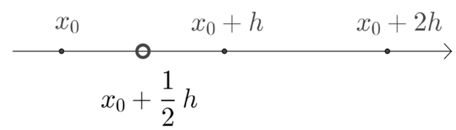
Exam 2: Tea time

Problems #1 and 2

Preliminaries:



In[905]:=
```
f[x_] := x * (x - 1) * E^x
f[-0.5]
Together[f'[x]]
Together[f''[x]]
Together[f'''[x]]
Together[f''''[x]]
Solve[f'[x] == 0, x]
N[Solve[f'''''[x] == 0, x]]
pf = Plot[f[x], {x, -3, 2}, PlotRange → {-0.5, 3}]

(* Plot[{f[x],f[-1]+f'[-1](x-(-1)),
  f[1/2 (-1+√5)]+(f[1/2 (-1+√5)]-f[-2])/(1/2 (-1+√5)-(-2))(x-1/2 (-1+√5))},
 {x,-3,2}]*)
```

Out[906]= 0.454897994784475

Out[907]= $e^x\left(-1+x+x^2\right)$

Out[908]= $e^x\left(3x+x^2\right)$

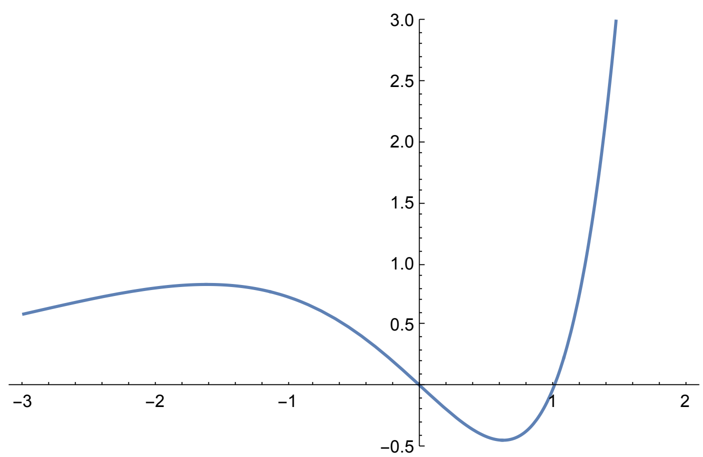
Out[909]= $e^x\left(3+5x+x^2\right)$

Out[910]= $e^x\left(8+7x+x^2\right)$

Out[911]= $\left\{\left\{x\to\frac{1}{2}\left(-1-\sqrt{5}\right)\right\},\left\{x\to\frac{1}{2}\left(-1+\sqrt{5}\right)\right\}\right\}$

Out[912]= $\{\{x\to-6.79128784747792\},\{x\to-2.20871215252208\}\}$

Out[913]=



Preliminaries:
Lagrange Polynomials make fitting points easy!

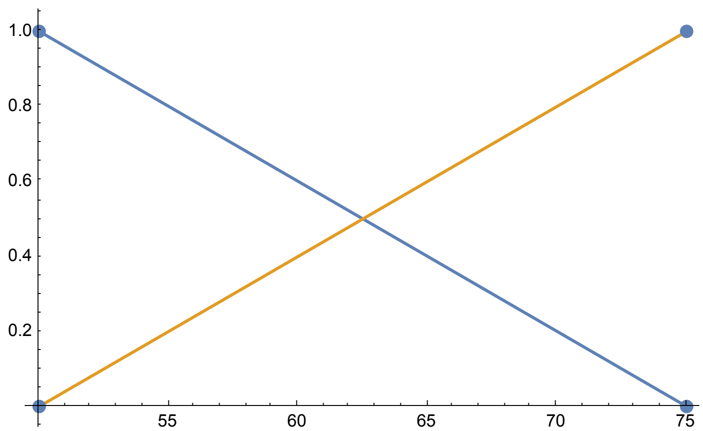
Lagrange's linear functions are linear functions which are zero at one value of x, and one at the other:

In[914]:=
```
lagrange[x_, x1_, x2_] := (x - x2) / (x1 - x2) (* 1 at x1, 0 at x2 *)
{x0, x1} = {50, 75}
{y0, y1} = {180, 230}
Show[
 Plot[{lagrange[x, x0, x1], lagrange[x, x1, x0]}, {x, x0, x1}],
 ListPlot[
  {{x0, 0}, {x1, 0}, {x0, 1}, {x1, 1}},
  PlotStyle → {PointSize → Large}]
]
```

Out[915]= {50, 75}

Out[916]= {180, 230}

Out[917]=



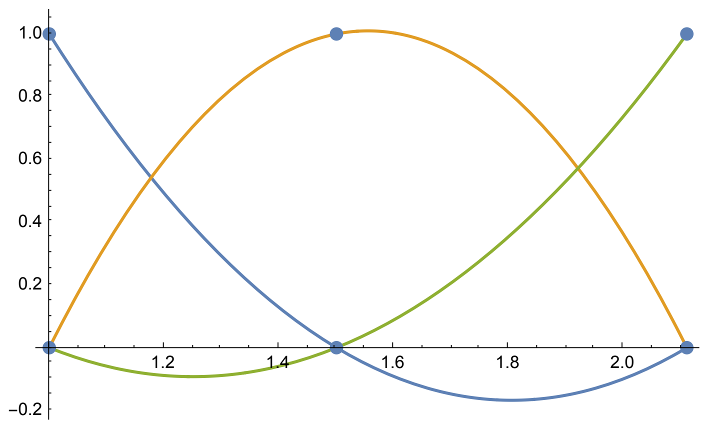
Lagrange linear polynomials are just sums of two Lagrange linear functions that fit two points

Lagrange quadratic functions are just products of Lagrange linear functions -- and they fit three points: two with function values zeros, and a function value of 1:

In[918]:=
```
lagrange2[x_, x1_, x2_, x3_] := lagrange[x, x1, x2] * lagrange[x, x1, x3]
(* lagrange2 is 1 at x1, zero at x2 and x3 *)
(* (1,2),(1.5,-0.83),and (2.11,-1) For problem #1: *)
{x0, x1, x2} = {1, 1.5, 2.11}
{y0, y1, y2} = {2, -0.83, -1}
Show[
 Plot[{lagrange2[x, x0, x1, x2],
   lagrange2[x, x1, x0, x2], lagrange2[x, x2, x0, x1]}, {x, x0, x2}],
 ListPlot[
  {{x0, 0}, {x1, 0}, {x2, 0}, {x0, 1}, {x1, 1}, {x2, 1}},
  PlotStyle → {PointSize → Large}]
]
```

Out[919]= {1, 1.5, 2.11}

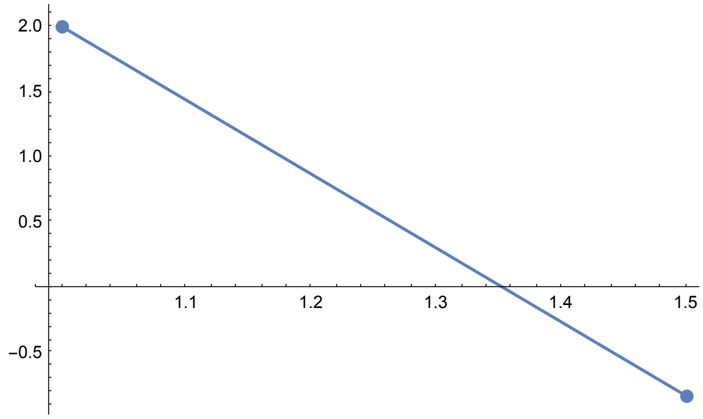
Out[920]= {2, -0.83, -1}

Out[921]=


Let’s use these Lagrange linear functions fit two points: (50,180), (75, 230):

In[922]:=
```
L1[x_, x0_, x1_, y0_, y1_] := y0 lagrange[x, x0, x1] + y1 lagrange[x, x1, x0]
p1 = ListPlot[{{x0, y0}, {x1, y1}}, PlotStyle → {PointSize → Large}];
p2 = Plot[L1[x, x0, x1, y0, y1], {x, x0, x1}];
Show[p1, p2]
```

Out[925]=



In[926]:=
```
lagrange3[x_, x1_, x2_, x3_, x4_] :=
 lagrange[x, x1, x2] * lagrange[x, x1, x3] * lagrange[x, x1, x4]
(* (0,0),(1,2),(4,-3),and (10,-1) *)
```

Exercise #1: Lagrange Polynomials

1. Let’s run a cubic through some points, and see how well it fits:

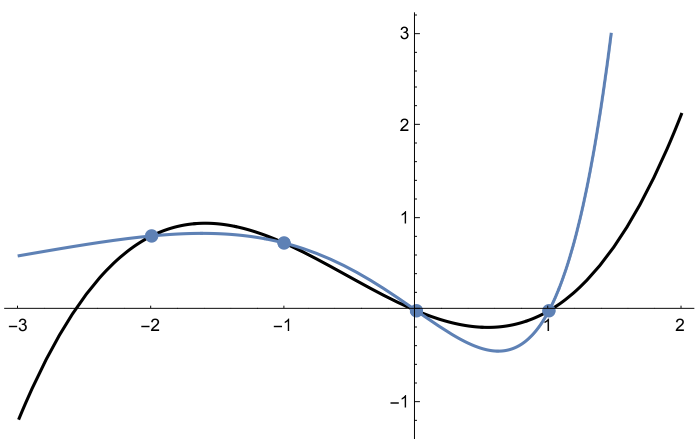
In[927]:=
```
{x0, x1, x2, x3} = {-2, -1, 0, 1}
{y0, y1, y2, y3} = f[{x0, x1, x2, x3}]
L3[x_, x0_, x1_, x2_, x3_, y0_, y1_, y2_, y3_] :=
 y0 lagrange3[x, x0, x1, x2, x3] + y1 lagrange3[x, x1, x2, x0, x3] +
  y2 lagrange3[x, x2, x0, x1, x3] + y3 lagrange3[x, x3, x2, x0, x1]
L3[x, x0, x1, x2, x3, y0, y1, y2, y3]
p1 =
  ListPlot[{{x0, y0}, {x1, y1}, {x2, y2}, {x3, y3}}, PlotStyle → {PointSize → Large}];
p2 = Plot[L3[x, x0, x1, x2, x3, y0, y1, y2, y3], {x, -3, 2}, PlotStyle → Black];
Show[p2, pf, p1, PlotRange → All]
Show[pf, p1, PlotRange → {-1, 2}]
```

Out[927]= {-2, -1, 0, 1}

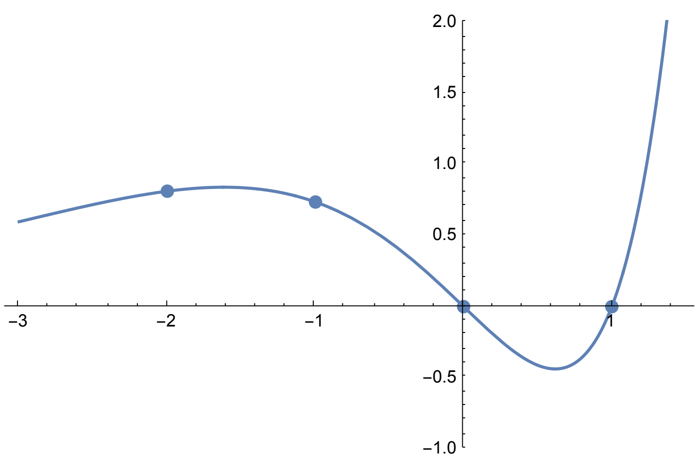
Out[928]= $\left\{\frac{6}{e^2}, \frac{2}{e}, 0, 0\right\}$

Out[930]= $-\frac{(-1-x)\,(1-x)\,x}{e^2} - \frac{(1-x)\,x\,(2+x)}{e}$

Out[933]=



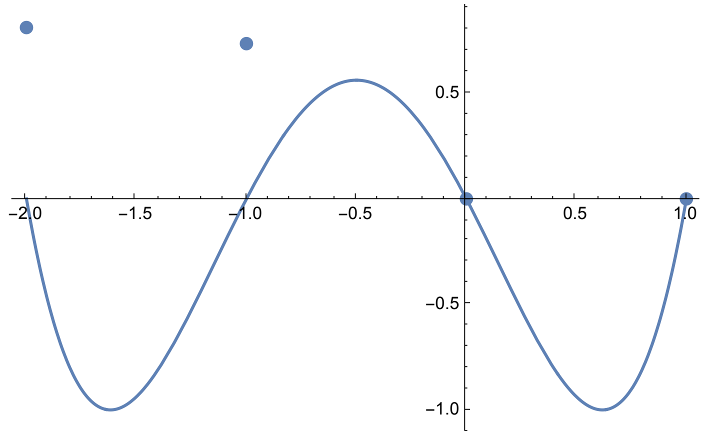
Out[934]=



In[935]:=

In[936]:=
```
p1 =
  ListPlot[{{x0, y0}, {x1, y1}, {x2, y2}, {x3, y3}}, PlotStyle → {PointSize → Large}];
p2 = Plot[(x - x0) (x - x1) (x - x2) (x - x3), {x, -2, 1}];
Show[p2, p1, PlotRange → All]
```

Out[938]=



Exercise #2: Newton Polynomials

In[939]:=
```
dividedDifference[data_, verbose_] :=
 Module[{n = Length[data], xs, ys, i, j, p},
  p = ConstantArray[0, {n, n}];
  {xs, ys} = Transpose[data];  (* The data is a list of coordinates *)
  p⟦All, 1⟧ = ys; (* We initialize the first column of p with the y-values *)
  For[j = 2, j ≤ n + 1, j++,
   For[i = 1, i ≤ n + 1 - j, i++,
     p⟦i, j⟧ = (* Successive columns are filled with divided differences,
       from the previous column. *)
       (p⟦i + 1, j - 1⟧ - p⟦i, j - 1⟧) / (xs⟦i + j - 1⟧ - xs⟦i⟧);
    ];
  ];
  If[verbose, Print[MatrixForm[p]]];
  p⟦1, All⟧
 ]
```

In[940]:=
```
interpolator[x_, coefs_, xs_, verbose_] :=
 Module[
  {result, n = Length[xs]},
  result = coefs⟦n⟧;
  For[i = n - 1, i > 0, i--,
   If[verbose, Print[{xs⟦i⟧, coefs⟦i⟧}]];
   result = coefs⟦i⟧ + (x - xs⟦i⟧) result;
  ];
  result
 ]
```

In[941]:=
```
xpoints = {0, -1, 1, -2};
ypoints = f[xpoints] * 1.0;
newtonCoefs = Simplify[dividedDifference[Transpose[{xpoints, ypoints}], True]]
ypoints = f[xpoints];
newtonCoefs = Simplify[dividedDifference[Transpose[{xpoints, ypoints}], True]]
```

$$\begin{pmatrix} 0. & -0.735758882342885 & 0.367879441171442 & 0.23254415793483 \\ 0.735758882342885 & -0.367879441171442 & -0.0972088746982169 & 0 \\ 0. & -0.270670566473225 & 0 & 0 \\ 0.812011699419676 & 0 & 0 & 0 \end{pmatrix}$$

Out[943]= {0., -0.735758882342885, 0.367879441171442, 0.23254415793483}

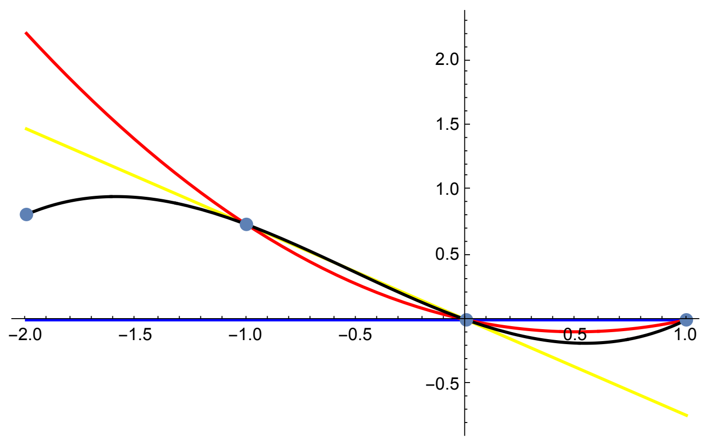
$$\begin{pmatrix} 0 & -\frac{2}{e} & \frac{1}{e} & \frac{1}{2}\left(-\frac{2}{e^2}+\frac{2}{e}\right) \\ \frac{2}{e} & -\frac{1}{e} & \frac{2}{e^2}-\frac{1}{e} & 0 \\ 0 & -\frac{2}{e^2} & 0 & 0 \\ \frac{6}{e^2} & 0 & 0 & 0 \end{pmatrix}$$

Out[945]= $\left\{0, -\frac{2}{e}, \frac{1}{e}, \frac{-1+e}{e^2}\right\}$

In[946]:=
```
{newtonCoefs⟦1⟧, newtonCoefs⟦2⟧}
Show[
 Plot[interpolator[x, newtonCoefs⟦1 ;; 1⟧, xpoints⟦1 ;; 1⟧, False],
  {x, Min[xpoints], Max[xpoints]}, PlotStyle → Blue],
 Plot[interpolator[x, newtonCoefs⟦1 ;; 2⟧, xpoints⟦1 ;; 2⟧, False],
  {x, Min[xpoints], Max[xpoints]}, PlotStyle → Yellow],
 Plot[interpolator[x, newtonCoefs⟦1 ;; 3⟧, xpoints⟦1 ;; 3⟧, False],
  {x, Min[xpoints], Max[xpoints]}, PlotStyle → Red],
 Plot[interpolator[x, newtonCoefs⟦1 ;; 4⟧, xpoints⟦1 ;; 4⟧, False],
  {x, Min[xpoints], Max[xpoints]}, PlotStyle → Black],
 ListPlot[Transpose[{xpoints, ypoints}], PlotStyle → {PointSize → Large}],
 PlotRange → All
]
```

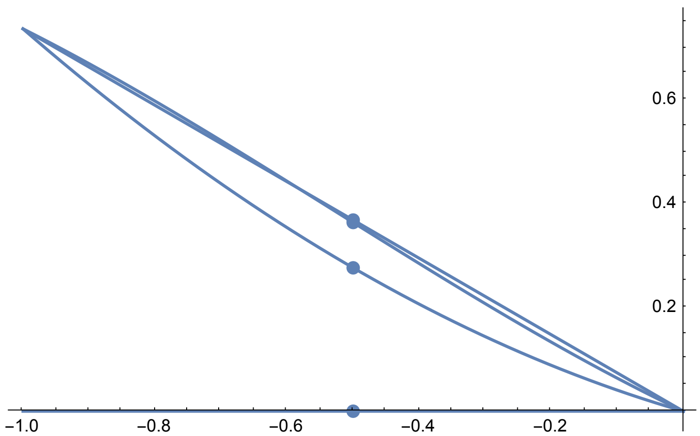
Out[946]= $\left\{0, -\frac{2}{e}\right\}$

Out[947]=



In[948]:=
```
Show[Plot[interpolator[x, newtonCoefs⟦1 ;; 1⟧, xpoints⟦1 ;; 1⟧, False], {x, -1, 0}],
 Plot[interpolator[x, newtonCoefs⟦1 ;; 2⟧, xpoints⟦1 ;; 2⟧, False], {x, -1, 0}],
 Plot[interpolator[x, newtonCoefs⟦1 ;; 3⟧, xpoints⟦1 ;; 3⟧, False], {x, -1, 0}],
 Plot[interpolator[x, newtonCoefs⟦1 ;; 4⟧, xpoints⟦1 ;; 4⟧, False], {x, -1, 0}],
 ListPlot[{{-0.5, interpolator[-0.5, newtonCoefs⟦1 ;; 1⟧, xpoints⟦1 ;; 1⟧, False]},
   {-0.5, interpolator[-0.5, newtonCoefs⟦1 ;; 2⟧, xpoints⟦1 ;; 2⟧, False]},
   {-0.5, interpolator[-0.5, newtonCoefs⟦1 ;; 3⟧, xpoints⟦1 ;; 3⟧, False]},
   {-0.5, interpolator[-0.5, newtonCoefs⟦1 ;; 4⟧, xpoints⟦1 ;; 4⟧, False]}},
  PlotStyle → {PointSize → Large}], PlotRange → All
]
```

Out[948]=



In[949]:=
```
e0 = interpolator[-0.5, newtonCoefs⟦1 ;; 1⟧, xpoints⟦1 ;; 1⟧, False];
e1 = interpolator[-0.5, newtonCoefs⟦1 ;; 2⟧, xpoints⟦1 ;; 2⟧, False];
e2 = interpolator[-0.5, newtonCoefs⟦1 ;; 3⟧, xpoints⟦1 ;; 3⟧, False];
e3 = interpolator[-0.5, newtonCoefs⟦1 ;; 4⟧, xpoints⟦1 ;; 4⟧, False];
true = f[-0.5]
MatrixForm[{{-0.5, e0, Abs[e0 - true]}, {-0.5, e1, Abs[e1 - true]},
  {-0.5, e2, Abs[e2 - true]}, {-0.5, e3, Abs[e3 - true]}}]
```

Out[953]= 0.454897994784475

Out[954]//MatrixForm=

-0.5	0	0.454897994784475
-0.5	0.367879441171442	0.0870185536130327
-0.5	0.275909580878582	0.178988413905893
-0.5	0.363113640104143	0.0917843546803322