HW3\_p2 redo  
> **makeJac:=proc(f)**

**local N,jac;**

**N:=LinearAlgebra:-Dimension(f);**

**jac:=VectorCalculus:-Jacobian(f,[seq(y[i],i=1..N)]);**

**unapply(jac,y);**

**end proc:**

> **Newton:=proc(f,y0,makeJac,tol,maxiter)**

**#global**

**local N,dy,Jac,F,yold,ynew,Err,iter;**

**N:=LinearAlgebra:-Dimension(f);**

**F:=unapply(f,y);**

**Jac:=makeJac(f);**

**yold:=y0;**

**dy:=LinearAlgebra:-LinearSolve(-Jac(yold),F(yold));**

**ynew:=yold+dy;**

**yold:=ynew;**

**Err:=10;**

**iter:=1;**

**while Err>tol and iter < maxiter do**

**iter:=iter+1;**

**dy:=LinearAlgebra:-LinearSolve(-Jac(yold),F(yold));**

**ynew:=yold+dy;**

**Err:=LinearAlgebra:-Norm(yold-ynew);**

**yold:=ynew;**

**end;**

> **#[Err,ynew,iter];**

**ynew;**

**end proc:**

> **y00:=Vector(2,[1.0,0.0]);**

**eq1:=diff(y[1](t),t)=-1000\*y[1]^2:**

**eq2:=diff(y[2](t),t)=1000\*y[1]^2-y[2];**

**f:=Vector(2,[rhs(eq1),rhs(eq2)]);**

**F:=unapply(h\*f+y0-Vector(LinearAlgebra:-Dimension(f),[seq(y[i],i=1..LinearAlgebra:-Dimension(f))]),y0,h);**

**F(y00,0.1);**

**makeJac(f);**

**yy[0]=y00;**

**yy[1]:=Newton(F(y00,0.1),y00,makeJac,1e-6,50);**

**Newton(F(y00,0.1),y00,makeJac,1e-6,50);**



















> **EulerBDF:=proc(f,y00,tf,N)**

**local F,YY,h,y0,i;**

**F:=unapply(h\*f+y0-Vector(LinearAlgebra:-Dimension(f),[seq(y[i],i=1..LinearAlgebra:-Dimension(f))]),y0,h);**

**h:=tf/N;**

**YY[0]:=y00;**

**YY[1]:=Newton(F(y00,h),y00,makeJac,1e-6,50);#print(YY[1]);**

**for i from 2 to N do**

**YY[i]:=Newton(F(YY[i-1],h),YY[i-1],makeJac,1e-6,50);**

**od;**

**[seq([i\*h,YY[i]],i=0..N)];#for printing all the values**

**YY[N];**

**end proc:**

>

> **N:=10;**

**for i from 1 to N do**

**YYY[i]:=EulerBDF(f,y00,1,2^i):od;**

**for i from 1 to N-1 do**

**Z[i]:={seq(abs(YYY[i+1][j]-YYY[i][j]),j=1..LinearAlgebra:-Dimension(f))};od;**

**for i from 1 to N-2 do**

**ZZ[i]:={seq(Z[i][j]/Z[i+1][j],j=1..LinearAlgebra:-Dimension(f))};od;**

























































>

> > **F:=unapply(h\*f+y0-Vector(LinearAlgebra:-Dimension(f),[seq(y[i],i=1..LinearAlgebra:-Dimension(f))]),y0,h);**

**h0:=1e-2;**

**YY[0]:=y00;**

**F(y00,h0);**

**singlestepback(F,y00,h0,1);**

**singlestep(FF,y00,h0);**

> **singlestepback:=proc(F,y0,h,ord)**

**local Yh, Yh2,Ymid,Ord;**

**Ord:=ord;**

**Yh:=Newton(F(y0,h),y0,makeJac,1e-6,50);**

**print(Yh);**

**Ymid:=Newton(F(y0,h/2),y0,makeJac,1e-6,50);**

**print(Ymid);**

**Yh2:=Ymid+h/2\*Newton(F(Ymid,h),Ymid,makeJac,1e-6,50);**

**print(Yh2);**

**LinearAlgebra:-Norm(Yh-Yh2),(2^Ord\*Yh2-Yh)/(2^Ord-1);**

**end proc;**

**singlestep:=proc(F,y0,h)**

**local Yh, Yh2,Ymid,Ord;**

**Ord:=1;**

**Yh:=y0+h\*F(y0,h);**

**Ymid:=y0+h/2\*F(y0,h/2);**

**Yh2:=Ymid+h/2\*F(Ymid,h);**

**LinearAlgebra:-Norm(Yh-Yh2),(2^Ord\*Yh2-Yh)/(2^Ord-1);**

**end proc;**

**EulerBDF:=proc(f,y00,tf,N)**

**local F,YY,h,y0,i;**

**F:=unapply(h\*f+y0-Vector(LinearAlgebra:-Dimension(f),[seq(y[i],i=1..LinearAlgebra:-Dimension(f))]),y0,h);**

**h:=tf/N;**

**YY[0]:=y00;**

**YY[1]:=Newton(F(y00,h),y00,makeJac,1e-6,50);#print(YY[1]);**

**for i from 2 to N do**

**YY[i]:=Newton(F(YY[i-1],h),YY[i-1],makeJac,1e-6,50);**

**od;**

**[seq([i\*h,YY[i]],i=0..N)];#for printing all the values**

**YY[N];**

**end proc;**

**EulerBDFstep:=proc(f,y00,tf,N)**

**local F,YY,h,y0,i,h0;**

**F:=unapply(h\*f+y0-Vector(LinearAlgebra:-Dimension(f),[seq(y[i],i=1..LinearAlgebra:-Dimension(f))]),y0,h);**

**h0:=tf/N;**

**YY[0]:=y00;**

**print(F(y00,h0));**

**#YY[1]:=singlestepback(F(y00,h0),y00,h0,1);**

**YY[1]:=singlestep(F(y00,h0),y00,h0);**

**for i from 2 to N do**

**YY[i]:=Newton(F(YY[i-1],h0),YY[i-1],makeJac,1e-6,50);**

**od;**

**[seq([i\*h,YY[i]],i=0..N)];#for printing all the values**

**YY[N];**

**end proc;**