数値計算演習

渡部重十
shw@ep.sci.hokudai.ac.jp

１．目的

・ UNIX ワークステーションの環境を理解し利用できる．
・ メールの送受信，telnet，ftp を使える．
・ 基本コマンド，ls，mv，rm，cp，cd，grep，cat，less,... を使える．
・ エディタ，vi，emacs,... を使える．
・ FORTRAN でプログラムを作成できる．
・ GNUPLOT で図を作成できる．
・ なぜ数値計算が必要か理解する．

２．内容

連立１次方程式と逆行列
　　消去法（ガウス法）
　　反復法（ガウス・ザイデル法）
　　逆行列
代数方程式
　　ニュートン法
　　はさみうち法
数値積分
　　台形公式
　　シンプソンの公式
　　フーリエ級数
常微分方程式
　　ルンゲ・クッタ法
　　連立微分方程式
ポアソン方程式・ラプラス方程式
　　差分法
偏微分方程式
　　熱伝導の方程式
　　波動方程式

3. On WEB

http://www.ep.sci.hokudai.ac.jp/~shw/computer.pdf

UNIX 基本コマンド

コマンド	機能	使用例
ls		ls, ls –l, ls –al
cp	コピー	cp file1 file2
mv	ファイル名の変更	mv file1 file2
rm	ファイルやディレクトリの削除	rm file1, rm –r direct1
less		less file1
tar	アーカイブの解凍・作成	tar tvf file1.tar, tar xvf file1.tar
mkdir	ディレクトリの作成	mkdir direct2
cd	ディレクトリの変更	cd, cd /usr/local
pwd	現ディレクトリの表示	pwd
find	ファイルやディレクトリの検索	find / -name file1
grep	文字検索	grep name file1
telnet		telnet comp1.hokudai.ac.jp
ftp		ftp comp1.hokudai.ac.jp
		ftp –i comp1.hokudai.ac.jp

emacs の基本操作

起動方法	emacs filename
新しいウィンドウを作らずに起動	emacs –nw filename
カーソルの移動	上 Ctrl-+p　下 Ctrl+n　左 Ctrl+b　右 Ctrl+f
1 画面後ろへ	Ctrl+v
1 画面前へ	Esc+v
1 文字削除	Ctrl+d
カーソルから行の最後まで削除	Ctrl+k
移動とコピー	Ctrl+y
ファイルの後方へサーチ	Ctrl+s string
ファイルへ保存	Ctrl+x Ctrl+s
ファイル名を指定して保存	Ctrl+x Ctrl+w filename
コマンドの実行中止	Ctrl+g
終了	Ctrl+x Ctrl+c

vi の基本操作

起動方法	vi filename
カーソルの移動	**上** j **下** k **左** h **右** l
1 **画面後ろへ**	Ctrl+f
1 **画面前へ**	Ctrl+b
1 **文字削除**	x
行の削除	dd
コピー	yy
ペースト	p
挿入	i
アペンド	a
ファイルの後方へサーチ	/string
ファイルへ保存	:w
ファイル名を指定して保存	:w filename
コマンドの実行モードへ	ESC
終了	:q
セーブして終了	:wq
変更しないで終了	:q!

FORTRAN **プログラムのコンパイル**	f77 file1.f
	f77 file1.f –o file1
	g77 file1.f
	g77 file1.f –o file1
	g90 file1.f
	g90 file1.f –o file1
実行	./a.out
	./file1
	./a.out > outfile
	./file1 > outfile
	./a.out < infile
GNUPLOT **プログラムの実行**	gnuplot file2

課題１　連立１次方程式

$$2x_1 + 3x_2 + x_3 - 3x_4 = 1$$

$$-x_1 + 2x_2 + 2x_3 + 4x_4 = 6$$

$$4x_1 + x_2 - 3x_3 + 5x_4 = 3$$

$$5x_1 - 4x_2 - 4x_3 + x_4 = 3$$

をガウスの消去法で解け.

（解法） n **元連立１次方程式は次のように表すことができる.**

$$\begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n-1} & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n-1} & a_{2n} \\ & & \dots & & \\ & & \dots & & \\ a_{n1} & a_{n2} & \dots & a_{nn-1} & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \dots \\ x_{n-1} \\ x_n \end{bmatrix} = \begin{bmatrix} t_1 \\ t_2 \\ \dots \\ t_{n-1} \\ t_n \end{bmatrix}$$

１列を a_{11} **で割ると**

$$\begin{bmatrix} 1 & a_{12}/a_{11} & \dots & a_{1n-1}/a_{11} & a_{1n}/a_{11} \\ a_{21} & a_{22} & \dots & a_{2n-1} & a_{2n} \\ & & \dots & & \\ & & \dots & & \\ a_{n1} & a_{n2} & \dots & a_{nn-1} & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \dots \\ x_{n-1} \\ x_n \end{bmatrix} = \begin{bmatrix} t_1/a_{11} \\ t_2 \\ \dots \\ t_{n-1} \\ t_n \end{bmatrix}$$

１列を a_{21} **倍して２列から引くと**

$$\begin{bmatrix} 1 & a_{12}/a_{11} & \dots & a_{1n-1}/a_{11} & a_{1n}/a_{11} \\ 0 & a_{22} - a_{21}a_{12}/a_{11} & \dots & a_{2n-1} - a_{21}a_{1n-1}/a_{11} & a_{2n} - a_{21}a_{1n}/a_{11} \\ & & \dots & & \\ & & \dots & & \\ a_{n1} & a_{n2} & \dots & a_{nn-1} & a_{nn} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \dots \\ x_{n-1} \\ x_n \end{bmatrix} = \begin{bmatrix} t_1/a_{11} \\ t_2 - a_{21}t_1/a_{11} \\ \dots \\ t_{n-1} \\ t_n \end{bmatrix}$$

同様にして計算し，最終的に

$$\begin{bmatrix} 1 & 0 & \dots & 0 & 0 \\ 0 & 1 & \dots & 0 & 0 \\ & & \dots & & \\ & & \dots & & \\ 0 & 0 & \dots & 0 & 1 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ \dots \\ x_{n-1} \\ x_n \end{bmatrix} = \begin{bmatrix} A_1 \\ A_2 \\ \dots \\ A_{n-1} \\ A_n \end{bmatrix}$$

となったとき， $A_1, A_2, \dots A_{n-1}, A_n$ が答えである．

（FORTRAN プログラム例）

```
c     Gauss method
c
       real*4 a(4,5)
c
       a(1,1) =  2.0
       a(1,2) =  3.0
       a(1,3) =  1.0
       a(1,4) = -3.0
       a(1,5) =  1.0
       a(2,1) = -1.0
       a(2,2) =  2.0
       a(2,3) =  2.0
       a(2,4) =  4.0
       a(2,5) =  6.0
       a(3,1) =  4.0
       a(3,2) =  1.0
       a(3,3) = -3.0
       a(3,4) =  5.0
       a(3,5) =  3.0
       a(4,1) =  5.0
       a(4,2) = -4.0
       a(4,3) = -4.0
       a(4,4) =  1.0
       a(4,5) =  3.0
c
       do k=1,4
          p = a(k,k)
          do j=k,5
```

```
         a(k,j) = a(k,j)/p
      enddo
      do i=1,4
         if( i .ne. k ) then
            aik = a(i,k)
            do j=k,5
               a(i,j) = a(i,j) - aik*a(k,j)
            enddo
         endif
      enddo
   enddo
   do i=1,4
      write(6,*) i,'   ',(a(i,j),j=1,5)
   enddo
   stop
   end
```

（実行結果）

```
1      1.  0.  0.  0.  1.99999952
2      0.  1.  0.  0. -0.999999523
3      0.  0.  1.  0.  2.99999928
4      0.  0.  0.  1.  0.999999881
```

課題２　連立１次方程式

$$4x_1 - x_2 + x_3 = 4$$

$$x_1 + 6x_2 + 2x_3 = 9$$

$$-x_1 - 2x_2 + 5x_3 = 2$$

をガウス・ザイデルの反復法で解け．

（解法）n 元連立１次方程式は次のように変形することができる．

$$x_1^{(k+1)} = \frac{1}{a_{11}}\left(t_1 - a_{12}x_2^{(k)} - a_{13}x_3^{(k)} - \ldots - a_{1n}x_n^{(k)}\right)$$

$$x_2^{(k+1)} = \frac{1}{a_{22}}\left(t_2 - a_{21}x_1^{(k+1)} - a_{23}x_3^{(k)} - \ldots - a_{2n}x_n^{(k)}\right)$$

・・・

$$x_n^{(k+1)} = \frac{1}{a_{nn}}\left(t_n - a_{n1}x_2^{(k+1)} - a_{n2}x_2^{(k+1)} - \ldots - a_{nn-1}x_{n-1}^{(k+1)}\right)$$

$x_i^{(0)}$ に任意の初期値を入れて反復計算を実行する．数値解は

$$\left|x_i^{(k+1)} - x_i^{(k)}\right| < e$$

で判定する．対角要素が他の要素より大きいとき，ガウス・ザイデル法は収束する．

消去法
・方程式の数で演算回数が決まる．
・誤差が積み重ねられる．
反復法
・収束しない場合がある．
・演算回数が多い．
・０を多く含んだ方程式では演算回数が大幅に減り，有効である．

（FORTRAN プログラム例）

```
c    EX2
c    Gauss-Seidel method
c
      real*4  x(3), xb(3)
      real*4  a(3,4)
c
      read(5,*) ((a(i,j),j=1,4),i=1,3)
```

```
c
      do i=1,3
         x(i) = 0.0
      enddo
c
      do n=1,100
         do i=1,3
            xb(i) = x(i)
            x(i) = a(i,4)/a(i,i)
            do j=1,3
               if( i .ne. j ) then
                   x(i) = x(i) - a(i,j)*x(j)/a(i,i)
               endif
            enddo
         enddo
         ic = 0
         do i=1,3
            if( abs(x(i)-xb(i)) .lt. 1.0e-6 ) ic=ic+1
         enddo
         write(6,*) n,x
         if( ic .eq. 3 ) go to 200
      enddo
  200 continue
      write(6,*) 'Answer'
      write(6,*) x
      stop
      end
```

（入力データ）

```
4.0  -1.0  1.0  4.0
1.0   6.0  2.0  9.0
-1.0  -2.0  5.0  2.0
```

（実行結果）

```
1  1.  1.33333337  1.13333333
2  1.05000007  0.947222292  0.988888919
3  0.989583313  1.00543976  1.00009251
4  1.00133681  0.999746382  1.00016594
5  0.999895096  0.999962151  0.99996388
6  0.999999642  1.00001204  1.00000477
7  1.00000179  0.999998093  0.999999642
8  0.999999642  1.00000012  0.99999994
9  1.  1.  1.
 Answer
1.  1.  1.
```

課題3

$$A=\begin{bmatrix}2 & 3 & 4\\ 3 & 5 & 2\\ 4 & 3 & 30\end{bmatrix}$$

の逆行列を求めよ．

（解法）$|A|\neq 0$ の行列 A に対して， $AB=I$ （I は単位行列）を満たす行列 B を A の逆行列という． A の逆行列を A^{-1} とすると

$$AA^{-1}=A^{-1}A=I$$

となる．

$$A=\begin{bmatrix}a_{11} & a_{12} & a_{13}\\ a_{21} & a_{22} & a_{23}\\ a_{31} & a_{32} & a_{33}\end{bmatrix}$$

の逆行列を

$$A^{-1}=\begin{bmatrix}x_{11} & x_{12} & x_{13}\\ x_{21} & x_{22} & x_{23}\\ x_{31} & x_{32} & x_{33}\end{bmatrix}$$

とすると

$$\begin{bmatrix}a_{11} & a_{12} & a_{13}\\ a_{21} & a_{22} & a_{23}\\ a_{31} & a_{32} & a_{33}\end{bmatrix}\begin{bmatrix}x_{11} & x_{12} & x_{13}\\ x_{21} & x_{22} & x_{23}\\ x_{31} & x_{32} & x_{33}\end{bmatrix}=\begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\\ 0 & 0 & 1\end{bmatrix}$$

となる．したがって，

$$\begin{bmatrix}a_{11} & a_{12} & a_{13}\\ a_{21} & a_{22} & a_{23}\\ a_{31} & a_{32} & a_{33}\end{bmatrix}\begin{bmatrix}x_{11}\\ x_{21}\\ x_{31}\end{bmatrix}=\begin{bmatrix}1\\ 0\\ 0\end{bmatrix}$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} x_{12} \\ x_{22} \\ x_{32} \end{bmatrix} = \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}$$

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{bmatrix} \begin{bmatrix} x_{13} \\ x_{23} \\ x_{33} \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}$$

の３元連立１次方程式の３つの組から x_{ij} **を求めることになる．これは，課題１の方法と同様のアルゴリズムで解くことができる．すなわち，**

$$\begin{bmatrix} a_{11} & a_{12} & a_{13} & 1 & 0 & 0 \\ a_{21} & a_{22} & a_{23} & 0 & 1 & 0 \\ a_{31} & a_{32} & a_{33} & 0 & 0 & 1 \end{bmatrix}$$

を消去法により

$$\begin{bmatrix} 1 & 0 & 0 & a_{11}^{'} & a_{12}^{'} & a_{13}^{'} \\ 0 & 1 & 0 & a_{21}^{'} & a_{22}^{'} & a_{23}^{'} \\ 0 & 0 & 1 & a_{31}^{'} & a_{32}^{'} & a_{33}^{'} \end{bmatrix}$$

と変形した時，

$$\begin{bmatrix} a_{11}^{'} & a_{12}^{'} & a_{13}^{'} \\ a_{21}^{'} & a_{22}^{'} & a_{23}^{'} \\ a_{31}^{'} & a_{32}^{'} & a_{33}^{'} \end{bmatrix}$$

が逆行列である．

（FORTRAN プログラム例）

```
c
c       ex-3
c
      real*4 a(10,20)
c
      read(5,*) n,((a(i,j),j=1,n),i=1,n)
       do i=1,n
         do j=n+1,2*n
           if( i .eq. (j-n) ) then
               a(i,j) = 1.0
            else
               a(i,j) = 0.0
            endif
         enddo
       enddo
      write(6,*) ' Input Data '
      write(6,800) ((a(i,j),j=1,2*n),i=1,n)
  800 format(6f5.1)
c
       do k=1,n
          p = a(k,k)
          do j=k,2*n
            a(k,j) = a(k,j)/p
          enddo
          do i=1,n
            if( i .ne. k ) then
               aik = a(i,k)
               do j=k,2*n
                    a(i,j) = a(i,j) - aik*a(k,j)
               enddo
            endif
         enddo
       enddo
c
      write(6,*) ' Inverse Matrix '
      write(6,810) ((a(i,j),j=n+1,2*n),i=1,n)
  810 format(3e13.4)
c
       stop
       end
```

（入力データ）

```
3
2 3 4
3 5 2
4 3 30
```

（実行方法）

```
f90 ex3.f
./a.out < ex3.dat
```

（実行結果）

```
 Input Data
  2.0   3.0   4.0   1.0   0.0   0.0
  3.0   5.0   2.0   0.0   1.0   0.0
  4.0   3.0  30.0   0.0   0.0   1.0
 Inverse Matrix
  -0.7200E+02   0.3900E+02   0.7000E+01
   0.4100E+02  -0.2200E+02  -0.4000E+01
   0.5500E+01  -0.3000E+01  -0.5000E+00
```

(課題＋α)

課題１のプログラムでｎ行ｍ列の入力データをファイルから読み込み計算するプログラムに変更せよ.

課題４

$$f(x) = x^3 + 3x^2 + 3x + 1 = 0$$

の根を Newton **法で求めよ．ただし，** $x_0 = 0$ **とし，２つの近似根間の相対差** $e = 10^{-5}$ **を収束判定条件とする．**

（解法）関数 $f(x)$ **が単調連続で変曲点がなく，かつ** $f(x)$ **の導関数が求められるとき，** $f(x) = 0$ **の近似根を** x_0 **，真の根を** $x_0 + h$ **とすると，** $f(x_0 + h) = 0$ **である．テーラー（**Taylor**）展開すると，**

$$f(x_0 + h) = f(x_0) + hf'(x_0) + \frac{h^2}{2} f''(x_0) + \ldots$$

h **を十分小さくとり，** h^2 **以降を無視すると**

$$f(x_0) + hf'(x_0) = 0$$

であるから

$$h = -\frac{f(x_0)}{f'(x_0)}$$

したがって，第１次近似根は

$$x_1 = x_0 + h = x_0 - \frac{f(x_0)}{f'(x_0)}$$

第２次以下の近似根は

$$x_2 = x_1 - \frac{f(x_1)}{f'(x_1)}$$

$$\cdots$$

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

$$\cdots$$

となる．

$$|x_{n+1} - x_n| < e$$

のとき x_n **を解とする**

収束判定の条件として，以下のような種類がある．

（１）２つの近似根間の差

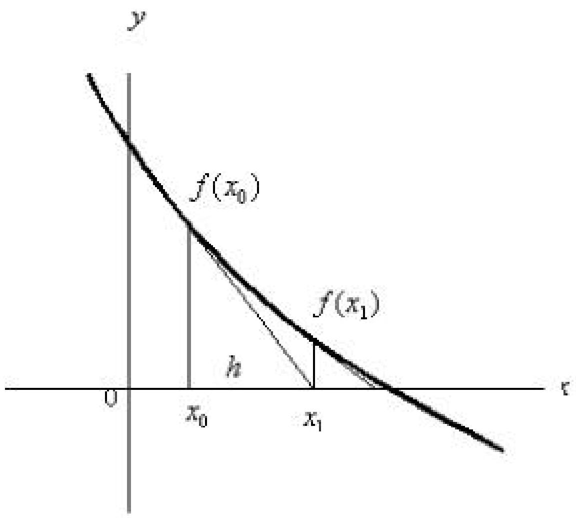
$$\left| x_{n+1} - x_n \right| < e$$

（２）２つの近似根間の相対差

$$\left| \frac{x_{n+1} - x_n}{x_n} \right| < e$$

（３）２つの近似根の関数値間の相対差

$$\left| \frac{f(x_{x+1}) - f(x_n)}{f(x_n)} \right| < e$$



（FORTRAN プログラム例）

```
c
c       ex4
c
      x = 0.0
      do i=1,1000
         fx = x**3 + 3.0*x**2 + 3.0*x + 1.0
         fxd = 3.0*x**2 + 6.0*x + 3.0
         dx = -fx/fxd
         write(6,*) i,x,fx,fxd,dx
         x = x + dx
         if( abs(dx/x) .lt. 1.0e-5 ) then
```

```
        stop
      endif
    enddo
    stop
    end
```

（実行結果）

```
 1  0.  1.  3. -0.333333343
2 -0.333333343  0.296296269  1.33333325 -0.222222209
3 -0.555555582  0.0877914801  0.592592537 -0.148148134
4 -0.703703701  0.0260122959  0.263374478 -0.0987654403
5 -0.802469134  0.0077073467  0.117055327 -0.0658436194
6 -0.868312776  0.0022836572  0.0520245731 -0.0438957401
7 -0.912208498  0.000676639669  0.0231220443 -0.0292638335
8 -0.941472352  0.000200485621  0.0102764573 -0.0195092149
9 -0.960981548  5.94032354E-05  0.0045673186 -0.0130061507
10 -0.973987699  1.76009598E-05  0.00202991953 -0.00867076684
11 -0.982658446  5.21511674E-06  0.000902188476 -0.00578051805
12 -0.988438964  1.54521979E-06  0.000400972669 -0.0038536787
13 -0.992292643  4.57842901E-07  0.000178210073 -0.00256911898
14 -0.994861782  1.35655583E-07  7.92038627E-05 -0.00171273947
15 -0.996574521  4.01942479E-08  3.52017196E-05 -0.00114182627
16 -0.997716367  1.19090959E-08  1.5644935E-05 -0.000761210977
17 -0.998477578  3.52862095E-09  6.95330482E-06 -0.000507473946
18 -0.998985052  1.04551734E-09  3.09035772E-06 -0.000338315964
19 -0.999323368  3.09782922E-10  1.3734923E-06 -0.000225543976
20 -0.999548912  9.1787529E-11  6.1044102E-07 -0.000150362655
21 -0.999699295  2.71909145E-11  2.71271261E-07 -0.000100235149
22 -0.99979955  8.05417191E-12  1.20541117E-07 -6.68167995E-05
23 -0.999866366  2.38642136E-12  5.3573828E-08 -4.45445366E-05
 24 -0.999910891  7.07560963E-13  2.38212117E-08 -2.97029801E-05
25 -0.999940574  2.09858173E-13  1.05942881E-08 -1.98086145E-05
26 -0.999960363  6.22737539E-14  4.71329642E-09 -1.32123569E-05
27 -0.999973595  1.84099697E-14  2.09164952E-09 -8.80165135E-06
```

● **テイラー展開**

ある関数 $f(x)$ **をべき級数で表すことができるとする．**

$$f(x)=\sum_{n=0}^{\infty} {}_{n} x_{n} = c_0 + c_1 x + c_2 x^2 + c_3 x^3 + \ldots + c_n x^n + \ldots \qquad \text{(T-1)}$$

これは $x=0$ **のまわりで展開している．** $f(x)$ **は無限回微分可能とする．式（T-1）において**

$x=0$ **とすると**

$$c_0 = f(0)$$

式（T-1）を 1 回微分して

$$f'(x) = c_1 + 2c_2 x + 3c_3 x^2 + \ldots + nc_n x^{n-1} + \ldots$$

ここで， $x=0$ **とすると**

$$c_1 = f'(0)$$

同様に

$$f''(x) = 2c_2 + 3\cdot 2c_3 x + \ldots + n(n-1)c_n x^{n-2} + \ldots$$

$$f'''(x) = 3\cdot 2c_3 + \ldots + n(n-1)(n-2)c_n x^{n-3} + \ldots$$

であるから

$$c_2 = \frac{1}{2} f''(0)，\quad c_3 = \frac{1}{3\cdot 2} f'''(0)，\ldots，\quad c_n = \frac{1}{n!} f^{(n)}(0)$$

したがって，

$$f(x) = f(0) + \frac{f'(0)}{1!}x + \frac{f''(0)}{2!}x^2 + \frac{f'''(0)}{3!}x^3 + \ldots + \frac{f^{(n)}(0)}{n!}x^n + \ldots$$

となる．これをマクローリン級数という．べき級数の収束半径を r **として** $|x| < r$ **で成立する．**

$f(x)$ **を** $x=a$ **のまわりで展開したとき**

となる．これをテイラー級数（展開）という．

課題５　区間$[-3,3]$における

$$f(x)=x^3+3x^2+3x+1=0$$

の根を二分法で求めよ．

（解法）関数 $f(x)$ **は区間**$[a,b]$**で連続で，根が１つ存在するとき，** $f(a)$**，** $f(b)$ **は異符合になる．** $f(a)f(b)<0$**である．２点**$(a,f(a))$**，**$(b,f(b))$**を直線で結び，この直線と**x**軸との交点を**c**とすると**

$$c=\frac{af(b)-bf(a)}{f(b)-f(a)} \tag{5-1}$$

となる．

$$f(a)f(c)<0 \quad \text{ならば} \quad b=c,\ f(b)=f(c)$$

$$f(b)f(c)<0 \quad \text{ならば} \quad a=c,\ f(a)=f(c)$$

の変換を繰り返して

$$(c-a)(b-c)<e$$

または $$|f(c)|<e$$

となったとき，c**を根とする．これを** Regula falsi **法という．**
(5-1)**式の代わりに**

$$c=\frac{a+b}{2}$$

とした方法を二分法という．収束条件として

$$|b-a|<e$$

または $$|f(c)|<e$$

を用いることが多い．この場合，反復計算回数n**は**

$$\frac{|b-a|}{2^n}<e$$

から計算することができる．

（FORTRAN プログラム例）

```
c
c       ex5
c
      f(x) = x**3 + 3.0*x**2 + 3.0*x + 1.0
c
       a  =  -3.0
       b  =   3.0
c
       fa = f(a)
       do i=1,1000
          c = (a + b)/2.0
           fc = f(c)
          write(6,*) i,c,fc
           if( abs(b-a) .lt. 1.0e-5 ) then
                 stop
           endif
         if( fa*fc .lt. 0.0 ) then
                 b  =  c
           else
                 a  =  c
                fa  =  fc
           endif
       enddo
       stop
       end
```

（実行結果）

```
1   0.  1.
 2 -1.5 -0.125
 3 -0.75  0.015625
 4 -1.125 -0.001953125
 5 -0.9375  0.000244140625
 6 -1.03125 -3.05175781E-05
 7 -0.984375  3.81469727E-06
 8 -1.0078125 -4.76837158E-07
 9 -0.99609375  5.96046448E-08
 10 -1.00195312 -7.4505806E-09
 11 -0.999023438  9.31322575E-10
 12 -1.00048828 -1.16415322E-10
 13 -0.999755859  1.45519152E-11
 14 -1.00012207 -1.8189894E-12
 15 -0.999938965  2.27373675E-13
```

```
16 -1.00003052 -2.84217094E-14
17 -0.999984741  3.55271368E-15
18 -1.00000763 -4.4408921E-16
19 -0.999996185  5.55111512E-17
20 -1.00000191 -6.9388939E-18
21 -0.999999046  8.67361738E-19
```

● 追加課題

$$f(x) = x^3 - x = 0$$

の根をすべて求めるプログラムを作成せよ.

課題 6

$$I = \int_0^p \sin\ x dx$$

を台形公式を用いて，x の幅 20°，10°，5°，1° で計算値 I がどのようになるか示せ．

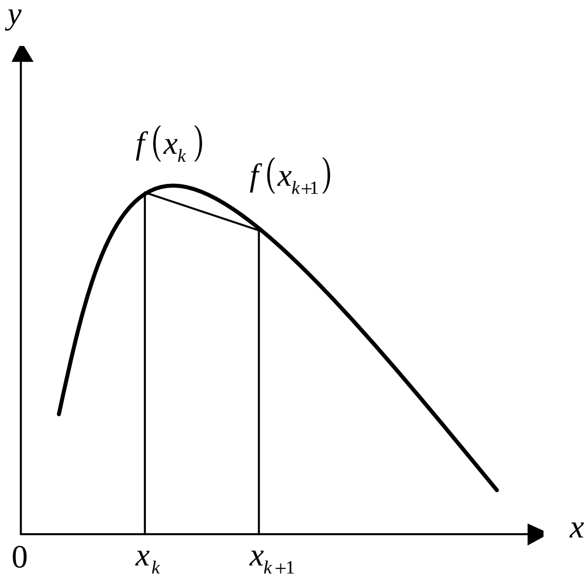
（解法）区間 $(a,b) \equiv (x_0, x_n)$ における定積分を n 個の幅 h の区間に分けて

$$I = \int_{x_0}^{x_n} f(x)dx \approx \sum_{i=0}^{n-1} \frac{\left(f(x_i) + f(x_{i+1})\right)h}{2}$$

で計算する（台形公式）．この式は

$$I \approx \frac{h}{2}\left(f(x_0) + 2f(x_1) + 2f(x_2) + \ldots + 2f(x_{n-1}) + f(x_n)\right)$$

となる．



＊＊＊プログラム例＊＊＊

```
c
c   ex6
c
      data pai / 3.1415927 /
c
      call integd(0.0,pai,pai*20.0/180.0,ans20)
      call integd(0.0,pai,pai*10.0/180.0,ans10)
      call integd(0.0,pai,pai*5.0/180.0,ans5)
      call integd(0.0,pai,pai*1.0/180.0,ans1)
      write(6,*) 'ans20 =',ans20
      write(6,*) 'ans10 =',ans10
      write(6,*) 'ans5  =',ans5
      write(6,*) 'ans1  =',ans1
      stop
      end
c*****integd
      subroutine integd(x0,xn,h,ans)
c
      f(t) = sin(t)
c      f(t) = cos(t)
c      f(t) = 3.0/2.0*sin(t)*cos(t)**2.0
c
      ans = f(x0)
      do x=x0+h,xn-h,h
         ans = ans + 2.0*f(x)
      enddo
      ans = ans + f(xn)
      ans = ans*h/2.0
      return
      end
```

＊＊＊実行結果＊＊＊

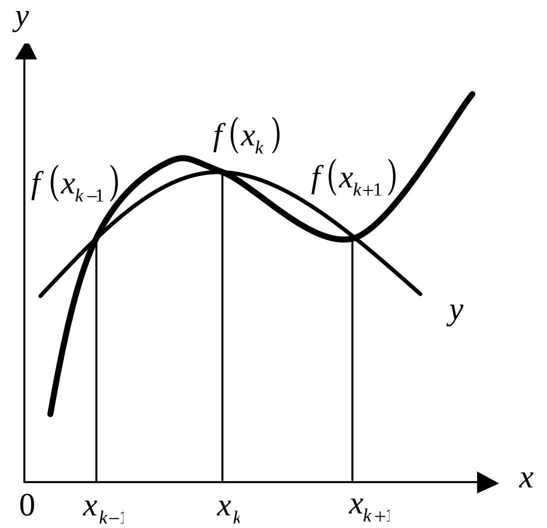
```
ans20 =  1.86026323
ans10 =  1.96461344
ans5  =  1.99112535
ans1  =  1.99995089
```

課題７

$$I=\int_0^1 xe^x dx$$

をシンプソン（ Simpson **）の公式を用いて求めよ．分割数を** $n=4$ **，** $n=10$ **，** $n=100$ **とし，積分結果を比較せよ．また，分割数を同じにしたとき，台形公式とシンプソン公式を用いた場合の計算結果の違いを比較せよ．**

（解法）区間 (x_0, x_n) **における定積分を幅** h **の** $2n$ **個の区間に分ける．**



３点 $\left(f(x_{k-1}), f(x_k), f(x_{k+1})\right)$ **を通る２次曲線を** $y=a(x-x_k)^2+b(x-x_k)+c$ **としたとき，**

(x_{k-1}, x_{k+1}) **間の面積は**

$$I_k=\int_{x_{k-1}}^{x_{k+1}} y dx=\left[\frac{1}{3}a(x-x_k)^2+\frac{1}{2}b(x-x_k)+c(x-x_k)\right]_{x_{k-1}}^{x_{k+1}}$$

$$=\frac{2}{3}ah^3+2ch$$

となる．ここで， $x_{k-1}=x_k-h$ **，** $x_{k+1}=x_k+h$

とした．この関係を用いると

$$f(x_{k-1})=ah^2-bh+c\text{，}\quad f(x_k)=c\text{，}\quad f(x_{k+1})=ah^2+bh+c$$

であるから，

$$I_k=\frac{h}{3}\left[f(x_k)+4f(x_k)+f(x_{k+1})\right]$$

である．区間(x_0, x_n)における定積分Iは

$$I = \frac{h}{3} f(x_0) + \frac{2h}{3}[f(x_2) + f(x_4) + \ldots + f(x_{n-2})] + \frac{4h}{3}[f(x_1) + f(x_3) + \ldots + f(x_{n-1})] + \frac{h}{3} f(x_n)$$

となる．

＊＊＊プログラム例＊＊＊

```
c
c    ex7
c
      call integs(0.0,1.0,4,ans1)
      call integs(0.0,1.0,10,ans2)
      call integs(0.0,1.0,100,ans3)
      write(6,*) 'ans1 =',ans1
      write(6,*) 'ans2 =',ans2
      write(6,*) 'ans3 =',ans3
      stop
      end
c*****integs
      subroutine integs(x0,xn,n,ans)
c
      f(t) = t*exp(t)
c
      h = (xn-x0)/float(n)
      ans = f(x0)
      do i=1,n-1,2
         x = x0 + h*float(i)
         ans = ans + 4.0*f(x)
      enddo
      do i=2,n-2,2
         x = x0 + h*float(i)
         ans = ans + 2.0*f(x)
      enddo
      ans = ans + f(xn)
      ans = ans*h/3.0
      return
      end
```

＊＊＊実行結果＊＊＊

```
ans1 =  1.00016904
ans2 =  1.00000441
ans3 =  1.00000036
```

課題８

関数　$f(x) = e^{-(x-1)^2}$，$-p \le x \le p$

をフーリエ級数で表せ．フーリエ級数の次数が $M = 1,2,3$ のときのフーリエ係数を求めよ．結果を図示せよ．

（解法）$-p \le x \le p$ で定義されている関数 $f(x)$ が

$$f(x) = \frac{1}{2}a_0 + \sum_{n=1}^{\infty}\left(a_n \cos nx + b_n \sin nx\right)$$

の形に展開できたとき，右辺をフーリエ級数といい，a_0，$a_1, \ldots$ b_1，$b_2, \ldots$ をフーリエ係数という．計算の便宜上 $\frac{1}{2}a_0$ とした．フーリエ係数は

$$a_n = \frac{1}{p}\int_{-p}^{p} f(x)\cos nx dx \qquad (n = 0,1,2,\ldots)$$

$$b_n = \frac{1}{p}\int_{-p}^{p} f(x)\sin nx dx \qquad (n = 1,2,3,\ldots)$$

となる．

―――――――――――――― 参考 ―――――――――――――――――――――――――

$n \ne m$ で $n + m > 0$ のとき

$$\int_{-p}^{p}\cos nx \cos mx dx = \frac{1}{2}\int_{-p}^{p}\{\cos(n+m)x + \cos(n-m)x\}dx$$

$$= \frac{1}{2}\left[\frac{1}{n+m}\sin(n+m)x + \frac{1}{n-m}\sin(n-m)x\right]_{-p}^{p} = 0$$

$$\int_{-p}^{p}\sin nx \sin mx dx = -\frac{1}{2}\int_{-p}^{p}\{\cos(n+m)x - \cos(n-m)x\}dx$$

$$= -\frac{1}{2}\left[\frac{1}{n+m}\sin(n+m)x - \frac{1}{n-m}\sin(n-m)x\right]_{-p}^{p} = 0$$

$n = m \geq 1$のとき

$$\int_{-p}^{p} \cos nx \cos nx dx = \frac{1}{2}\int_{-p}^{p} (\cos 2nx + 1) dx = \frac{1}{2}\left[\frac{1}{2n}\sin 2nx + x \right]_{-p}^{p} = p$$

$$\int_{-p}^{p} \sin nx \sin nx dx = -\frac{1}{2}\int_{-p}^{p} (\cos 2nx - 1) dx = -\frac{1}{2}\left[\frac{1}{2n}\sin 2nx - x \right]_{-p}^{p} = p$$

＊＊＊プログラム例＊＊＊

```
c
c   ex9
c
      real*4  an(20),bn(20)
      real*4  fxx(4,100)
      data pai /3.141592 /
c
      dx = 2.0*pai/100.0
      do i=1,100
         x = dx*float(i-1) - pai
         fxx(1,i) = exp(-(x-1.0)**2.0)
      enddo
c
      call fourier(-pai,pai,100,1,an,bn)
      write(6,*) ' M = 1'
         write(6,*) 'an(',0,') = ',an(1)
      do i=2,2
         write(6,*) 'an(',i-1,') = ',an(i),'   bn(',i-1,') = ',bn(i)
      enddo
      do i=1,100
         x = dx*float(i-1) - pai
         fxx(2,i) = an(1)/2.0
         do n=2,2
            fxx(2,i) = fxx(2,i) + an(n)*cos(float(n-1)*x)
     -                          + bn(n)*sin(float(n-1)*x)
         enddo
      enddo
c
      call fourier(-pai,pai,100,2,an,bn)
      write(6,*) ' M = 2'
         write(6,*) 'an(',0,') = ',an(1)
      do i=2,3
         write(6,*) 'an(',i-1,') = ',an(i),'   bn(',i-1,') = ',bn(i)
```

```
      enddo
      do i=1,100
         x = dx*float(i-1) - pai
         fxx(3,i) = an(1)/2.0
         do n=2,3
            fxx(3,i) = fxx(3,i) + an(n)*cos(float(n-1)*x)
     -                          + bn(n)*sin(float(n-1)*x)
         enddo
      enddo
c
      call fourier(-pai,pai,100,3,an,bn)
      write(6,*) ' M = 3'
         write(6,*) 'an(',0,') = ',an(1)
      do i=2,4
         write(6,*) 'an(',i-1,') = ',an(i),'   bn(',i-1,') = ',bn(i)
      enddo
      do i=1,100
         x = dx*float(i-1) - pai
         fxx(4,i) = an(1)/2.0
         do n=2,4
            fxx(4,i) = fxx(4,i) + an(n)*cos(float(n-1)*x)
     -                          + bn(n)*sin(float(n-1)*x)
         enddo
      enddo
c
      do i=1,100
         x = dx*float(i-1) - pai
         write(7,*) x,fxx(1,i),fxx(2,i),fxx(3,i),fxx(4,i)
      enddo
c
      stop
      end
c*****fourier
      subroutine fourier(x0,xn,n,m,an,bn)
      real*4  an(20),bn(20)
c
      h = (xn-x0)/float(n)
c
      do j=1,m+1
         an(j) = fcos(j-1,x0)
         do i=1,n-1,2
            x = x0 + h*float(i)
            an(j) = an(j) + 4.0*fcos(j-1,x)
         enddo
         do i=2,n-2,2
            x = x0 + h*float(i)
```

```
            an(j) = an(j) + 2.0*fcos(j-1,x)
         enddo
         an(j) = an(j) + fcos(j-1,xn)
         an(j) = an(j)*h/3.0*2.0/(xn-x0)
      enddo
      do j=2,m+1
         bn(j) = fsin(j-1,x0)
         do i=1,n-1,2
            x = x0 + h*float(i)
            bn(j) = bn(j) + 4.0*fsin(j-1,x)
         enddo
         do i=2,n-2,2
            x = x0 + h*float(i)
            bn(j) = bn(j) + 2.0*fsin(j-1,x)
         enddo
         bn(j) = bn(j) + fsin(j-1,xn)
         bn(j) = bn(j)*h/3.0*2.0/(xn-x0)
      enddo
c
      return
      end
c*****fcos
      function fcos(m,t)
      fcos = exp(-(t-1.0)**2.0)*cos(float(m)*t)
      return
      end
c*****fsin
      function fsin(m,t)
      fsin = exp(-(t-1.0)**2.0)*sin(float(m)*t)
      return
      end
```

＊＊＊gnuplot　例＊＊＊

```
set xlabel 'X'
set ylabel 'Y'
plot 'fort.7' using 1:2 title 'exp(-x*x)' with lines,\
     'fort.7' using 1:3 title 'M=1' with lines,\
     'fort.7' using 1:4 title 'M=2' with lines,\
     'fort.7' using 1:5 title 'M=3' with lines
pause -1
```

課題９

$$\frac{dy}{dx} = -xy$$

の微分方程式をルンゲ・クッタ（Runge-Kutta**）法で解け．ただし，計算区間は**$(0.0, 15.0)$**とし，**$y(0) = 15.0$**とする．ステップ幅**$h = 0.3$，$h = 0.25$，$h = 0.2$**について解け．**

（解法）

$$\begin{cases} \dfrac{dy}{dx} = f(x, y) \\ y(x_0) = y_0 \end{cases} \tag{9-1}$$

を数値的に解くとき，$x = x_n$**において近似解**y_n**が求められていたとする．最も簡単な２段公式ではステップ幅**h**だけ進んだ**$x_{n+1} = x_n + h$**における値**y_{n+1}**を**

$$y_{n+1} = y_n + (g_1 k_1 + g_2 k_2)$$
$$\begin{cases} k_1 = hf(x_n, y_n) \\ k_2 = hf(x_n + ah, y_n + bk_1) \end{cases} \tag{9-2}$$

を用いて計算する．式（9-1）の真の解$y(x)$**のテイラー展開は**

$$\begin{aligned} y(x_n + h) &= y(x_n) + hf(x_n, y(x_n)) + \frac{1}{2}h^2 f'(x_n, y(x_n)) + O(h^3) \\ &= y(x_n) + hf(x_n, y(x_n)) + \frac{1}{2}h^2 \{f_x(x_n, y(x_n)) + f_y(x_n, y(x_n)) f(x_n, y(x_n))\} \\ &\quad + O(h^3) \end{aligned}$$

となる．$y_n = y(x_n)$**として式（9-2）に適用すると**

$$\begin{aligned} y_{n+1} &= y(x_n) + h(g_1 + g_2) f(x_n, y_n) \\ &\quad + h^2 g_2 \{af_x(x_n, y_n) + bf_y(x_n, y_n) f(x_n, y_n)\} + O(h^3) \end{aligned}$$

となる．h^2**の項まで一致するように**

$$\begin{cases} g_1 + g_2 = 1 \\ g_2 a = g_2 b = \dfrac{1}{2} \end{cases}$$

とおく．

$g_2 = 1/2$**とした公式を改良オイラー公式またはホイン公式，**$g_2 = 1$**のときを修正オイラー公式という．１ステップ幅の中においてもとの方程式の右辺の値をいくつか計算し，その重みつき平均で新しいステップにおける近似解を求める公式を総称してルンゲ・クッタ公式，その解法をルンゲ・クッタ法という．**

●全微分と偏微分

関数 $z = f(x, y)$ において，点 $P(x, y)$ を固定し x と y の増分 Δx と Δy が十分小さいとするとき， z の増分 Δz は次のように表される．

$$\begin{aligned}\Delta z &= f(x+\Delta x, y+\Delta y) - f(x, y) \\ &= \left[f(x+\Delta x, y+\Delta y) - f(x+\Delta x, y)\right] + \left[f(x+\Delta x, y) - f(x, y)\right]\end{aligned}$$

平均値の定理

$$f(x+\Delta x, y) - f(x, y) = \frac{\partial f(x, y)}{\partial x}\Delta x$$

を用いると

$$\Delta z = \frac{\partial f(x, y)}{\partial x}\Delta x + \frac{\partial f(x, y)}{\partial y}\Delta y$$

となる． Δx と Δy を無限小とするとき，微分 dx と dy を用いて書くことができる．

$$dz = \frac{\partial f(x, y)}{\partial x}dx + \frac{\partial f(x, y)}{\partial y}dy$$

（プログラム例）

```
c
c       ex9
c
        y = 15.0
        h = 0.3
        do x=0.0,15.0,h
           write(10,*) x,y
           call runge(h,x,y)
        enddo
        y = 15.0
        h = 0.25
        do x=0.0,15.0,h
           write(11,*) x,y
           call runge(h,x,y)
        enddo
        y = 15.0
        h = 0.2
        do x=0.0,15.0,h
```

```
        write(12,*) x,y
        call runge(h,x,y)
      enddo
      stop
      end
c*****runge
      subroutine runge(h,x,y)
      real*4   k1, k2
      k1 = h*(-x*y)
      k2 = h*(-(x+h)*(y+k1))
      y = y + 0.5*(k1+k2)
      return
      end
```

（gnuplot プログラム）

```
set xrange [0:15]
set yrange [0:15]
set xlabel 'X'
set ylabel 'Y'
plot 'fort.10' title 'h=0.3' with lines,\
     'fort.11' title 'h=0.25' with lines,\
     'fort.12' title 'h=0.2' with lines
pause -1
```

―――――おまけプログラム――――――――――――――――――――――――――――――――

```
c
c     ex9-1
c
      write(6,*) 'h ='
      read(5,*) h
      y = 1.0
      do x=0.0,15.0,h
```

```
        write(10,*) x,y
        y = y - h*sin(x)
      enddo
      y = 1.0
      do x=0.0,15.0,h
        write(11,*) x,y
        call runge(h,x,y)
      enddo
      do x=0.0,15.0,0.1
        write(12,*) x,cos(x)
      enddo
      stop
      end
c*****runge
      subroutine runge(h,x,y)
      real*4   k1, k2
      k1 = -h*sin(x)
      k2 = -h*sin(x+h)
      y = y + 0.5*(k1+k2)
      return
      end
```

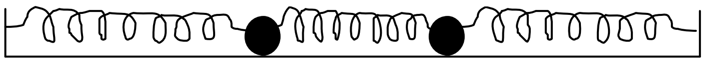
（gnuplot プログラム）

```
set xrange [0:6.28]
set yrange [-1:1]
set xlabel 'X'
set ylabel 'Y'
plot 'fort.10' title 'Euler' with lines,\
     'fort.11' title 'Runge' with lines,\
     'fort.12' title 'Exact' with lines
pause -1
```

―――

課題１０　連成振動をルンゲ・クッタ（Runge-Kutta）法で解け．また，ばねの定数 c_1，c_2，

k **や質点の質量** m_1，m_2 **を変えて，計算してみよ．**



（解法）連立常微分方程式

$$\begin{cases} \dfrac{dy_1}{dx} = f_1(x, y_1, y_2) \\ \dfrac{dy_2}{dx} = f_2(x, y_1, y_2) \end{cases}$$

をルンゲ・クッタ（Runge-Kutta）**法の２段公式で解くときは，課題９と同様に，**

$$\begin{aligned} k_{11} &= hf_1(x_n, y_{1n}, y_{2n}) \\ k_{21} &= hf_2(x_n, y_{1n}, y_{2n}) \\ k_{12} &= hf_1(x_n + h, y_{1n} + k_{11}, y_{2n} + k_{21}) \\ k_{22} &= hf_2(x_n + h, y_{1n} + k_{11}, y_{2n} + k_{21}) \\ y_{1,n+1} &= y_{1,n} + \frac{1}{2}h(k_{11} + k_{12}) \\ y_{2,n+1} &= y_{2,n} + \frac{1}{2}h(k_{21} + k_{22}) \end{aligned}$$

とすればよい．

連成振動の運動方程式は

$$\begin{cases} m_1 \dfrac{d^2 x_1}{dt^2} = -(c_1 + k)x_1 + kx_2 \\ m_2 \dfrac{d^2 x_2}{dt^2} = kx_1 - (c_2 + k)x_2 \end{cases}$$

である．速度を用いて

$$\begin{cases} m_1 \dfrac{dv_1}{dt} = -(c_1 + k)x_1 + kx_2 \\ \dfrac{dx_1}{dt} = v_1 \\ m_2 \dfrac{dv_2}{dt} = kx_1 - (c_2 + k)x_2 \\ \dfrac{dx_2}{dt} = v_2 \end{cases}$$

の連立微分方程式を解くことになる．

———最もよく使われるルンゲ・クッタ（Runge-Kutta）法———

$$\begin{aligned} k_1 &= hf(x_0, y_0) \\ k_2 &= hf\left(x_0 + \frac{h}{2}, y_0 + \frac{k_1}{2}\right) \\ k_3 &= hf\left(x_0 + \frac{h}{2}, y_0 + \frac{k_2}{2}\right) \\ k_4 &= hf\left(x_0 + h, y_0 + k_3\right) \\ y_1 &= y_0 + \frac{1}{6}\left(k_1 + 2k_2 + 2k_3 + k_4\right) \end{aligned}$$

————————————————————————————————

＊＊＊プログラム例＊＊＊

```
c
c     ex10
c
      real*8   dt,pm,pn,pc,pk,x1,x2,v1,v2
c
      pm = 1.0
      pn = 0.8
      pc = 1.0
      pk = 0.1
      x1 = 0.1
      x2 = 0.0
      v1 = 0.0
      v2 = 0.0
      dt = 1.0d-3
      tm = 0.0
c
      do i=1,100000
         if( mod(i,100) .eq. 1 ) then
            write(10,*) tm,x1,x2,v1,v2
         endif
         call runge(dt,pm,pn,pc,pk,x1,x2,v1,v2)
         tm = tm + dt
      enddo
      stop
      end
c*****runge
      subroutine runge(dt,pm,pn,pc,pk,x1,x2,v1,v2)
      real*8   k11,k21,k31,k41
      real*8   k12,k22,k32,k42
      real*8   k13,k23,k33,k43
      real*8   k14,k24,k34,k44
      real*8   dt,pm,pn,pc,pk,x1,x2,v1,v2
      k11 = dt/pn*(-(pc+pk)*x1+pk*x2)
      k21 = dt*v1
      k31 = dt/pm*(pk*x1-(pc+pk)*x2)
      k41 = dt*v2
      k12 = dt/pn*(-(pc+pk)*((x1+k11/2.0)+pk*(x2+k31/2.0)))
      k22 = dt*(v1+k21/2.0)
      k32 = dt/pm*(pk*(x1+k11/2.0)-(pc+pk)*(x2+k31/2.0))
      k42 = dt*(v2+k41/2.0)
      k13 = dt/pn*(-(pc+pk)*((x1+k12/2.0)+pk*(x2+k32/2.0)))
      k23 = dt*(v1+k22/2.0)
      k33 = dt/pm*(pk*(x1+k12/2.0)-(pc+pk)*(x2+k32/2.0))
      k43 = dt*(v2+k42/2.0)
      k14 = dt/pn*(-(pc+pk)*((x1+k13)+pk*(x2+k33)))
```

```
      k24 = dt*(v1+k23)
      k34 = dt/pm*(pk*(x1+k13)-(pc+pk)*(x2+k33))
      k44 = dt*(v2+k43)
      x1 = x1 + (k21+2.0*k22+2.0*k23+k24)/6.0
      x2 = x2 + (k41+2.0*k42+2.0*k43+k44)/6.0
      v1 = v1 + (k11+2.0*k12+2.0*k13+k14)/6.0
      v2 = v2 + (k31+2.0*k32+2.0*k33+k34)/6.0
      return
      end
```

＊＊＊gnuplot のプログラム例＊＊＊

```
set xlabel 't'
set ylabel 'X'
plot 'fort.10' using 1:2 title 'x1' with lines,\
     'fort.10' using 1:3 title 'x2' with lines
pause -1
set xlabel 't'
set ylabel 'V'
plot 'fort.10' using 1:4 title 'v1' with lines,\
     'fort.10' using 1:5 title 'v2' with lines
pause -1
```

課題１１　r m 離れたところに，大きさを無視できる電荷$\pm q$がある．その周囲の電位を求めよ．　また，電気力線を示せ．

（解法）ガウスの法則

$$\nabla\mathbf{E} = \frac{r}{e_0}$$

を用いて電位f

$$\mathbf{E} = -\nabla f$$

を求める．ここで，$\mathbf{E}$ は電場，$r = q_+ n_+ - q_- n_-$ は電荷密度，$e_0 = 8.854\times 10^{-12}$ F/m は誘電率である．電荷$\pm q$がr m 離れているとき，

$$-\Delta f = \frac{q(\mathbf{r}_+) - q(\mathbf{r}_-)}{e_0}$$

である．ここで，$|\mathbf{r}_+ - \mathbf{r}_-| = r$ である．x軸上に電荷があり，xy平面内での電位

$$\frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} = -\frac{q(r_+) - q(r_-)}{e_0}$$

を求めることにする．ポアソン方程式の一般形は，

$$\Delta f(x,y,z) = g(x,y,z)$$

であり，２次元の平面上では

$$\frac{\partial^2 f(x,y)}{\partial x^2} + \frac{\partial^2 f(x,y)}{\partial y^2} = g(x,y)$$

となる．また，

$$\Delta f(x,y,z) = 0$$

をラプラス方程式という．

電荷q_tを持った粒子は電気力線に沿って運動する．そこで，テスト粒子を系の中に入れ粒子の軌跡を追うことにする．運動方程式は

$$m\frac{d\mathbf{v}}{dt} = -\frac{q_t q}{4pe_0(r-r_+)^2}\frac{\mathbf{r}-\mathbf{r}_+}{|\mathbf{r}-\mathbf{r}_+|} + \frac{q_t q}{4pe_0(r-r_-)^2}\frac{\mathbf{r}-\mathbf{r}_+}{|\mathbf{r}-\mathbf{r}_+|}$$

となる．また，電場$\mathbf{E}$を用いると

$$m\frac{d\mathbf{v}}{dt} = -q_t\mathbf{E}$$

となる．ここで，

$$\mathbf{E} = \frac{q}{4pe_0(r-r_+)^2}\frac{\mathbf{r}-\mathbf{r}_+}{|\mathbf{r}-\mathbf{r}_+|} - \frac{q}{4pe_0(r-r_-)^2}\frac{\mathbf{r}-\mathbf{r}_+}{|\mathbf{r}-\mathbf{r}_+|}$$

である．$\mathbf{E} = -\nabla f$ とルンゲ・クッタ（Runge-Kutta）法を用いて解いてみよ．

偏微分方程式を数値的に解く方法として差分法がある．関数 $f(r)$ のテイラー展開は

$$f(r+\Delta r)=f(r)+\frac{f'(r)}{1!}\Delta r+\frac{f''(r)}{2!}(\Delta r)^2+\dots$$

$$f(r-\Delta r)=f(r)-\frac{f'(r)}{1!}\Delta r+\frac{f''(r)}{2!}(\Delta r)^2+\dots$$

である． $f''(r)$ 以下を無視して $f'(r)$ について求めると

$$f'(r)=\frac{f(r+\Delta r)-f(r-\Delta r)}{2\Delta r}$$

となる．これを中央差分という．同様にして

$$f'(r)=\frac{f(r+\Delta r)-f(r)}{\Delta r}$$

を前進差分，

$$f'(r)=\frac{f(r)-f(r-\Delta r)}{\Delta r}$$

を後退差分という． $f''(r)$ を求めるときは， $f'''(r)$ 以下を無視して

$$f''(r)=\frac{f(r+\Delta r)-2f(r)+f(r-\Delta r)}{(\Delta r)^2}$$

となる．ポアソン方程式を差分化すると

$$\frac{f(x+\Delta x,y)-2f(x,y)+f(x-\Delta x,y)}{(\Delta x)^2}+\frac{f(x,y+\Delta y)-2f(x,y)+f(x,y-\Delta y)}{(\Delta y)^2}=g(x,y)$$

となる．

x 軸を n 分割， y 軸を m 分割したときの，空間格子点を k とする．差分方程式は

$$f(x_{k-1},y_k)\frac{1}{(\Delta x)^2}+f(x_{k+1},y_k)\frac{1}{(\Delta x)^2}-f(x_k,y_k)\left(\frac{2}{(\Delta x)^2}+\frac{2}{(\Delta y)^2}\right)+f(x_k,y_{k-1})\frac{1}{(\Delta y)^2}+f(x_k,y_{k+1})\frac{1}{(\Delta y)^2}=g_k(x_k,y_k)$$

となる．境界条件は何らかの形で与えなくてはいけない．

差分方程式を反復法で解く場合（行列を作って解く方法もある），

$$f(x_k,y_k)=f(x_{k-1},y_k)\frac{(\Delta y)^2}{2\left((\Delta x)^2+(\Delta y)^2\right)}+f(x_{k+1},y_k)\frac{(\Delta y)^2}{2\left((\Delta x)^2+(\Delta y)^2\right)}+f(x_k,y_{k-1})\frac{(\Delta x)^2}{2\left((\Delta x)^2+(\Delta y)^2\right)}+f(x_k,y_{k+1})\frac{(\Delta x)^2}{2\left((\Delta x)^2+(\Delta y)^2\right)}-g_k(x_k,y_k)\frac{(\Delta x)^2(\Delta y)^2}{2\left((\Delta x)^2+(\Delta y)^2\right)}$$

を，すべての分割点に対して行う． $f(x_k, y_k)$ が収束するまで繰り返し，収束したときの値を解とする．

＊＊＊Fortran プログラム例＊＊＊

```
c
c     ex11
c
      real*4  fd(32,32)
c
      do i=1,32
         do j=1,32
            fd(i,j) = 0.0
         enddo
      enddo
c
      dx = 1.0
      dy = 1.0
      qc = 1.0
      kx1 = 11
      kx2 = 21
      ky1 = 16
      ky2 = 16
c
      tt = 0.0
      do n=1,1000
         tb = tt
         tt = 0.0
         do i=2,31
         do j=2,31
            fd(i,j) = fd(i-1,j)*dy*dy/2.0/(dx*dx+dy*dy)
     -                +fd(i+1,j)*dy*dy/2.0/(dx*dx+dy*dy)
     -                +fd(i,j-1)*dx*dx/2.0/(dx*dx+dy*dy)
     -                +fd(i,j+1)*dx*dx/2.0/(dx*dx+dy*dy)
            if( i .eq. kx1 .and. j .eq. ky1 ) then
```

```
            fd(i,j) = fd(i,j) + qc*dx*dx*dy*dy/2.0/(dx*dx+dy+dy)
         endif
         if( i .eq. kx2 .and. j .eq. ky2 ) then
            fd(i,j) = fd(i,j) - qc*dx*dx*dy*dy/2.0/(dx*dx+dy+dy)
         endif
         tt = tt + abs(fd(i,j))
      enddo
      enddo
      dtt = abs(tt-tb)/(abs(tt)+abs(tb))
      write(6,*) n,tt,dtt
      if( dtt .lt. 1.0e-6 ) then
          go to 900
      endif
   enddo
900 continue
   do j=1,32
      do i=1,32
         write(10,*) fd(i,j)
      enddo
      write(10,*)
   enddo
   stop
   end
```

＊＊＊Gnuplot プログラム例＊＊＊

```
set xlabel 'X'
set ylabel 'Y'
set zlabel 'V'
set contour
splot 'fort.10' with lines
pause -1
```

課題１２　温度 100℃で厚さａｍ の板を 0℃の水中に入れたとき，板の内部の温度変化をを求めよ．

（解法）板は十分大きいとして，１次元の熱伝導方程式

$$\frac{\partial T}{\partial t} = k^2 \frac{\partial^2 T}{\partial x^2}$$

を解くことにする．差分法を用いると，熱伝導方程式は

$$\frac{T(x,t+\Delta t) - T(x,t)}{\Delta t} = k^2 \frac{T(x+\Delta x,t) - 2T(x,t) + T(x-\Delta x,t)}{(\Delta x)^2}$$

となる．空間格子点を j**，時間格子点を** n **とすると，**

$$T_j^{n+1} - T_j^n = l\left(T_{j+1}^n - 2T_j^n + T_{j-1}^n\right)$$

となる．ここで，

$$l = \frac{k^2 \Delta t}{(\Delta x)^2}$$

である．したがって，

$$T_j^{n+1} = T_j^n + l\left(T_{j+1}^n - 2T_j^n + T_{j-1}^n\right)$$

の計算を繰り返し解く．この方法を陽（Explicit）スキームという．収束条件は

$$0 \le l \le \frac{1}{2}$$

である．

$$T_j^{n+1} - l\left(T_{j+1}^{n+1} - 2T_j^{n+1} + T_{j-1}^{n+1}\right) = T_j^n$$

として解く方法を陰（Implicit）スキームといい， l **の値によらず安定である．行列を解く，あるいは反復法で解くことになる．**

$$T_j^{n+1} - T_j^n = l\left(T_{j+1}^{n+1} - 2T_j^{n+1} + T_{j-1}^{n+1}\right) + l\left(T_{j+1}^n - 2T_j^n + T_{j-1}^n\right)$$

の形で解く方法をクランク・ニコルソン（Crank-Nicolson）スキームといい無条件に安定である．ここで，

$$l = \frac{k^2 \Delta t}{2(\Delta x)^2}$$

である．

＊＊＊Fortran プログラム例＊＊＊

```
c
c     ex12
c
      real*4  temp(64), temb(64)
c
      do i=2,63
         temp(i) = 100.0
      enddo
         temp(1) = 0.0
         temp(64) = 0.0
c
      dd = 0.5
c
      do n=1,2000
         do i=2,63
            temb(i) = temp(i) +
     -                dd*(temp(i+1)-2.0*temp(i)+temp(i-1))
         enddo
         do i=2,63
            temp(i) = temb(i)
         enddo
         if( mod(n,100) .eq. 0 ) then
            write(10,*) temp(1)
            do i=2,63
               write(10,*) temp(i)
            enddo
            write(10,*) temp(64)
            write(10,*)
            write(10,*)
         endif
      enddo
  900 continue
      stop
      end
```

＊＊＊Gnuplot プログラム例＊＊＊

```
set xlabel "x"
set ylabel "T"
set xrange [0:63]
plot 'fort.10' with lines
pause -1
```

課題１３　１次元波動の伝播を数値的に解け．初期条件を変えて伝播の様子を調べてみよ．

（解法）１次元の波動方程式

$$\frac{\partial^2 f}{\partial t^2} = c^2 \frac{\partial^2 f}{\partial x^2}$$

を解く．陽（Explicit）**スキームの差分法を用いて，空間格子点を** j **，時間格子点を** n **とすると，**

$$f_j^{n+1} - 2f_j^n + f_j^{n-1} = l\left(f_{j+1}^n - 2f_j^n + f_{j-1}^n\right)$$

となる．ここで，

$$l = \frac{c^2 (\Delta t)^2}{(\Delta x)^2}$$

である．したがって，

$$f_j^{n+1} = 2f_j^n - f_j^{n-1} + l\left(f_{j+1}^n - 2f_j^n + f_{j-1}^n\right)$$

の計算を繰り返し解く．

＊＊＊Fortran プログラム例＊＊＊

```
c
      real*4 u(3,128)
      data dxdt2 / 0.1 /
c
      open(1,file='plot.out')
         do i=1,128
            read(1,*,end=100) j,u(1,i),u(2,i)
         enddo
         go to 200
  100    continue
         do i=1,128
            u(1,i)=0.0
            u(2,i)=0.0
            if( i .ge. 8 .and. i .le. 20 ) then
                u(1,i)=float(i-8)/12.0
                u(2,i)=float(i-8)/12.0
            endif
            if( i .ge. 20 .and. i .le. 32 ) then
                u(1,i)=1.0-float(i-20)/12.0
```

```
               u(2,i)=1.0-float(i-20)/12.0
            endif
         enddo
  200    continue
      close(1)
c
      do i=2,127
         u(3,i) = 2.0*u(2,i) - u(1,i) +
     -             ( u(2,i+1) - 2.0*u(2,i) + u(2,i-1) )*dxdt2
      enddo
      do i=1,128
         u(1,i) = u(2,i)
         u(2,i) = u(3,i)
      enddo
c
      open(1,file='plot.out')
         do i=1,128
            write(1,*) i,u(1,i),u(2,i)
         enddo
      close(1)
c
      stop
      end
```

＊＊＊Gnuplot プログラム例＊＊＊

```
# plot.gnu
!rm plot.out
!./a.out
plot [1:128] [-1:1] "plot.out" using 1:3 with lines
load "plot13a.gnu"
```

＊＊＊Gnuplot プログラム例＊＊＊

```
# plot13a.gnu
!./a.out
replot
reread
```