**main.f90**

program reacao

 parameter (nelmax=100)

 dimension tp(nelmax),t0(nelmax)

 external fcn

 common /const/ ht2

 common /param/ rk,rm

 open(1,file='input.txt')

 open(2,file='outnum.txt')

 open(3,file='out-read.txt')

 open(10,file='out-a.txt')

 open(11,file='out-b.txt')

 open(12,file='out-c.txt')

!

! integrador de EDO's com passo fixo - Euler

!

 read(1,\*)n

 write(3,\*)'n=',n

 read(1,\*)rk

 write(3,\*)'rk=',rk

 read(1,\*) rm

 write(3,\*)'rm=',rm

 read(1,\*) a0

 write(3,\*)'a0=',a0

 read(1,\*) b0

 write(3,\*)'b0=',b0

 read(1,\*) c0

 write(3,\*)'c0=',c0

 read(1,\*) tau0

 write(3,\*)'tau0=',tau0

 read(1,\*) dtau

 write(3,\*)'dtau=',dtau

 read(1,\*) ht2

 write(3,\*)'ht2=',ht2

 read(1,\*) tend

 write(3,\*)'tend=',tend

 read(1,\*) iperiod

 write(3,\*)'iperiod=',iperiod

 read(1,\*) tol1

 write(3,\*)'tol1=',tol1

!

! initial values

!

 time=tau0

 tp(1)=a0

 tp(2)=b0

 tp(3)=c0

!\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

!

!

! loop para apresentar resultados intermediarios

! em cada intervalo dtau durante a integracao das equacoes

!

 k=0

 write(\*,\*) ' Table of results'

 write(\*,\*)'-------------------------------------------------'

 write(\*,\*)' Passo Nr time a(i) b(i) c(i) '

 write(\*,\*)'-------------------------------------------------'

 write(\*,\*)k,time,(tp(l),l=1,n)

 write(2,\*)time,(tp(l),l=1,n)

 write(10,\*)time,tp(1)

 write(11,\*)time,tp(2)

 write(12,\*)time,tp(3)

! beginning of time loop

!

 50 k=k+1

 tendi=time+dtau

 write(\*,\*)'-------------time=',tendi

!

! armazena em t0(i) as variaveis do instante anterior

!

 do 110 i=1,n

 t0(i)=tp(i)

 110 continue

! resolve por Forward Euler

 call fore(n,fcn,time,tp,tendi,nelmax)

 write(\*,\*)k,tendi,(tp(l),l=1,n)

 write(2,\*)tendi,(tp(l),l=1,n)

 write(10,\*)tendi,tp(1)

 write(11,\*)tendi,tp(2)

 write(12,\*)tendi,tp(3)

!\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

!

 if(iperiod.eq.0) then

!

! simula até entrar em regime

 t0norm=rnorm2(n,t0,nelmax)

 do 120 l=1,n

 t0(l)=tp(l)-t0(l)

 120 continue

 dtnorm=rnorm2(n,t0,nelmax)

 if(dtnorm.lt.tol1\*dtau.or.tendi.ge.tend) then

 time=tendi

 goto 300

 endif

 else

!

! simula até tempo especificado

 if(tendi.ge.tend) then

 time=tendi

 goto 300

 endif

 endif

 time=tendi

 goto 50

!

!\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

!

 write(\*,\*) 'no steady state'

 write(\*,\*) 'dtnorm=',dtnorm

 write(\*,\*)'-------------time=',time

 do l=1,n

 write(\*,\*) l,'tp=',tp(l)

 enddo

 close(1)

 close(2)

 close(3)

 close(10)

 close(11)

 close(12)

call system ('notepad outnum.txt') ! listagem dos dados

call system ('wgnuplot dados.gnu') ! gráfico de concentracoes

 stop

 300 continue

 do l=1,n

 write(\*,\*) l,'tp=',tp(l)

 enddo

 close (1)

 close (2)

 close (3)

 close(10)

 close(11)

 close(12)

call system ('notepad outnum.txt') ! listagem dos dados

call system ('wgnuplot dados.gnu') ! gráfico de concentracoes

stop

end

!----------------------------------------------------------------

 function rnorm2(n,x,nd)

! compute euclidean norm of a vector

 dimension x(nd)

 sum=0.d0

 do i=1,n

 sum=sum+x(i)\*x(i)

 enddo

 aux=sqrt(sum)

 rnorm2=aux

 return

 end

!\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

 subroutine fcn(n,t,fi,f,nelmax)

 dimension fi(nelmax),f(nelmax)

 common /param/ rk,rm

!

 f(1) = -rk\*fi(1)\*\*rm

 f(2) = rk\*fi(1)\*\*rm

 f(3) = rk\*fi(1)\*\*rm

 return

 end

!\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

 subroutine fore(n,fcn,time,fi,tend,nelmax)

!

! implicit real \*8 (a-h,o-z)

 parameter (nd1=100)

 dimension fi(nelmax),f(nd1)

 common /const/ ht2

 external fcn

 k=0

 50 k=k+1

 time=min(time+ht2,tend)

 call fcn(n,time,fi,f,nelmax)

 do 100 i=1,n

 fi(i)=fi(i)+ht2\*f(i)

 100 continue

 if (time.lt.tend) goto 50

 return

 end

!------------------------------------------------------------------

**dados.gnu**

set data style linespoints

set grid

set xlabel 'Tempo t'

set ylabel 'Solucao numerica a(t),b(t),c(t)'

set title 'Concentracoes'

plot 'out-a.txt','out-b.txt','out-c.txt'

pause -1

**input.txt**

3 ! n = numero de incognitas

8.5 ! rk = const cinetica [L/(m.s)]

2. ! rm = ordem da reacao

0.4 ! a0 = valor inicial de a [M]

0. ! b0 = valor inicial de b [M]

0. ! c0 = valor inicial de c [M]

0. ! tau0 = tempo inicial [s]

0.1 ! dtau = incremento de tempo para visualizacao [s]

0.01 ! ht2 = incremento de tempo para integracao [s]

1.e6 ! tend = tempo final de integracao [s]

0 ! iperiod = 0 - simula até regime permanente; 1 - simula até tend

1.e-6 ! tol1 = tolerancia para o regime permanente