Differentiation and Integration

曹玉清 天眷1701班 U201710185

— Repeated Simpson quadrature and Repeated trapezoid quadrature

1、问题描述



Homework



Write a program to compute the integral

$$I(f) = \int_{1}^{5} \sin(x) dx$$
 $h = 0.1$

with the following methods

- > repeated Simpson quadrature
- > repeated trapezoid quadrature

and provide the errors



2、主要算法与公式





Repeated trapezoid quadrature

$$h = \frac{b-a}{n}, x_i = a+ih, i = 0,\dots, n$$
$$\int_{x_i}^{x_{i+1}} f(x) = \frac{h}{2} (f(x_i) + f(x_{i+1}))$$

$$T_n(f) = \sum_{i=0}^{n-1} \left(\frac{h}{2} (f(x_i) + f(x_{i+1})) \right)$$
$$= \frac{h}{2} \left(f(a) + 2 \sum_{i=1}^{n-1} f(x_i) + f(b) \right)$$

N.



Repeated Simpson quadrature

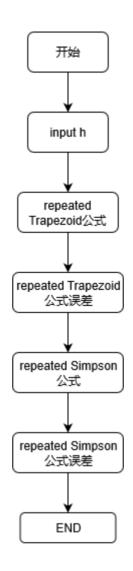
$$h=\frac{b-a}{n}, x_i=a+ih, i=0,\cdots,n; n=2m$$

$$\int_{x_{2i}}^{x_{2i+2}} f(x) = \frac{2h}{6} \left(f(x_{2i}) + 4f(x_{2i+1}) + f(x_{2i+2}) \right)$$

$$S_n(f) = \frac{h}{3} \left(f(a) + 4 \sum_{i=0}^{m-1} f(x_{2i+1}) + 2 \sum_{i=1}^{m-1} f(x_{2i}) + f(b) \right)$$



3、流程图



3、源代码

```
1
    program ST
    implicit none
 2
 3
        real*8 :: sum_fxk,h,sum_odd,sum_even
        real*8 :: trape_40,simp_40
4
 5
        real*8 :: trape_20,simp_20
        real*8 :: errorTrape,errorSimp
 6
 7
        integer :: k
 8
        h = 0.1
9
10
        sum_fxk = 0.0
        do k = 1,39
11
            sum_fxk = sum_fxk + sin(1.0 + k*h)
12
13
        trape_{40} = 1/20.0 * (sin(1.0) + 2*sum_fxk + sin(5.0))
14
15
16
        sum fxk = 0.0
        do k = 1,19
17
            sum_fxk = sum_fxk + sin(1.0 + k*h)
18
19
        end do
```

```
20
        trape_20 = 1/20.0 * (sin(1.0) + 2*sum_fxk + sin(5.0))
21
        errorTrape = 1/3.0 * (trape_40 -trape_20)
22
        print "(a)", "using repeated trapezoid quadrature:"
23
24
        print "(f8.6)", trape_40
25
        print "(a)", "repeated trapezoid quadrature error analysis:"
26
        print *, errorTrape
27
        print "(a)", "-----
28
29
        sum odd = 0.0
30
        sum_even = 0.0
        k = 1
31
32
        do while (k \le 39)
33
            sum odd = sum odd + sin(1.0+k*h)
            k = k+2
34
        end do
35
        k = 2
36
37
        do while (k \le 39)
38
            sum even = sum even + sin(1.0+k*h)
            k = k+2
39
40
        end do
        simp_40 = 1/30.0 * (sin(1.0) + 4*sum_odd + 2*sum_even + sin(5.0))
41
42
43
        sum odd = 0.0
44
        sum even = 0.0
45
        k = 1
46
        do while (k<=19)
47
            sum odd = sum odd + sin(1.0+k*h)
            k = k+2
48
49
        end do
50
        k = 2
        do while (k<=19)
51
52
            sum_even = sum_even + sin(1.0+k*h)
53
            k = k+2
54
        end do
        simp 20 = 1/30.0 * (sin(1.0) + 4*sum odd + 2*sum even +sin(5.0))
55
        errorSimp = 1/15.0 * (simp_40 - simp_20)
56
57
        print "(a)", "using repeated Simpson quadrature:"
58
        print "(f8.6)", simp_40
59
60
        print "(a)", "repeated Simpson quadrature error analysis:"
61
        print *, errorSimp
62
63
    end program ST
```

4、运行时结果

E:\cyg\cyg essays\Phy HUST\计算物理\hw\cyg\5\code>ST. exe using repeated trapezoid quadrature: 0.256426

repeated trapezoid quadrature error analysis:

-0. 40586367038482418

using repeated Simpson quadrature:

0.256640

repeated Simpson quadrature error analysis:

-8. 2465828046232370E-002

5、误差分析与画图

repeated trapezoid guadrature的绝对误差是 0.405864

repeated Simpson guadrature的绝对误差是 0.082466

Ordinary Differential Equation

— Basic Euler and improved Euler method

1、问题描述







Write a program to solve the following ordinary differential equation by 1) basic Euler 2) improved **Euler** method

$$\begin{cases} y' = -x^2 y^2 \\ y(0) = 3 \end{cases} \qquad (0 \le x \le 1.5)$$

and calculate y(1.5) with stepsize=0.1, 0.1/2, 0.1/4, 0.1/8

Compare it with analytic solution (in figure)

$$y(x) = 3/(1+x^3)$$
 CPCM



2、主要算法与公式





Error for basic Euler method

$$y(x_{n+1}) - y_{n+1}^{(h)} = O(h^{2}) \approx ch^{2} \quad \text{step size } h$$

$$y(x_{n+1}) - y_{n+1}^{(h/2)} = 2O\left((h/2)^{2}\right) \approx 2c\left(h/2\right)^{2} \quad \text{step size } h/2$$

$$\frac{y(x_{n+1}) - y_{n+1}^{(h/2)}}{y(x_{n+1}) - y_{n+1}^{(h)}} \approx \frac{1}{2} \quad \text{step size } h/2$$

$$y(x_{n+1}) = 2y_{n+1}^{(h/2)} - y_{n+1}^{(h)}$$

$$y(x_{n+1}) - y_{n+1}^{(h/2)} \approx y_{n+1}^{(h/2)} - y_{n+1}^{(h)}$$

$$y(x_{n+1}) - y_{n+1}^{(h)} \approx 2(y_{n+1}^{(h/2)} - y_{n+1}^{(h)}) \quad \text{CP}_{16}M$$





Error for basic Euler method

This is only the error of one rectangle integration

$$y(x_{n+1}) - y_{n+1}^{(h)} \approx 2(y_{n+1}^{(h/2)} - y_{n+1}^{(h)})$$

The total error for y_{n+1} (n+1 rectangles) is

$$E(h) = y(x_{n+1}) - y_{n+1}^{(h)} = \sum_{i=1}^{n+1} 2(y_i^{(h/2)} - y_i^{(h)})$$







Error for improved Euler method

$$y(x_{n+1}) - y_{n+1}^{(h)} = O(h^3) \approx ch^3$$

step size h

$$y(x_{n+1}) - y_{n+1}^{(h/2)} = 2O((h/2)^{2}) \approx 2c(h/2)^{3}$$

$$\frac{y(x_{n+1}) - y_{n+1}^{(h/2)}}{y(x_{n+1}) - y_{n+1}^{(h)}} \approx \frac{1}{4}$$

$$y(x_{n+1}) = \frac{4y_{n+1}^{(h/2)} - y_{n+1}^{(h)}}{3}$$

$$\begin{cases} y(x_{n+1}) - y_{n+1}^{(h/2)} \approx \frac{y_{n+1}^{(h/2)} - y_{n+1}^{(h)}}{3} \\ y(x_{n+1}) - y_{n+1}^{(h)} \approx \frac{4}{3}(y_{n+1}^{(h/2)} - y_{n+1}^{(h)}) \end{cases}$$

$$\text{CP6M}$$





Error for improve Euler method

This is only the error of one trapezoid integration

$$y(x_{n+1}) - y_{n+1}^{(h)} \approx \frac{4}{3} (y_{n+1}^{(h/2)} - y_{n+1}^{(h)})$$

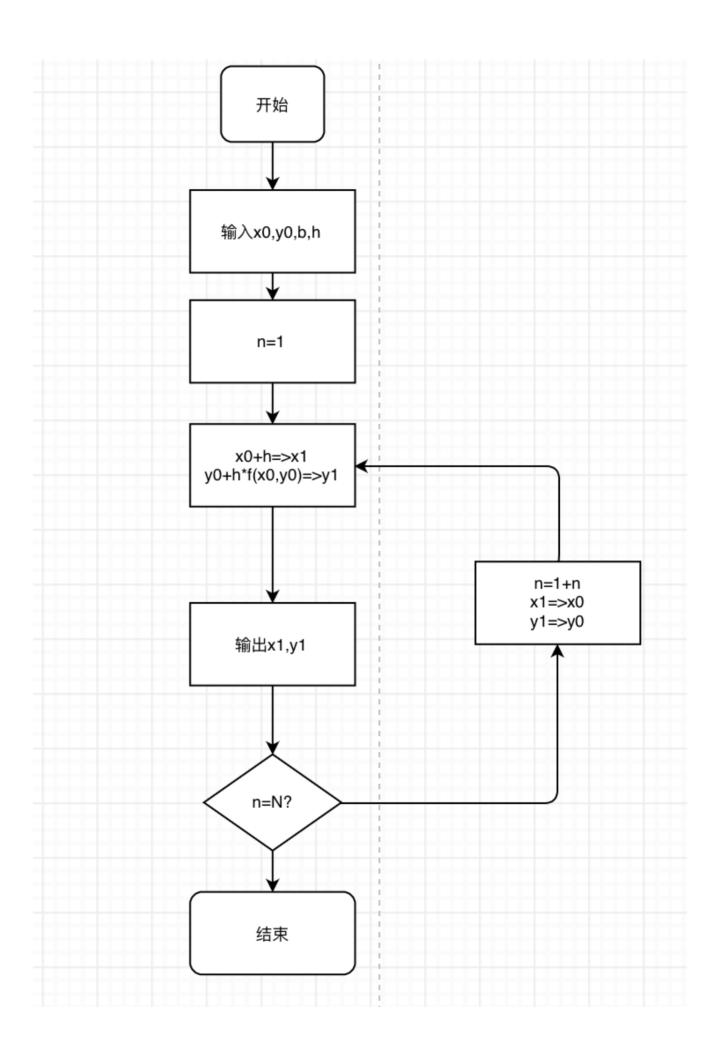
The total error for y_{n+1} (n+1 trapezoids) is

$$E(h) = y(x_{n+1}) - y_{n+1}^{(h)} = \sum_{i=1}^{n+1} \frac{4}{3} \left(y_i^{(h/2)} - y_i^{(h)} \right)$$

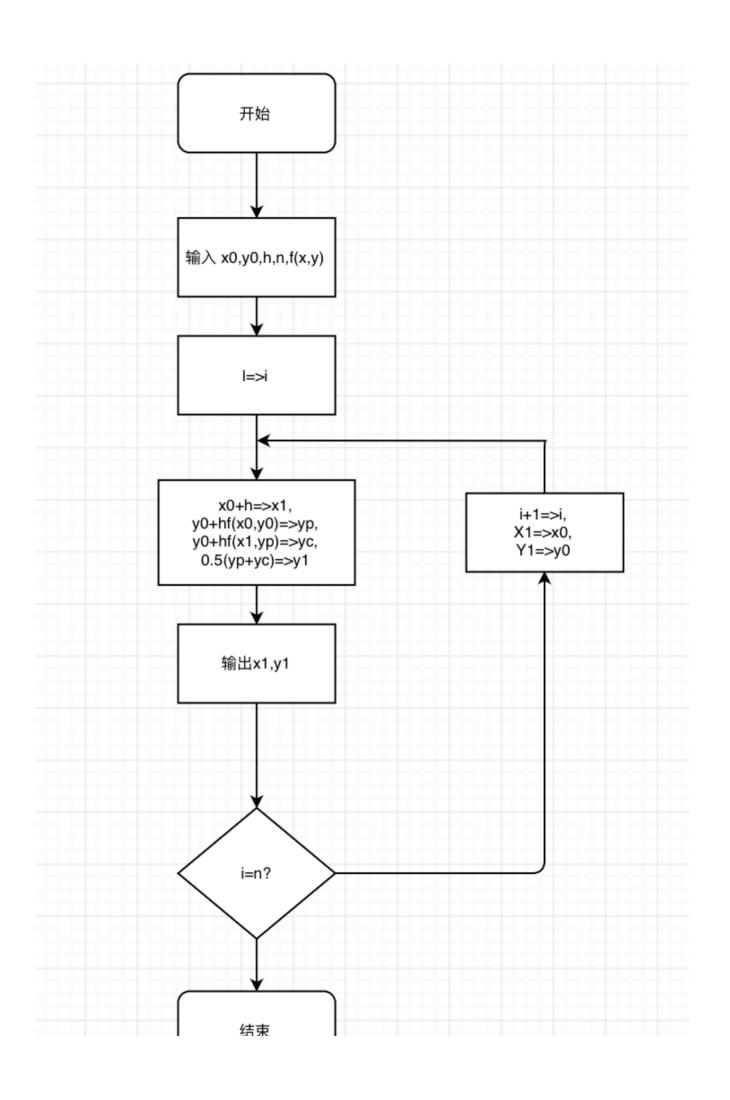


3、流程图

1) Basic Euler method



| 2) | improved Euler method |
|----|-----------------------|
| | |
| | |
| | |



3、源代码

```
1
    program RK
 2
    implicit none
 3
 4
        real*8 :: fxy
 5
        real*8 :: h
 6
        real*8 :: K1, K2, K3, K4
 7
 8
        real*8 :: xn
 9
        integer :: interval
10
        real*8 :: y1(16),y2(31),y3(61),y4(121)
11
        h = 0.1
12
        y1(1) = 3.0
13
        do interval = 1,15
14
15
            xn = 0.0 + interval * h
            K1 = fxy( xn,y1(interval) )
16
            K2 = fxy(xn+h/2.0, y1(interval)+h/2.0*K1)
17
            K3 = fxy(xn+h/2.0, y1(interval)+h/2.0*K2)
18
            K4 = fxy(xn+h, y1(interval)+h*K3)
19
20
           y1(interval+1) = y1(interval) + h/6.0 * (K1+2*K2+2*K3+K4)
        end do
21
22
23
        print "(a)", "when stepsize = 0.1, y= :"
24
        print "(16f8.3)", y1(:)
25
        print "(a)", "-----
26
        h = 0.1/2.0
27
        y2(1) = 3.0
28
29
        do interval = 1,30
30
           xn = 0.0 + interval * h
31
            K1 = fxy( xn,y2(interval) )
            K2 = fxy(xn+h/2.0, y2(interval)+h/2.0*K1)
32
            K3 = fxy(xn+h/2.0, y2(interval)+h/2.0*K2)
33
            K4 = fxy(xn+h, y2(interval)+h*K3)
34
35
           y2(interval+1) = y2(interval) + h/6.0 * (K1+2*K2+2*K3+K4)
        end do
36
37
        print "(a)", "when stepsize = 0.1, y= :"
38
        print "(31f8.3)", y2(:)
39
40
        print "(a)", "------
41
42
        h = 0.1/4.0
43
        y3(1) = 3.0
        do interval = 1,60
44
45
           xn = 0.0 + interval * h
46
            K1 = fxy( xn,y3(interval) )
47
            K2 = fxy(xn+h/2.0, y3(interval)+h/2.0*K1)
```

```
K3 = fxy(xn+h/2.0, y3(interval)+h/2.0*K2)
48
49
            K4 = fxy(xn+h, y3(interval)+h*K3)
50
            y3(interval+1) = y3(interval) + h/6.0 * (K1+2*K2+2*K3+K4)
        end do
51
52
        print "(a)", "when stepsize = 0.1, y= :"
53
        print "(31f8.3)", y3(:)
54
55
        print "(a)", "-----
56
57
        h = 0.1/8.0
        y4(1) = 3.0
58
59
        do interval = 1,120
60
            xn = 0.0 + interval * h
            K1 = fxy( xn,y4(interval) )
61
            K2 = fxy(xn+h/2.0, y4(interval)+h/2.0*K1)
62
            K3 = fxy(xn+h/2.0, y4(interval)+h/2.0*K2)
63
            K4 = fxy(xn+h, y4(interval)+h*K3)
64
            y4(interval+1) = y4(interval) + h/6.0 * (K1+2*K2+2*K3+K4)
65
66
67
        print "(a)", "when stepsize = 0.1, y= :"
68
        print "(31f8.3)", y4(:)
69
70
        print "(a)", "-----
71
72
    end program RK
73
74
    function fxy(x,y)
75
    implicit none
76
        real*8 :: x,y
77
        real*8 :: fxy
78
        fxy = -x^{**2} * y^{**2}
79
    end function
```

4、运行时结果

```
E:\cyq\cyq essays\Phy HUST\计算物理\hw\cyq\5\code>RK.exe when stepsize = 0.1, y=: 3.000 2.979 2.924 2.822 2.669 2.469 2.235
                                              2. 669 2. 469 2. 235 1. 985
                                                                                          1.736
                                                                                                    1.501
                                                                                                                1.288
                                                                                                                                     0.939
                                                                                                                                                0.802
                                                                                                                                                          0.686
                                                                                                                                                                      0.589
when stepsize = 0.1, y=
  3. 000 2. 997
1. 859 1. 735
                                                                                         2.750
1.016
                                                                                                   2.667
0.938
                                                                                                                         2. 467
0. 801
                                                                                                                                   2. 354
0. 741
                                               2.954
                                                         2.921
                                                                                                                                               2.234
                                                                                                                                                                      1.984
                       1.615
                                   1.500
                                                        1.287
                                                                   1.190
                                                                              1.100
                                                                                                              0.867
                                                                                                                                              0.686
                                              1.390
when stepsize = 0.1, y=
3.000 3.000 2.999
                                                                                                                                                          2.850
                                               2.994
                                                                                          2.966
                                                                                                               2.939
                                                                    2.984
                                                                               2.976
                                                                                                                                                                      2.820
            2. 749
1. 921
                       2.710
                                                                   2. 521
                                                                             2. 467
1. 557
                                                                                        2. 411
1. 500
                                                                                                   2. 354
2. 354
1. 444
                                              2.621
                                   2, 667
  2.786
                                               1. 735
                                                          1. 675
                                                                     1. 615
                        1.016
                                                        0.902
                                                                                         0.801
when stepsize = 0.1, y=
                                                                                                                          2.990
                                                                                                                                     2.987
                                    3.000
                                               2.999
                                                          2.999
                                                                     2.998
                                                                               2.997
                                                                                                     2.994
                                                                                                                                                2.984
                                                                                                                                                                      2.976
                                                                                                                                              2. 850
2. 547
                                                                                                                         2. 877
2. 597
                                                                                                                                    2.864
                        2.960
                                   2. 954
2. 768
             2,966
                                                                                                                                                          2.835
                                              2.947
                                                        2, 939
                                                                   2.930
                                                                               2. 689
               2.803
                                                                                                                                                                      2.494
                                                                                                                                              2. 079
1. 615
                                   1. 921
1. 444
                                                                   1.827
1.364
                                                                              1. 796
1. 338
                                                                                                                         1.675
1.238
                                                                                                                                    1.645
1.214
    2.016
             1.500
                                              1.417
                        1.472
                                                                                                                                               1, 190
                                                                                                                                                          1.167
             1. 122
                                                                   1. 016
0. 741
                                                                              0. 996
0. 727
                                                                                                                                                0.884
                        0.801
  0.833
                                   0.786
```

二、Runge-Kutta method

1、问题描述



Homework



Write a program to solve the following ordinary differential equation by four-order *Runge-Kutta* method

$$\begin{cases} y' = -x^2 y^2 & (0 \le x \le 1.5) \\ y(0) = 3 & \end{cases}$$

and calculate y(1.5) with stepsize=0.1, 0.1/2, 0.1/4, 0.1/8

Compare it with analytic solution (in figure)

$$y(x) = 3/(1+x^3)$$
 CPCM

2、主要算法与公式





=

Runge-Kutta Method with 4-order precision

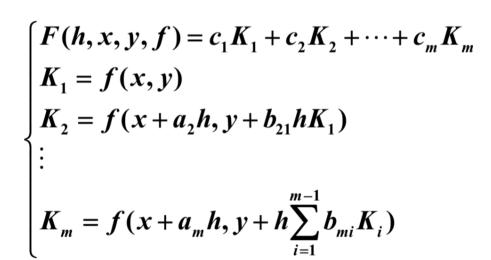
$$\begin{cases} y_{n+1} = y_n + \frac{h}{6}(K_1 + 2K_2 + 2K_3 + K_4) \\ K_1 = f(x_n, y_n) \\ K_2 = f(x_n + \frac{h}{2}, y_n + \frac{h}{2}K_1) \\ K_3 = f(x_n + \frac{h}{2}, y_n + \frac{h}{2}K_2) \\ K_4 = f(x_n + h, y_n + hK_3) \end{cases}$$



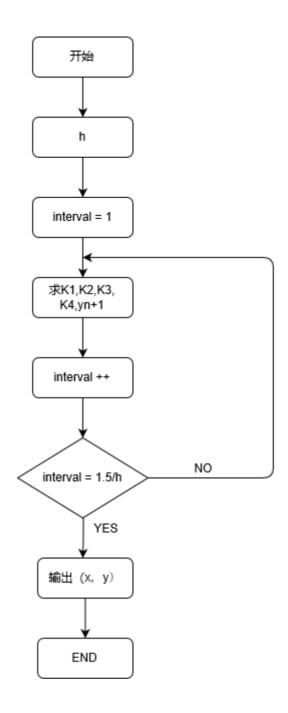




Runge-Kutta Method with m-order precision







3、源代码

```
1
    program RK
2
    implicit none
 3
4
        real*8 :: fxy
5
        real*8 :: h
 6
7
        real*8 :: K1,K2,K3,K4
        real*8 :: xn
 8
9
        integer :: interval
10
        real*8 :: y1(16),y2(31),y3(61),y4(121)
11
        h = 0.1
12
        y1(1) = 3.0
13
14
        do interval = 1,15
```

```
xn = 0.0 + interval * h
15
16
            K1 = fxy( xn,y1(interval) )
17
            K2 = fxy(xn+h/2.0, y1(interval)+h/2.0*K1)
           K3 = fxy(xn+h/2.0, y1(interval)+h/2.0*K2)
18
19
            K4 = fxy(xn+h, y1(interval)+h*K3)
20
           y1(interval+1) = y1(interval) + h/6.0 * (K1+2*K2+2*K3+K4)
21
        end do
22
23
        print "(a)", "when stepsize = 0.1, y= :"
        print "(16f8.3)", y1(:)
24
        print "(a)", "-----
25
26
27
        h = 0.1/2.0
28
        y2(1) = 3.0
29
        do interval = 1,30
           xn = 0.0 + interval * h
30
            K1 = fxy( xn,y2(interval) )
31
32
            K2 = fxy(xn+h/2.0, y2(interval)+h/2.0*K1)
33
            K3 = fxy(xn+h/2.0, y2(interval)+h/2.0*K2)
            K4 = fxy(xn+h, y2(interval)+h*K3)
34
35
           y2(interval+1) = y2(interval) + h/6.0 * (K1+2*K2+2*K3+K4)
        end do
36
37
        print "(a)", "when stepsize = 0.1, y= :"
38
39
        print "(31f8.3)", y2(:)
40
        print "(a)", "-----
41
42
        h = 0.1/4.0
43
        v3(1) = 3.0
        do interval = 1,60
44
45
           xn = 0.0 + interval * h
            K1 = fxy( xn,y3(interval) )
46
            K2 = fxy(xn+h/2.0, y3(interval)+h/2.0*K1)
47
48
            K3 = fxy(xn+h/2.0, y3(interval)+h/2.0*K2)
49
            K4 = fxy(xn+h, y3(interval)+h*K3)
50
           y3(interval+1) = y3(interval) + h/6.0 * (K1+2*K2+2*K3+K4)
        end do
51
52
53
        print "(a)", "when stepsize = 0.1, y= :"
54
        print "(31f8.3)", y3(:)
55
        print "(a)", "-----
56
        h = 0.1/8.0
57
        y4(1) = 3.0
58
59
        do interval = 1,120
60
           xn = 0.0 + interval * h
            K1 = fxy( xn,y4(interval) )
           K2 = fxy(xn+h/2.0, y4(interval)+h/2.0*K1)
62
            K3 = fxy(xn+h/2.0, y4(interval)+h/2.0*K2)
63
            K4 = fxy(xn+h, y4(interval)+h*K3)
64
65
           y4(interval+1) = y4(interval) + h/6.0 * (K1+2*K2+2*K3+K4)
        end do
66
67
```

```
print "(a)", "when stepsize = 0.1, y= :"
68
69
     print "(31f8.3)", y4(:)
     print "(a)", "-----
70
71
72
  end program RK
73
  function fxy(x,y)
74
75
   implicit none
    real*8 :: x,y
76
77
     real*8 :: fxy
    fxy = -x^{**2} * y^{**2}
78
79 end function
```

4、运行时结果

| when step | E:\cyq\cyq essays\Phy HUST\计算物理\hw\cyq\5\code>RK.exe when stepsize = 0.1, y=: 3.000 2.979 2.924 2.822 2.669 2.469 2.235 1.985 1.736 1.501 1.288 1.100 0.939 0.802 0.686 0.589 | | | | | | | | | | | | | | 0 500 |
|-----------------------------|---|-------------------------------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|------------------|--------|
| 3. 000 | 2. 979 | 2.924 | 2.822 | 2.009 | 2. 409 | 2. 235 | 1. 985 | 1. 736 | 1. 501 | 1. 288 | 1. 100 | 0. 939 | 0.802 | 0. 686 | 0. 589 |
| when step 3.000 1.859 | 2.997 1.735 | .1, y=: 2.990 1.615 | 2. 977 1. 500 | 2. 954 1. 390 | 2. 921 1. 287 | 2. 877 1. 190 | 2. 820 1. 100 | 2.750 1.016 | 2. 667 0. 938 | 2. 572 0. 867 | 2. 467 0. 801 | 2. 354 0. 741 | 2. 234 0. 686 | 2. 110 0. 635 | 1. 984 |
| when step | when stepsize = 0.1, y= : | | | | | | | | | | | | | | |
| 3. 000 | 3. 000 | 2. 999 | 2.997 | 2.994 | 2.990 | 2.984 | 2.976 | 2.966 | 2.954 | 2.939 | 2.921 | 2.900 | 2.877 | 2.850 | 2.820 |
| 2. 786 | 2.749 | 2.710 | 2.667 | 2.621 | 2. 572 | 2. 521 | 2.467 | 2. 411 | 2. 354 | 2. 294 | 2. 234 | 2. 172 | 2. 110 | 2.047 | |
| 1. 984 1. 100 | 1. 921 1. 057 | 1. 859 1. 016 | 1. 797 0. 976 | 1. 735 0. 938 | 1. 675 0. 902 | 1. 615 0. 867 | 1. 557 0. 833 | 1. 500 0. 801 | 1. 444 0. 770 | 1. 390 0. 741 | 1. 338 0. 713 | 1. 287 0. 686 | 1. 238 0. 660 | 1. 190 | 1. 144 |
| 1 | · | | | | | | | | | | | | | | |
| when step 3.000 2.971 | 3.000 2.966 | . 1, y= : 3. 000 2. 960 | 3. 000 2. 954 | 2. 999 2. 947 | 2. 999 2. 939 | 2. 998 2. 930 | 2. 997 2. 921 | 2. 996 2. 911 | 2. 994 2. 900 | 2. 992 2. 889 | 2. 990 2. 877 | 2. 987 2. 864 | 2. 984 2. 850 | 2. 980 2. 835 | 2. 976 |
| 2.820 | 2.803 | 2.786 | 2.768 | 2.749 | 2.730 | 2.710 | 2.689 | 2.667 | 2.644 | 2.621 | 2.597 | 2.572 | 2.547 | 2.521 | 2.494 |
| 2. 467 | 2. 439 | 2. 411 | 2. 383 | 2.354 | 2. 324 | 2. 294 | 2. 264 | 2. 234 | 2. 203 | 2. 172 | 2. 141 | 2. 110 | 2.079 | 2.047 | |
| 2. 016 1. 528 | 1. 984 1. 500 | 1. 953 1. 472 | 1. 921 1. 444 | 1. 890 1. 417 | 1.859 1.390 | 1.827 1.364 | 1. 796 1. 338 | 1. 766 1. 312 | 1. 735 1. 287 | 1. 705 1. 262 | 1. 675 1. 238 | 1. 645 1. 214 | 1. 615 1. 190 | 1. 586 1. 167 | 1. 557 |
| 1. 128 | 1. 122 | 1. 100 | 1. 078 | 1. 417 | 1. 036 | 1. 016 | 0. 996 | 0.976 | 0. 957 | 0. 938 | 0.920 | 0.902 | 0.884 | 0.867 | 0.850 |
| 0.833 | 0.817 | 0.801 | 0. 786 | 0. 770 | 0. 756 | 0. 741 | 0. 727 | 0.713 | 0. 699 | 0.686 | 0. 673 | 0. 502 | 0.004 | 0.001 | 0.000 |
| | | | | | | | | | | | | | | | |
| | | | | | | | | | | | | | | | |