% Date - 30 / 01 / 2024
% Name - Divyansh Aggarwal %
% Roll No.-21 MTS5713 %

PRACTICAL : 1
BISECTION METHOD WITH CONDITION OF CONVERGANCE
AIM : - To perform the iteration of bisection method for the function f1 (x) = x ^ 3 + 2 x ^ 2 - 3 x - 1, f2 (x) = x ^ 3 + 2 x ^ 2 - 3 x - 3 and f3 (x) = Sin[x] on the interval [1, 2], [1, 2], and [3, 4] respectively within an absolute convergance (error tolerance) of 10 ^ (-7)

```
Bisection[a0_, b0_, m_] := Module[{a = N[a0], b = N[b0]}, c = (a + b) / 2;
k = 0;
While[k < m && ((b - a) / 2) > 10 ^ (-7), If[Sign[f[b]] == Sign[f[c]], b = c, a = c];
c = (a + b) / 2;
k = k + 1;];
Print["c=", NumberForm[c, 16]];
Print["f[c]=", NumberForm[f[c], 16]];];
```

In[4]:=
```
f[x_] = x ^ 3 + 2 * x ^ 2 - 3 * x - 1;
Bisection[1, 2, 30]
```

c=1.198691189289093

f[c]=-3.310740535056311 × 10^{-7}

In[6]:=
```
f[x_] = x ^ 3 + 2 * x ^ 2 - 3 * x - 3;
Bisection[1, 2, 30]
```

c=1.460504829883575

f[c]=-3.708990110595778 × 10^{-7}

In[8]:=
```
f[x_] = Sin[x];
Bisection[1, 2, 30]
```

c=1.000000059604645

f[c]=0.841471017012422

In[10]:=
```
f[x_] = x ^ 3 - 5 * x + 1
Bisection[1, 2, 30]
```

Out[10]=

$1 - 5x + x^3$

c=1.000000059604645

f[c]=-3.000000119209279

% Date - 06 / 02 / 2024
% Name - Divyansh Aggarwal %
% Roll No.-21 MTS5713 %

practical : 2

SECANT METHOD

AIM : - To perform the iteration of Secant Method for the functions f (x) = x ^ 3 + 2 * x - 5,
f (2 x) = Cos[x] - x and f (3 x) = Sin[x] on the intervals [1, 2], [0, 1],
and [0, 1] respectively within an absolute convergence of 5 * 10 ^ (-7).

In[9]:=
```
SecantMethod[x0_, x1_, max_] := Module[{}, k = 1; p0 = N[x0];
p1 = N[x1];
p2 = p1;
p1 = p0;
While[(k < max && Abs[f[p2]] > 5 * 10 ^ (-7)),
p0 = p1;
p1 = p2;
p2 = p1 - (f[p1] (p1 - p0) / (f[p1] - f[p0]));
k = k + 1;];
Print["p", k, "=", NumberForm[p2, 11]];
Print["f[p", k, "]=", NumberForm[f[p2], 11]];]
```

In[22]:=
```
f[x_] := x ^ 3 - 2 * x - 5;
SecantMethod[1, 2, 50]
```

p6=2.0945514814

f[p6]=$-2.0090498154 \times 10^{-9}$

In[12]:=
```
f[x_] := x ^ 3 - 5 * x + 1;
SecantMethod[0, 1, 50]
```

p6=0.20163967572

f[p6]=$1.0352718682 \times 10^{-11}$

In[14]:=
```
f[x_] := Cos[x] - x * Exp[x];
SecantMethod[0, 1, 50]
```

p7=0.51775737075

f[p7]=$-2.1513164583 \times 10^{-8}$

In[16]:=
```
f[x_] := Cos[x] - x;
SecantMethod[0, 1, 50]
```

p5=0.73908511213

f[p5]=$3.5292622824 \times 10^{-8}$

In[20]:=
```
f[x_] := Sin[x];
SecantMethod[0, 1, 50]
```

p2=0.

f[p2]=0.

% Date - 09 / 02 / 2024
% Name - Divyansh Aggarwal %
% Roll No.-21 MTS5713 %

PRACTICAL - 2. part B
AIM : - To perform the iteration of regula falsi Method for the \
functions f (x) = x ^ 3 + 2 * x ^ 2 - 3 * x - 1,
f (x) = x ^ 3 + 2 x - 1 and f (x) =
e ^ (-x) - x on the intervals [1, 2], [0, 1] and [0,
1] respectively within an absolute convergence of 10 ^ (12)

In[9]:=
```
RegulaFalsi[a0_, b0_, m_] := Module[{}, a = N[a0]; b = N[b0];
 If[f[a] * f[b] > 0, Print["interval is not correct"]; Break[],
  c = (a * f[b] - b * f[a]) / (f[b] - f[a]);
  k = 0;
  While[(k < m && Abs[f[c]] > 10 ^ (-12)),
   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["the result after ", k, "iterations=  ", NumberForm[c, 16]];
  Print["f[c]=", NumberForm[f[c], 16]];];]
```

In[10]:=
```
f[x_] = x ^ 3 + 2 * x ^ 2 - 3 * x - 1;
RegulaFalsi[1, 2, 50]
```

the result after 35iterations= 1.19869124351587

f[c]=$-7.780442956573097 \times 10^{-13}$

%Date-13/02/2024
%Name-Divyansh Aggarwal%
%Roll No.-21MTS5713%

Practical - 3
AIM : - To perform the iteration of Newton Raphson Method for the functions f (x) =
x ^ 3 + 2 * x ^ 2 - 3 * x - 1, f (2 x) = Cos[x] - x and f (3 x) = e ^ (- x) - x on the intervals [1, 2], [
0, 1], and [0, 1] respectively within an absolute convergence of 10 ^ (- 8)

In[1]:=
```
NewtonRaphson[x0_, max_] := Module[{}, k = 0; p0 = N[x0];
p1 = p0;
While[(k < max && Abs[f[p1]] > 10^(-8)),
p0 = p1;
If[f'[p0] == 0, Print["p0 is not correct"]; Exit[];,
p1 = p0 - f[p0] / f'[p0];
k = k + 1;];];
Print["p", k, "iterations =", NumberForm[p1, 16]];
Print["f[p]=", NumberForm[f[p1], 16]];]
```

In[2]:=
```
f[x_] = x^3 + 2*x^2 - 3*x - 1;
NewtonRaphson[2, 13];
```

p5iterations =1.19869124352843

f[p]=$7.59046159259924 \times 10^{-11}$

In[4]:=
```
f[x_] = Cos[x] - x;
NewtonRaphson[1, 30];
```

p3iterations =0.739085133385284

f[p]=$-2.847205804457076 \times 10^{-10}$

In[7]:=
```
f[x_] = Exp[-x] - x;
NewtonRaphson[1, 20];
```

p3iterations =0.567143285989123

f[p]=$6.927808993140161 \times 10^{-9}$

%Date-13/02/2024%
%Name-Divyansh Aggarwal%
%Roll No.-21MTS5713%

Practical - 4

In[28]:=
```
Gausselim[A0_] := Module[{a = N[A0]}, Print[MatrixForm[a]];
size = Dimensions[a];
n = size[[1]];
m = size[[2]];
For[i = 1, i ≤ n - 1, i = i + 1,
For[k = i + 1, k ≤ n, k = k + 1,
(factor = a[[k, i]] / a[[i, i]];);
For[p = i, p ≤ m, p = p + 1,
a[[k, p]] = a[[k, p]] - factor * a[[i, p]];];];];
Print[MatrixForm[a]];
ClearAll[x, i];
x[n] = a[[n, m]] / a[[n, n]];
Print[x[n]];
For[i = n - 1, i ≥ 1, i = i - 1,
s = 0;
For[j = i + 1, j ≤ n, j = j + 1,
s = s + a[[i, j]] * x[j];];
x[i] = (a[[i, m]] - s) / (a[[i, i]]);
Print[x[i]];];];
```

In[31]:=
```
a = {{2, 1, 1, 10}, {3, 2, 3, 18}, {1, 4, 9, 16}};
Gausselim[a]
```

$$\begin{pmatrix} 2. & 1. & 1. & 10. \\ 3. & 2. & 3. & 18. \\ 1. & 4. & 9. & 16. \end{pmatrix}$$

$$\begin{pmatrix} 2. & 1. & 1. & 10. \\ 0. & 0.5 & 1.5 & 3. \\ 0. & 0. & -2. & -10. \end{pmatrix}$$

5.

-9.

7.

% Date - 20 / 02 / 2024 × %

% Name - Divyansh Aggarwal %

% Roll No.-21 MTS5713 %

Practical - 5

In[3]:=
```
(*Programming*)
Gaussjacobi[A0_, B0_, X0_, max_] :=
 Module[{A = N[A0], B = N[B0], i, j, k = 0, n = Length[X0], X = X0, Xold = X0},
Print["X", 0, "=" X];
While[k < max,
For[i = 1, i ≤ n, i = i + 1,
X[[i]] = (B[[i]] - Sum[A[[i, j]] * Xold[[j]], {j, 1, i - 1}] - Sum[A[[i, j]] * Xold[[j]], {j, i + 1, n}]) / A[[i, i]];
Print["X", k + 1, "=", NumberForm[X, 10]];
If[Max[Abs[X - Xold]] < 5 * 10 ^ (-6),
Print["Solution with convergence tolerance of 5*10^(-6)=",
NumberForm[X, 10]];
Break[]; '
Xold = X;
k = k + 1;];];]
```

```
% Date - 20 / 02 / 2024
% Name - Divyansh Aggarwal %
% Roll No.-21 MTS5713 %
Practical - 6
```

```
(*Aim-To solve the following system of linear equations by using Gauss-Seidal Method
    within an absolute tolerance of 5*10^(-6): 4x1-x2=2 -x1+4x2-x3=4 -x2+4x3=10 *)
```

In[1]:=
```
GaussSeidal[A0_, B0_, X0_, max_] :=
 Module[{A = N[A0], B = N[B0], i, j, k = 0, n = Length[X0], X = X0, Xold = X0},
Print["X", 0, "=" X];
While[k < max,
For[i = 1, i ≤ n, i = i + 1,
X[[i]] = (B[[i]] - Sum[A[[i, j]] * Xold[[j]], {j, 1, i - 1}] - Sum[A[[i, j]] * Xold[[j]], {j, i + 1, n}]) / A[[i, i]];
Print["X", k + 1, "=", NumberForm[X, 10]];
If[Max[Abs[X - Xold]] < 5 * 10 ^ (-6),
Print["Solution with convergence tolerance of 5*10^(-6)=",
NumberForm[X, 10]];
Break[]; '
Xold = X;
k = k + 1;];];]
```

%Date-27/02/2024%

%Name-Divyansh Aggarwal%

%Roll No.-21MTS5713%

```
(*Aim To estimate the values of e^0.5,
e^-0.7 and e^0.3by constructing the lagrange's form of interpolationg
 polynomial for f passing through (- 1, e^-1) , (0, 1) and (1, e^1) *)
```

In[19]:=
```
Lagrange1[x_, f_, y_] := Module[{}, s = 0; m = Length[x]; p = 1;
For[i = 1, i ≤ m, i = i + 1,
For[j = 1, j ≤ m, j = j + 1,
If[j ≠ i,
p = p * (y - x[[j]]) / (x[[i]] - x[[j]]); Continue;];];
Print["The Polynomial=" , p];
s = s + p * f[[i]]; p = 1;];
Print["Function value at y=", s];
Print["Absoulte error=", Abs[s - Exp[y]]];]
```

In[20]:=
```
x = {-1, 0, 1};
f = {Exp[-1], 1, Exp[1]};
Lagrange1[x, f, 0.5]
```

```
The Polynomial=-0.125
The Polynomial=0.75
The Polynomial=0.375
Function value at y=1.72337
Absoulte error=0.0746495
```

In[23]:=
```
Lagrange1[x, f, -0.7]
```

```
The Polynomial=0.595
The Polynomial=0.51
The Polynomial=-0.105
Function value at y=0.443469
Absoulte error=0.0531166
```

In[24]:=
```
Lagrange1[x, f, 0.3]
```

```
The Polynomial=-0.105
The Polynomial=0.91
The Polynomial=0.195
Function value at y=1.40144
Absoulte error=0.0515788
```

%Date-05/03/2024
%Name-Divyansh Aggarwal%
%Roll No.-21MTS5713%

% Newton interpolation Method

```
In[1]:= NthDividedDiff[x0_, f0_, start_, end_] :=
  Module[{x = x0, f = f0, i = start, j = end, ans}, If[i == j, Return[f[[i]]],
    ans = (NthDividedDiff[x, f, i + 1, j] - NthDividedDiff[x, f, i, j - 1]) / (x[[j]] - x[[i]]);
Return[ans]];];

NewtonDDPoly[x0_, f0_] := Module[{x1 = x0, f = f0, n, P, k, j},
n = Length[x1];
P[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]])];
P[y_] = P[y] + NthDividedDiff[x1, f, 1, i] * prod[y]];
Return[P[y]];];
nodes = {0, 1, 3};
values = {1, 3, 55};
NewtonPoly[y_] = NewtonDDPoly[nodes, values];
NewtonPoly[y]
NewtonPoly[y_] = Simplify[NewtonPoly[y]];
NewtonPoly[y]
NewtonPoly[2]

Out[6]= 1 + 2 y + 8 (-1 + y) y

Out[8]= 1 - 6 y + 8 y^2

Out[9]= 21

% Date - 19 / 03 / 2024
% Name - Divyansh Aggarwal %
% Roll No.-21 MTS5713 %

trapezoidaRule[a0_, b0_, n_, f_] := Module[{a = a0, b = b0, h, ai}, h = (b - a) / n;
ai = h / 2 * (f[a] + f[b] + 2 * Sumf[a + h * k], {k = 1, n - 1});
Return[ai];]

Syntax: "(" cannot be followed by "f[a] + f[b] + 2 * Sum {f[a + h * k]}, {k = 1, n - 1})".
```