x^3-x-13. Perform 6 iterations of Newton Raphson Method to find the root of the function starting with initial approximation 0.

```
f[x_] := x^3 - x - 13;
Plot[f[x], {x, -2, 3}]
IA = 0
n = 6
k = N[IA];
If[f'[k] == 0,
 Print["We cannot continue with the Newton Raphson Method"];
 Abort[]]
i = 0;
p = k - f[k] / f'[k];
OutputDetails = {{i, k, f[k]}};
While[i < n, p = k - f[k] / f'[k]; k = p; i++;
 OutputDetails = Append[OutputDetails, {i, k, f[k]}]; ] (* Combining the output details with the headings of the table *)
Print[NumberForm[ N[TableForm[OutputDetails,
     TableHeadings -> {None, {"i", "ki", "f[ki]"}}]], 8]] (* Printing Table *)
Print["Root after ", n, " iteration ", NumberForm[N[k], 8]] (* 8 is used to give the reslt with 8 digits precision *)
Print["Function value at approximated root f[x]= ", NumberForm[N[f[k]], 8]];
```

Out[92]=



Out[93]= 0

Out[94]= 6

i	ki	f[ki]
0.	0.	-13.
1.	-13.	-2197.
2.	-8.6581028	-653.37703
3.	-5.7397849	-196.35817
4.	-3.7327588	-61.277593
5.	-2.2308742	-21.871739
6.	-0.66080141	-12.627743

Root after 6 iteration -0.66080141

Function value at approximated root f[x]= -12.627743

4) Gaussian elimination method and Gauss-Jordan method

a) Solve the following system of equations by Guass Jordan Method:

x1+x2-x3=9

x2+3x2=3

-x1-2x3=2.

```
In[ ]:= Clear[A, b, aug, matrix, lhs, rhs]

In[ ]:= A = {{1, 1, -1}, {0, 1, 3}, {-1, 0, -2}};
        b = {9, 3, 2};
        aug = Transpose[Append[Transpose[A], b]];
        aug[[3]] = aug[[1]] + aug[[3]];
        aug[[3]] = aug[[3]] - aug[[2]];
        aug[[1]] = aug[[1]] - aug[[2]]
        aug[[3]] = -aug[[3]] / 6
        aug[[1]] = aug[[1]] + 4 * aug[[3]]
        aug[[2]] = aug[[2]] - 3 * aug[[3]]
```

```
Out[ ]=
        {1, 0, -4, 6}
Out[ ]=
        {0, 0, 1, -4/3}
Out[ ]=
        {1, 0, 0, 2/3}
Out[ ]=
        {0, 1, 0, 7}

In[ ]:= matrix = {aug[[1]], aug[[2]], aug[[3]]}
Out[ ]=
        {{1, 0, 0, 2/3}, {0, 1, 0, 7}, {0, 0, 1, -4/3}}
```

```
In[ ]:= MatrixForm[matrix]
```

Out[ ]//MatrixForm=

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

```
In[ ]:= lhs = matrix[[All, {1, 2, 3}]]
        rhs = matrix[[All, 4]]
```

Out[ ]=

```
{{1, 0, 0}, {0, 1, 0}, {0, 0, 1}}
```

Out[ ]=

$$\left\{\frac{2}{3}, 7, -\frac{4}{3}\right\}$$

```
In[ ]:= LinearSolve[lhs, rhs]
```

Out[ ]=

$$\left\{\frac{2}{3}, 7, -\frac{4}{3}\right\}$$

b) Solve the following system of equations by Guass Elimination Method with pivoting:

2x1+6x2+10x3=0

X1+3x2+3x3=2

3x1+14x2+28x3=-8

```
In[ ]:= Clear[A, x, b, c, aug]
        A = {{2, 6, 10}, {1, 3, 3}, {3, 14, 28}}
        A // MatrixForm
Out[ ]=
        {{2, 6, 10}, {1, 3, 3}, {3, 14, 28}}
Out[ ]//MatrixForm=
```

$$\begin{pmatrix} 2 & 6 & 10 \\ 1 & 3 & 3 \\ 3 & 14 & 28 \end{pmatrix}$$

```
In[ ]:= x = {x1, x2, x3}
        MatrixForm[x]
Out[ ]=
        {x1, x2, x3}
Out[ ]//MatrixForm=
```

$$\begin{pmatrix} x1 \\ x2 \\ x3 \end{pmatrix}$$

```
In[ ]:= b = {{0}, {2}, {-8}}
        b // MatrixForm
Out[ ]=
        {{0}, {2}, {-8}}
Out[ ]//MatrixForm=
```

$$\begin{pmatrix} 0 \\ 2 \\ -8 \end{pmatrix}$$

```
In[ ]:= aug = ArrayFlatten[{{A, b}}];
        aug // MatrixForm
Out[ ]//MatrixForm=
```

$$\begin{pmatrix} 2 & 6 & 10 & 0 \\ 1 & 3 & 3 & 2 \\ 3 & 14 & 28 & -8 \end{pmatrix}$$

```
In[ ]:= Max[Abs[Take[aug, 3, 1]]]
Out[ ]=
        3
```

```
In[ ]:= row1 = aug[[3]];
        row2 = aug[[1]];
        aug[[1]] = row1;
        aug[[3]] = row2;
        aug // MatrixForm
Out[ ]//MatrixForm=
```

$$\begin{pmatrix} 3 & 14 & 28 & -8 \\ 1 & 3 & 3 & 2 \\ 2 & 6 & 10 & 0 \end{pmatrix}$$

In[ ]:=
```
aug[[2]] = aug[[2]] - (1 / 3) aug[[1]];
aug[[3]] = aug[[3]] - (2 / 3) aug[[1]];
aug // MatrixForm
```

Out[ ]//MatrixForm=

$$\begin{pmatrix} 3 & 14 & 28 & -8 \\ 0 & -\frac{5}{3} & -\frac{19}{3} & \frac{14}{3} \\ 0 & -\frac{10}{3} & -\frac{26}{3} & \frac{16}{3} \end{pmatrix}$$

In[ ]:=
```
Max[Abs[Take[aug, {2, 3}, {2, 2}]]]
```

Out[ ]=

$$\frac{10}{3}$$

In[ ]:=
```
r1 = aug[[2]];
r2 = aug[[3]];
aug[[2]] = r2;
aug[[3]] = r1;
aug // MatrixForm
```

Out[ ]//MatrixForm=

$$\begin{pmatrix} 3 & 14 & 28 & -8 \\ 0 & -\frac{10}{3} & -\frac{26}{3} & \frac{16}{3} \\ 0 & -\frac{5}{3} & -\frac{19}{3} & \frac{14}{3} \end{pmatrix}$$

In[ ]:=
```
aug[[3]] = aug[[3]] - (aug[[3, 2]] / aug[[2]]) aug[[2]];
aug // MatrixForm
```

Out[ ]//MatrixForm=

$$\begin{pmatrix} 3 & 14 & 28 & -8 \\ 0 & -\frac{10}{3} & -\frac{26}{3} & \frac{16}{3} \\ \text{Indeterminate} & 0 & -\frac{14}{3} & \frac{19}{3} \end{pmatrix}$$

In[ ]:=

```
U = Take[aug, 3, 3];
U // MatrixForm
```

Out[ ]//MatrixForm=

$$\begin{pmatrix} 3 & 14 & 28 \\ 0 & -\frac{10}{3} & -\frac{26}{3} \\ \text{Indeterminate} & 0 & -\frac{14}{3} \end{pmatrix}$$

In[ ]:=

```
H = Take[aug, 3, -1];
H // MatrixForm
```

Out[ ]//MatrixForm=

$$\begin{pmatrix} -8 \\ \frac{16}{3} \\ \frac{19}{3} \end{pmatrix}$$

In[ ]:=

```
U.x == H
```

Out[ ]=

$$\left\{3\,x1 + 14\,x2 + 28\,x3,\ -\frac{10\,x2}{3} - \frac{26\,x3}{3},\ \text{Indeterminate}\right\} == \left\{\{-8\},\ \left\{\frac{16}{3}\right\},\ \left\{\frac{19}{3}\right\}\right\}$$

In[ ]:=

c)Solve the following system of the equation (without partial pivoting):

x1 + 2x2 + 3x3 =1

2x1 + 6x2+ 10x3=0

3x1 + 14x2 + 28 x3 = -8

In[ ]:=
```
Clear[A]
A = {{1, 2, 3}, {2, 6, 10}, {3, 14, 28}};
A // MatrixForm
```
Out[ ]//MatrixForm=

$$\begin{pmatrix} 1 & 2 & 3 \\ 2 & 6 & 10 \\ 3 & 14 & 28 \end{pmatrix}$$

In[ ]:=
```
x = {x1, x2, x3};
MatrixForm[x]
```
Out[ ]//MatrixForm=

$$\begin{pmatrix} x1 \\ x2 \\ x3 \end{pmatrix}$$

In[ ]:=
```
b = {{1}, {0}, {-8}};
b // MatrixForm
```
Out[ ]//MatrixForm=

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

In[ ]:=
```
aug = ArrayFlatten[{{A, b}}];
aug // MatrixForm
```
Out[ ]//MatrixForm=

$$\begin{pmatrix} 1 & 2 & 3 & 1 \\ 2 & 6 & 10 & 0 \\ 3 & 14 & 28 & -8 \end{pmatrix}$$

In[ ]:=
```
aug[[2]] = aug[[2]] - 2 aug[[1]];
aug[[3]] = aug[[3]] - 3 aug[[1]];
aug // MatrixForm
```
Out[ ]//MatrixForm=

$$\begin{pmatrix} 1 & 2 & 3 & 1 \\ 0 & 2 & 4 & -2 \\ 0 & 8 & 19 & -11 \end{pmatrix}$$

```
In[ ]:= aug[[3]] = aug[[3]] - 4 aug[[2]];
        aug // MatrixForm
Out[ ]//MatrixForm=
```

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

```
In[ ]:= upper = Take[aug, 3, 3];
        upper // MatrixForm
Out[ ]//MatrixForm=
```

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

```
In[ ]:= c = Take[aug, 3, -1];
        c // MatrixForm
Out[ ]//MatrixForm=
```

$$\begin{pmatrix} 1 \\ -2 \\ -3 \end{pmatrix}$$

```
In[ ]:= upper.x == c
Out[ ]=
        {x1 + 2 x2 + 3 x3, 2 x2 + 4 x3, 3 x3} == {{1}, {-2}, {-3}}

In[ ]:= Solve[upper.x == c]
Out[ ]=
        {{x1 -> 2, x2 -> 1, x3 -> -1}}
```

d) Solve the following system of equations by Guass Jordan Method:

2x1+3x2+x3=-1

3x1+2x2+2x2=1

x1+2x2+2x3=6.

PracticalJOrdand.nb

```
In[55]:= Clear[A, b, aug, matrix, lhs, rhs]
         A = {{2, 3, 1}, {3, 2, 2}, {1, 2, 2}};
         b = {-1, 1, 6};
         aug = Transpose[Append[Transpose[A], b]];
         aug[[2]] = aug[[2]] - (3 / 2) * aug[[1]];
         aug[[3]] = aug[[3]] - (1 / 2) * aug[[1]];
         aug[[1]] = aug[[1]] + (6 / 5) * aug[[2]] ;
         aug[[3]] = aug[[3]] + (1 / 5) * aug[[2]] ;
         aug[[1]] = aug[[1]] - aug[[3]];
         aug[[2]] = aug[[2]] - (5 / 16) * aug[[3]]
         aug[[1]] = aug[[1]] / 2;
         aug[[2]] = aug[[2]] * (-2 / 5);
         aug[[3]] = aug[[3]] * (5 / 8);
         matrix = {aug[[1]], aug[[2]], aug[[3]]}
         MatrixForm[matrix]
         lhs = matrix[[All, {1, 2, 3}]]
         rhs = matrix[[All, 4]]
         LinearSolve[lhs, rhs]
```

Out[64]= $\left\{0, -\frac{5}{2}, 0, \frac{5}{16}\right\}$

Out[68]= $\left\{\left\{1, 0, 0, -\frac{5}{2}\right\}, \left\{0, 1, 0, -\frac{1}{8}\right\}, \left\{0, 0, 1, \frac{35}{8}\right\}\right\}$

Out[69]//MatrixForm=

$$\begin{pmatrix} 1 & 0 & 0 & -\frac{5}{2} \\ 0 & 1 & 0 & -\frac{1}{8} \\ 0 & 0 & 1 & \frac{35}{8} \end{pmatrix}$$

Out[70]= {{1, 0, 0}, {0, 1, 0}, {0, 0, 1}}

Out[71]= $\left\{-\frac{5}{2}, -\frac{1}{8}, \frac{35}{8}\right\}$

Out[72]= $\left\{-\frac{5}{2}, -\frac{1}{8}, \frac{35}{8}\right\}$

e) Solve the following system of equations by Guass Elimination Method with pivoting:

2x1+3x2+x3=-1

3x1+2x2+2x2=1

x1+2x2+2x3=6.

```
In[107]:= Clear[A, b, aug, matrix, lhs, rhs]
          A = {{2, 3, 1}, {3, 2, 2}, {1, 2, 2}};
          A // MatrixForm
```

Out[109]//MatrixForm=

$$\begin{pmatrix} 2 & 3 & 1 \\ 3 & 2 & 2 \\ 1 & 2 & 2 \end{pmatrix}$$

```
In[110]:= x = {x1, x2, x3};
          MatrixForm[x]
```

Out[111]//MatrixForm=

$$\begin{pmatrix} x1 \\ x2 \\ x3 \end{pmatrix}$$

```
In[112]:= b = {{2}, {4}, {6}};
          b // MatrixForm
```

Out[113]//MatrixForm=

$$\begin{pmatrix} 2 \\ 4 \\ 6 \end{pmatrix}$$

```
In[114]:= aug = ArrayFlatten[{{A, b}}];
          aug // MatrixForm
```

Out[115]//MatrixForm=

$$\begin{pmatrix} 2 & 3 & 1 & 2 \\ 3 & 2 & 2 & 4 \\ 1 & 2 & 2 & 6 \end{pmatrix}$$

```
In[116]:= Max[Abs[Take[aug, 3, 1]]]
Out[116]= 3
```

```
In[117]:= row = aug[[1]];
          row2 = aug[[2]];
          aug[[1]] = row2;
          aug[[2]] = row;
          aug // MatrixForm
```

Out[121]//MatrixForm=

$$\begin{pmatrix} 3 & 2 & 2 & 4 \\ 2 & 3 & 1 & 2 \\ 1 & 2 & 2 & 6 \end{pmatrix}$$

In[122]:=
```
aug[[2]] = aug[[2]] - (2/3) * aug[[1]];
aug[[3]] = aug[[3]] - (1/3) * aug[[1]];
aug // MatrixForm
```

Out[124]//MatrixForm=

$$\begin{pmatrix} 3 & 2 & 2 & 4 \\ 0 & \frac{5}{3} & -\frac{1}{3} & -\frac{2}{3} \\ 0 & \frac{4}{3} & \frac{4}{3} & \frac{14}{3} \end{pmatrix}$$

In[125]:=
```
Max[Abs[Take[aug, {2, 3}, {2, 2}]]]
```

Out[125]= $\frac{5}{3}$

In[126]:=
```
aug[[3]] = aug[[3]] - (4/5) * aug[[2]];
aug // MatrixForm
```

Out[127]//MatrixForm=

$$\begin{pmatrix} 3 & 2 & 2 & 4 \\ 0 & \frac{5}{3} & -\frac{1}{3} & -\frac{2}{3} \\ 0 & 0 & \frac{8}{5} & \frac{26}{5} \end{pmatrix}$$

In[128]:=
```
U = Take[aug, 3, 3]
U // MatrixForm
```

Out[128]= $\left\{\{3, 2, 2\}, \left\{0, \frac{5}{3}, -\frac{1}{3}\right\}, \left\{0, 0, \frac{8}{5}\right\}\right\}$

Out[129]//MatrixForm=

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

In[130]:=
```
H = Take[aug, 3, -1]
H // MatrixForm
```

Out[130]= $\left\{\{4\}, \left\{-\frac{2}{3}\right\}, \left\{\frac{26}{5}\right\}\right\}$

Out[131]//MatrixForm=

$$\begin{pmatrix} 4 \\ -\frac{2}{3} \\ \frac{26}{5} \end{pmatrix}$$

In[132]:=
```
U.x == H
```

Out[132]= $\left\{3\,x1 + 2\,x2 + 2\,x3, \frac{5\,x2}{3} - \frac{x3}{3}, \frac{8\,x3}{5}\right\} == \left\{\{4\}, \left\{-\frac{2}{3}\right\}, \left\{\frac{26}{5}\right\}\right\}$

5) Jacobi Method and Gauss-Seidel Method

a) Solve the system of equation by performing 10 iterations of the guass siedal iterative method with initial approximation x0 = [0 0 0]^T.

5x1 + x2 + 2x3 = 10

-3x1+9x2 +4x3= -14

x1 + 2x2 - 7x3= -33.

```
In[ ]:= A = {{5, 1, 2}, {-3, 9, 4}, {1, 2, -7}}
b = {{10}, {-14}, {-33}}
x0 = {0, 0, 0}
max = 10
k = 0;
Size = Dimensions[A];
m = Size[[1]];
n = Size[[2]];
xk = x0; (*xk*);
If[m ≠ n, Print["Not a square matrix, so we cannot proceed"];
   Return[]];
OutputDetails = {xk};
xk1 = Table[0, {m}]; (*xk+1*)
While[k < max, For[i = 1, i ≤ m, i++,
    xk1[[i]] = (1 / A[[i, i]]) (b[[i]] -
         Sum[A[[i, j]] * xk1[[j]], {j, 1, i-1}] - Sum[A[[i, j]] * xk[[j]], {j, i+1, m}]);];
  k++; OutputDetails = Append[OutputDetails, xk1]; xk = xk1;];
ColumnHeading = Table[x[j], {j, 1, m}];
Print[NumberForm[
   N[TableForm[OutputDetails, TableHeadings → {None, ColumnHeading}]], 7]];
Print["Number of iterations performed= ", max]
Print["Approximate solution after", max, " iterations is = ", NumberForm[N[xk1], 7]];

Out[ ]= {{5, 1, 2}, {-3, 9, 4}, {1, 2, -7}}

Out[ ]= {{10}, {-14}, {-33}}

Out[ ]= {0, 0, 0}

Out[ ]= 10
```

x[1.]	x[2.]	x[3.]
0.	0.	0.
2.	-0.8888889	4.746032
0.2793651	-3.571781	3.733686
1.220882	-2.808011	4.086409
0.9270388	-3.062724	3.971656
1.023883	-2.979442	4.009286
0.9921741	-3.006736	3.996958
1.002564	-2.997793	4.000997
0.9991599	-3.000723	3.999673
1.000275	-2.999763	4.000107
0.9999098	-3.000078	3.999965

Number of iterations performed= 10

Approximate solution after10 iterations is = {{0.9999098}, {-3.000078}, {3.999965}}

b) Perform 14 iterations up to 8 decimal plaaces to solve the following system of linear equations with x[0]=0 by siedal;

x+2y-2z=11

x+y-z=0

2x+2y+z=3

```
x = 0.0; y = 0.0; z = 0.0;
i = 1; output = {{x, y, z}};
While[i ≤ 14, {x = (11 - 2 y + 2 z), y = (z - x), z = (3 - 2 y - 2 x)};
  i = i + 1;
  output = Append[output, {x, y, z}]];
Print[
 NumberForm[TableForm[output, TableHeadings → {Automatic, {"x", "y", "z"}}], 8]]
```

	x	y	z
1	0.	0.	0.
2	11.	-11.	3.
3	39.	-36.	-3.
4	77.	-80.	9.
5	189.	-180.	-15.
6	341.	-356.	33.
7	789.	-756.	-63.
8	1397.	-1460.	129.
9	3189.	-3060.	-255.
10	5621.	-5876.	513.
11	12789.	-12276.	-1023.
12	22517.	-23540.	2049.
13	51189.	-49140.	-4095.
14	90101.	-94196.	8193.
15	204789.	-196596.	-16383.

c) Perform 14 iterations up to 8 decimal plaaces to solve the following system of linear equations with x[0]=0; GUASS JACOBI

5x+y+2z=10

-3x+9y+4z=-14

X+2y-7z=-33

```
x1 = 0.0;
x2 = 0.0;
x3 = 0.0;
i = 1; output = {{x1, x2, x3}};
While[i ≤ 14, {x = (10 - x2 - 2 x3) / 5, y = (-14 + 3 x1 - 4 x3) / 9, z = (-33 - x1 - 2 x2) / -7};
  i = i + 1;
  {x1 = x, x2 = y, x3 = z};
  output = Append[output, {x1, x2, x3}]];
Print[
 NumberForm[TableForm[output, TableHeadings → {Automatic, {"x", "y", "z"}}], 8]]
Print["The approximate value of x=", x, ",y=", y, " and z=", z]
```

	x	y	z
1	0.	0.	0.
2	2.	-1.5555556	4.7142857
3	0.42539683	-2.984127	4.5555556
4	0.77460317	-3.438448	3.922449
5	1.11871	-3.0406652	3.8425296
6	1.0711212	-2.8904432	4.00534
7	0.97595265	-2.9786663	4.0414621
8	0.9791484	-3.0264434	4.00266
9	1.0042247	-3.0081328	3.9894659
10	1.0058402	-2.99391	3.9982799
11	0.99947004	-2.9972888	4.0025743
12	0.99842803	-3.0013208	4.0006989
13	0.99998459	-3.0008346	3.9993981
14	1.0004077	-2.9997376	3.9997593
15	1.0000438	-2.9997571	4.0001332

The approximate value of x=1.00004,y=-2.99976 and z=4.00013

d) Perform 14 iterations up to 8 decimal plaaces to solve the following system of linear equations with x[0]=0; GUASS JACOBI

4x-y=0

2x+4y-z=2

-2y+4z-w=-3

-2z-4w=1

```
x1 = 0.0;
x2 = 0.0;
x3 = 0.0;
x4 = 0.0;
i = 1; output = {{x1, x2, x3, x4}};
While[i ≤ 14,
  {x = (x2) / 4, y = (2 - 2 x1 + x3) / 4, z = (-3 + 2 x2 + x4) / 4, w = (1 + 2 x3) / -4};
  i = i + 1;
  {x1 = x, x2 = y, x3 = z, x4 = w};
  output = Append[output, {x1, x2, x3, x4}]];
Print[
 NumberForm[TableForm[output, TableHeadings → {Automatic, {"x", "y", "z", "w"}}], 8]]
Print["The approximate value of x=", x, ",y=", y, ",z=", z, "w=", w]
```

	x	y	z	w
1	0.	0.	0.	0.
2	0.	0.5	-0.75	-0.25
3	0.125	0.3125	-0.5625	0.125
4	0.078125	0.296875	-0.5625	0.03125
5	0.07421875	0.3203125	-0.59375	0.03125
6	0.080078125	0.31445313	-0.58203125	0.046875
7	0.078613281	0.31445313	-0.58105469	0.041015625
8	0.078613281	0.31542969	-0.58251953	0.040527344
9	0.078857422	0.31506348	-0.58215332	0.041259766
10	0.078765869	0.31503296	-0.58215332	0.04107666
11	0.07875824	0.31507874	-0.58221436	0.04107666
12	0.078769684	0.31506729	-0.58219147	0.041107178
13	0.078766823	0.31506729	-0.58218956	0.041095734
14	0.078766823	0.3150692	-0.58219242	0.04109478
15	0.0787673	0.31506848	-0.58219171	0.04109621

```
The approximate value of x=0.0787673,y=0.315068,z=-0.582192w=0.0410962
```

6) Lagrange Interpolation and Newton Interpolation

a) SOLVE THE GIVEN PROBLEM BY USING LAGRANGE INTERPOLATION AT X = 2.8 AND X = 3.8 ,

GIVEN DATA:

X	2	3	4
F (X)	1.4142	1.7321	2

```
In[140]:=
LagrangePoly[x0_, f0_] :=
 Module[{xi = x0, fi = f0, n, m, polynomial},
  n = Length[xi];
  m = Length [fi];
  If[n ≠ m, Print ["Polynomail can not be foundout"]; Return [];];
  For[i = 1, i ≤ n, i++,
   L[i, x_] = (Product[(x - xi[[j]]) / (xi[[i]] - xi[[j]]), {j, 1, i - 1}]) *
      (Product[(x - xi[[j]]) / (xi[[i]] - xi[[j]]), {j, i + 1, n}]);];
  polynomial[x_] = Sum[L[k, x] * fi[[k]], {k, 1, n}];
  Return[polynomial[x]];]
data = {2, 3, 4};
fun = {1.4142, 1.7321, 2}
Lagpoly[x_] = LagrangePoly[data, fun];
poly = Simplify[Lagpoly[2.8]]
poly = Simplify[Lagpoly[3.8]]

Out[142]=
{1.4142, 1.7321, 2}

Out[144]=
1.67252

Out[145]=
1.95042
```

b) SOLVE THE GIVEN PROBLEM BY USING LAGRANGE INTERPOLATION AT X =1,

GIVEN DATA:

X	1	2	3
F (X)	2	5	10

```
In[143]:= LagrangePoly[x0_, f0_] :=
     Module[{xi = x0, fi = f0, n, m, polynomial},
      n = Length[xi];
      m = Length [fi];
      If[n ≠ m, Print ["Polynomail can not be foundout"]; Return []; ];
      For[i = 1, i ≤ n, i++,
       L[i, x_] = (Product[(x - xi[[j]]) / (xi[[i]] - xi[[j]]), {j, 1, i - 1}]) *
       (Product[(x - xi[[j]]) / (xi[[i]] - xi[[j]]), {j, i + 1, n}]); ];
      polynomial[x_] = Sum[L[k, x] * fi[[k]], {k, 1, n}];
      Return[polynomial[x]]; ]
     data = {1, 2, 3};
     fun = {2, 5, 10}
     Lagpoly[x_] = LagrangePoly[data, fun];
     poly = Simplify[Lagpoly[1]]

Out[145]= {2, 5, 10}

Out[147]= {1 + x1^2, 1 + x2^2, 1 + x3^2}
```

```
In[143]:= LagrangePoly[x0_, f0_] :=
 Module[{xi = x0, fi = f0, n, m, polynomial},
  n = Length[xi];
  m = Length [fi];
  If[n ≠ m, Print ["Polynomail can not be foundout"]; Return []; ];
  For[i = 1, i ≤ n, i++,
   L[i, x_] = (Product[(x - xi[[j]]) / (xi[[i]] - xi[[j]]), {j, 1, i - 1}]) *
   (Product[(x - xi[[j]]) / (xi[[i]] - xi[[j]]), {j, i + 1, n}]); ];
  polynomial[x_] = Sum[L[k, x] * fi[[k]], {k, 1, n}];
  Return[polynomial[x]]; ]
data = {1, 2, 3};
fun = {2, 5, 10}
Lagpoly[x_] = LagrangePoly[data, fun];
poly = Simplify[Lagpoly[1]]

Out[145]= {2, 5, 10}

Out[147]= {1 + x1², 1 + x2², 1 + x3²}
```

c) SOLVE THE GIVEN PROBLEM BY USING NEWTON INTERPOLATION

GIVEN DATA:

X	-7	-5	-4	-1
F (X)	10	5	2	10

In[19]:=

```
divideddiff[x0_, f0_, sv_, ev_] :=
 Module[{x = x0, f = f0, i = sv, j = ev, ans},
 If[i == j, Return[f[[i]]],
  ans =
     (divideddiff[x, f, i + 1, j] - divideddiff[x, f, i, j - 1]) / (x[[j]] - x[[i]]);
  Return[ans]];];

x = {-7, -5, -4, -1};
f = {10, 5, 2, 10};
divideddiff[x, f, 1, 4]
newtonpoly[x0_, f0_] :=
 Module[{x1 = x0, f1 = f0, n, np, k, j},
  n = Length[x1];
  np[y_] = 0;
  For[i = 1, i ≤ n, i++,
   prod[y_] = 1;
   For[k = 1, k ≤ i - 1, k++,
    prod[y_] = prod[y] * (y - x1[[k]])];
   np[y_] = np[y] + divideddiff[x1, f, 1, i] * prod[y]];
  Return[np[y]];];
x = {-7, -5, -4, -1};
f = {10, 5, 2, 10};
poly[y_] = Simplify[newtonpoly[x, f]]
```

Out[22]=

$$\frac{19}{72}$$

Out[26]=

$$\frac{1}{72}\left(1700 + 1253\,y + 292\,y^2 + 19\,y^3\right)$$

d) SOLVE THE GIVEN PROBLEM BY USING NEWTON INTERPOLATION

GIVEN DATA:

X	-8	-5	-3	1
F (X)	12	5	4	1

```
In[27]:= divideddiff[x0_, f0_, sv_, ev_] :=
  Module[{x = x0, f = f0, i = sv, j = ev, ans},
  If[i == j, Return[f[[i]]],
   ans =
      (divideddiff[x, f, i + 1, j] - divideddiff[x, f, i, j - 1]) / (x[[j]] - x[[i]]);
   Return[ans]];];

x = {-8, -5, -3, 1};
f = {12, 5, 4, 1};
divideddiff[x, f, 1, 4]
newtonpoly[x0_, f0_] :=
 Module[{x1 = x0, f1 = f0, n, np, k, j},
  n = Length[x1];
  np[y_] = 0;
  For[i = 1, i ≤ n, i++,
   prod[y_] = 1;
   For[k = 1, k ≤ i - 1, k++,
    prod[y_] = prod[y] * (y - x1[[k]])];
   np[y_] = np[y] + divideddiff[x1, f, 1, i] * prod[y]];
  Return[np[y]];];
x = {-8, -5, -3, 1};
f = {12, 5, 4, 1};
poly[y_] = Simplify[newtonpoly[x, f]]
```

Out[30]=

$$-\frac{49}{1080}$$

Out[34]=

$$\frac{2760 - 1243\, y - 388\, y^2 - 49\, y^3}{1080}$$

e) SOLVE THE GIVEN PROBLEM BY USING NEWTON INTERPOLATION

GIVEN DATA:

X	10	3	-7	-4	-5
F (X)	20	5	4	8	0

```
In[35]:= divideddiff[x0_, f0_, sv_, ev_] :=
          Module[{x = x0, f = f0, i = sv, j = ev, ans},
          If[i == j, Return[f[[i]]],
           ans =
               (divideddiff[x, f, i + 1, j] - divideddiff[x, f, i, j - 1]) / (x[[j]] - x[[i]]);
           Return[ans]];];

         x = {10, 3, -7, -4, -5};
         f = {20, 5, 4, 8, 0};
         divideddiff[x, f, 1, 4]
         newtonpoly[x0_, f0_] :=
          Module[{x1 = x0, f1 = f0, n, np, k, j},
           n = Length[x1];
           np[y_] = 0;
           For[i = 1, i ≤ n, i++,
            prod[y_] = 1;
            For[k = 1, k ≤ i - 1, k++,
             prod[y_] = prod[y] * (y - x1[[k]])];
            np[y_] = np[y] + divideddiff[x1, f, 1, i] * prod[y]];
           Return[np[y]];];
         x = {10, 3, -7, -4, -5};
         f = {20, 5, 4, 8, 0};
         poly[y_] = Simplify[newtonpoly[x, f]]
```

Out[38]=

$$\frac{529}{24990}$$

Out[42]=

$$\frac{6472200 - 350450\,y - 522333\,y^2 - 8026\,y^3 + 6129\,y^4}{199920}$$

f) SOLVE THE GIVEN PROBLEM BY USING NEWTON INTERPOLATION

GIVEN DATA:

X	1	2	3	4	5
F (X)	10	20	30	40	50

```
In[43]:= divideddiff[x0_, f0_, sv_, ev_] :=
          Module[{x = x0, f = f0, i = sv, j = ev, ans},
          If[i == j, Return[f[[i]]],
           ans =
               (divideddiff[x, f, i + 1, j] - divideddiff[x, f, i, j - 1]) / (x[[j]] - x[[i]]);
           Return[ans]];];

         x = {1, 2, 3, 4, 5};
         f = {10, 20, 30, 40, 50};
         divideddiff[x, f, 1, 4]
         newtonpoly[x0_, f0_] :=
          Module[{x1 = x0, f1 = f0, n, np, k, j},
           n = Length[x1];
           np[y_] = 0;
           For[i = 1, i ≤ n, i++,
            prod[y_] = 1;
            For[k = 1, k ≤ i - 1, k++,
             prod[y_] = prod[y] * (y - x1[[k]])];
            np[y_] = np[y] + divideddiff[x1, f, 1, i] * prod[y]];
           Return[np[y]];];
         x = {1, 2, 3, 4, 5};
         f = {10, 20, 30, 40, 50};
         poly[y_] = Simplify[newtonpoly[x, f]]
Out[46]=
         0

Out[50]=
         10 y
```

7) Trapezoid and Simpson's rule.

a) Find the approximate integration of f(x)=1/(1+x) on [0,1] by trapezoid rule.

```
In[1]:= f[x_] = 1 / (1 + x);
        a = 0;
        b = 1;
        APPINT = ((b - a) / 2) * (f[a] + f[b]);
        EXACTINT = N[Integrate[f[x], {x, a, b}]];
        Error = Abs[APPINT - EXACTINT];
        Print["APPINT= ", N[APPINT]]
        Print["EXACTINT= ", EXACTINT]
        Print["Errors ", Error]

        APPINT= 0.75

        EXACTINT= 0.693147

        Errors 0.0568528
```

b) Find the approximate integration of f(x)=1/(1+x^2) on [0,1] by trapezoid rule.

```
In[10]:= f[x_] = 1 / (1 + x^2);
         a = 0;
         b = 1;
         APPINT = ((b - a) / 2) * (f[a] + f[b]);
         EXACTINT = N[Integrate[f[x], {x, a, b}]];
         Error = Abs[APPINT - EXACTINT];
         Print["APPINT= ", N[APPINT]]
         Print["EXACTINT= ", EXACTINT]
         Print["Errors ", Error]

         APPINT= 0.75

         EXACTINT= 0.785398

         Errors 0.0353982
```

c) Find the approximate integration of f(x)=e^x on [0,1] by trapezoid rule.

```
In[19]:= f[x_] = Exp[x];
a = 0;
b = 1;
APPINT = ((b - a) / 2) * (f[a] + f[b]);
EXACTINT = N[Integrate[f[x], {x, a, b}]];
Error = Abs[APPINT - EXACTINT];
Print["APPINT= ", N[APPINT]]
Print["EXACTINT= ", EXACTINT]
Print["Errors ", Error]

APPINT= 1.85914

EXACTINT= 1.71828

Errors 0.140859
```

d) Using Simpson rule, find the approximate integration f(x)=1/1+x on [0,1].

```
In[64]:= f[x_] = 1 / (1 + x);
a = 0;
b = 1;
APPINT = ((b - a) / 6) * (f[a] + (4 * f[(a + b) / 2]) + f[b]);
EXACTINT = N[Integrate[f[x], {x, a, b}]]
Error = Abs[APPINT - EXACTINT]
Print["APPINT= ", N[APPINT]]
Print["EXACTINT= ", EXACTINT]
Print["Errors ", Error]
Out[68]=
0.693147
Out[69]=
0.00129726

APPINT= 0.694444

EXACTINT= 0.693147

Errors 0.00129726
```

e) Using Simpson rule, find the approximate integration f(x)=1/1+x^2 on [0,1].

```
In[73]:= f[x_] = 1 / (1 + x^2);
         a = 0;
         b = 1;
         APPINT = ((b - a) / 6) * (f[a] + (4 * f[(a + b) / 2]) + f[b]);
         EXACTINT = N[Integrate[f[x], {x, a, b}]]
         Error = Abs[APPINT - EXACTINT]
         Print["APPINT= ", N[APPINT]]
         Print["EXACTINT= ", EXACTINT]
         Print["Errors ", Error]
Out[77]=
         0.785398
Out[78]=
         0.00206483
         APPINT= 0.783333
         EXACTINT= 0.785398
         Errors 0.00206483
```

f) Using Simpson rule, find the approximate integration f(x)=tan-1x on [0,1].

```
In[82]:= f[x_] = ArcTan[x];
         a = 0;
         b = 1;
         APPINT = ((b - a) / 6) * (f[a] + (4 * f[(a + b) / 2]) + f[b]);
         EXACTINT = N[Integrate[f[x], {x, a, b}]]
         Error = Abs[APPINT - EXACTINT]
         Print["APPINT= ", N[APPINT]]
         Print["EXACTINT= ", EXACTINT]
         Print["Errors ", Error]
Out[86]=
         0.438825
Out[87]=
         0.00117353
         APPINT= 0.439998
         EXACTINT= 0.438825
         Errors 0.00117353
```

8) Euler methods for solving first order initial value problems of ODE's.

a) Using euler method, solve the given function f(x)=y+x in the interval [0,1] and number of iteraton is 2.

```
In[100]:=
f[x_, y_] := x + y;
n = 2;
a = 0;
b = 1;
h = (b - a) / n;
y[0] = 2;
temp = y[0];
For[i = 0, i ≤ n - 1, i++,
  x[i] = a + i * h;
  y[i] = temp;
  y[i + 1] = y[i] + h * f[x[i], y[i]];
  Print["The ", i + 1, " approximation is ", N[y[i + 1]]];
  temp = y[i + 1]];

The 1 approximation is 3.

The 2 approximation is 4.75
```

b) Using euler method, solve the given function f(x)=2x+y in the interval [0,1] and number of iteraton is 5.

In[108]:=

```
f[x_, y_] := 2 * x + y;
n = 5;
a = 0;
b = 1;
h = (b - a) / n;
y[0] = 2;
temp = y[0];
For[i = 0, i ≤ n - 1, i++,
  x[i] = a + i * h;
  y[i] = temp;
  y[i + 1] = y[i] + h * f[x[i], y[i]];
  Print["The ", i + 1, " approximation is ", N[y[i + 1]]];
  temp = y[i + 1]];
```

```
The 1 approximation is 2.4

The 2 approximation is 2.96

The 3 approximation is 3.712

The 4 approximation is 4.6944

The 5 approximation is 5.95328
```

c) Using euler method, solve the given function f(x)=1+y/x in the interval [1,6] and number of iteraton is 10.

```
In[132]:=
f[x_, y_] := 1 + y / x;
n = 10;
a = 1;
b = 6;
h = (b - a) / n;
y[0] = 2;
temp = y[0];
For[i = 0, i ≤ n - 1, i++,
  x[i] = a + i * h;
  y[i] = temp;
  y[i + 1] = y[i] + h * f[x[i], y[i]];
  Print["The ", i + 1, " approximation is ", N[y[i + 1]]];
  temp = y[i + 1]];

The 1 approximation is 3.5
The 2 approximation is 5.16667
The 3 approximation is 6.95833
The 4 approximation is 8.85
The 5 approximation is 10.825
The 6 approximation is 12.8714
The 7 approximation is 14.9804
The 8 approximation is 17.1448
The 9 approximation is 19.3593
The 10 approximation is 21.6193
```