### **Example 2.2.3a Diffusion with a Second Order Reaction**

> restart:

> with(plots):

The number of node points is entered here:

> N:=4;



The length of the domain is entered here:

> L:=1;



The governing equation is entered below:

> eq:=diff(y(x),x$2)-Phi^2\*y(x)^2;



The boundary conditions are entered here:

> bc1:=diff(y(x),x);



> bc2:=y(x)-1;



Next, a general program is written to convert the governing equation and the boundary conditions to finite difference form. The central difference expression for the second and first derivatives are:

> d2ydx2:=(y[m+1]-2\*y[m]+y[m-1])/h^2;



> dydx:=(y[m+1]-y[m-1])/2/h;



Three point forward and backward difference expressions for the derivative are:

> dydxf:=(-y[2]+4\*y[1]-3\*y[0])/(2\*h);



> dydxb:=(y[N-1]-4\*y[N]+3\*y[N+1])/(2\*h);



The governing equation in finite difference form is:

> Eq[m]:=subs(diff(y(x),x$2)=d2ydx2,diff(y(x),x)=dydx,y(x)=y[m],x=m\*h,eq);



The boundary conditions in finite difference form are:

> Eq[0]:=subs(diff(y(x),x)=dydxf,y(x)=y[0],bc1);



> Eq[N+1]:=subs(diff(y(x),x)=dydxb,y(x)=y[N+1],bc2);



A 'for loop' can be written for the interior node points as

> for i to N do Eq[i]:=subs(m=i,Eq[m]);od;



The node spacing is given by:

> h:=L/(N+1);



The value for Φ is sustained in the governing equations. The governing equations are stored in eqs:

> eqs:=seq(eval(subs(Phi=1,Eq[i])),i=0..N+1);



The variables are stored in vars:

> vars:=seq(y[i],i=0..N+1);



The 'fsolve' command sometimes gives negative values when guess values for the dependent variables are not provided. To avoid this, an initial guess of 1 is provided:

> fsolve({eqs},{vars});



> vars:=seq(y[i]=1,i=0..N+1);



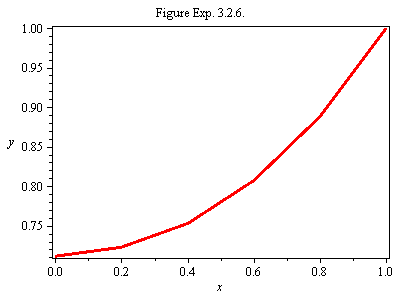
> sol:=fsolve({eqs},{vars});



The solution obtained is assigned and plotted:

> assign(sol):

> plot([seq([i\*h,y[i]],i=0..N+1)],thickness=4,axes=boxed,title="Figure Exp. 3.2.6.",labels=[x,y]);



The accuracy of the solution obtained can be checked by following the concentration at the center y[0] and increasing the number of node points:

> y[0];



>