

```

      subroutine rkf4st(f,x,xout,y,h,n,eps,iflag)
      implicit double precision (a-h,o-z)
      dimension y(2054),yp1(2054),yst1(2054),yp2(2054),yst2(2054)
      dimension yst3(2054),yp3(2054),yst4(2054),yp4(2054)
      common/ref/iwrite
      external f
c
c      programmer    a.segal
c      version 1    date    12-02-82
c
c
c      copyright (c) 1982  "ingenieursbureau sepra"
c      permission to copy or distribute this software or documentation
c      in hard copy or soft copy granted only by written license
c      obtained from "ingenieursbureau sepra".
c      all rights reserved. no part of this publication may be reproduced,
c      stored in a retrieval system ( e.g., in memory, disk, or core)
c      or be transmitted by any means, electronic, mechanical, photocopy,
c      recording, or otherwise, without written permission from the
c      publisher.
c
      ifail=0
      if (iflag.ne.1) goto 100
c  **  control input
      if (n.lt.1) iflag=5
      if (n.gt.2054) iflag=5
      if (xout.lt.x) iflag=5
      if (eps.gt.1.0d-12) goto 2
      eps=1.0d-12
      write(iwrite,1)
1      format(5x,'eps too small, new eps=1.0d-12')
2      if (iflag.eq.5) goto 3
c  **  initialize , compute appropriate step
      call f(x,y,yp1)
      b=1.0d-15
      do 10 i=1,n
         b=b+yp1(i)*yp1(i)
10     continue
      a=dabs(eps/dsqrt(b))
      h=dsqrt(a)
      iflag=2
      goto 105
3      write(iwrite,4)
4      format(5x,'wrong input')
      return
c  **  end of start-phase
100    call f(x,y,yp1)
105    if (h.gt.(xout-x)) h=xout-x+1.0d-15
      if (h.gt.(0.2d-15)) goto 110
c  **  h smaller than 0.2d-14 hence extrapolation
      do 108 i=1,n
         yst4(i)=y(i)+h*yp1(i)
108    continue
      goto 200
110    do 120 i=1,n
         yst1(i)=y(i)+h*yp1(i)/2.0d0
120    continue
      t=x+h/2d0
      call f(t,yst1,yp2)
      do 125 i=1,n
         yst2(i)=y(i)+.75d0*h*yp2(i)
125    continue
      t=x+.75d0*h
      call f(t,yst2,yp3)
      do 130 i=1,n
         yst3(i)=y(i)+h*(2*yp1(i)+3*yp2(i)+4*yp3(i))/9.0d0
130    continue
      do 135 i=1,n
         yst2(i)=y(i)+h*yp2(i)/2.0d0
135    continue
      t=x+h/2.0d0
      call f(t,yst2,yp3)
      do 140 i=1,n

```

```

        yst2(i)=y(i)+h*yp3(i)
140  continue
        t=x+h
        call f(t,yst2,yp4)
        do 145 i=1,n
            yst4(i)=y(i)+h*(yp1(i)+2*yp2(i)+2*yp3(i)+yp4(i))/6.0d0
145  continue
        b=1.0d-15
        do 150 i=1,n
            a=dabs(yst4(i)-yst3(i))
            if (a.gt.b) b=a
150  continue
        b=2*b
        alva=b/eps
        if (alva.gt.1.0d4) alva=1.0d4
        if (alva.lt.1.0d-4) alva=1.0d-4
        hn=dsqrt(alva)
        hn=h/dsqrt(hn)
        if (b.lt.eps) goto 200
c  **  accuracy test failed : new stepsize
        ifail=ifail+1
        if (ifail.eq.3) goto 300
        h=.9*hn
        goto 110
200  continue
c  **  accuracy test passed
        do 210 i=1,n
            y(i)=yst4(i)
210  continue
        x=x+h
        b=xout-x
        if (b.lt.(1.0d-15)) iflag=3
        h=hn
        return
300  continue
c  **  more than three failures at this point
        iflag=4
        write(iwrite,310) x
310  format(5x,'three failures at time=',f12.4)
        return
end

```

□