

# 数値計算法

第一回  
UNIXを使おう

若狭 智嗣

粒子物理学講座

# はじめに

- この講義は**計算機を使った演習形式**で行われます。
- 演習の資料(講義ノート)は  
<http://www.kutl.kyushu-u.ac.jp/~wakasa>  
の  
**数値計算法概論**  
というところをクリックすると、章ごとに入手できます。



ここを  
クリック

# はじめに

- ・ ブラウザ (Internet Explorer) のお気に入りに追加するなどして、すぐにアクセス出来るようにしておいてください。
- ・ 資料はPDF形式で用意されています。上記ページで見たい章をクリックすれば、Acrobat Readerが自動的に起動して、画面で見ることが出来ます。
- ・ 印刷可能枚数は、**半期で250枚**に制限されています。講義ノートを印刷すると、制限に抵触する可能性があるため、以下のことを心がける。
  - 原則、画面で見る
  - 必要なページだけ印刷する

# 目的

- ・ **FORTRAN**、Cなどのプログラミング言語を習得する。
- ・ **よい**プログラミング・スタイルを身につけ、将来の大規模なソフトウェア開発や共同開発の基礎とする
- ・ 数値計算法の基礎を学び、**物理の諸問題に適用**する
- ・ 数値計算や(学生)実験の結果を**視覚化**し、物理の理解を深める
- ・ 論文清書、電子メール、インターネット(WWW)等の**計算機文化**を体験する

# 注意点

- ・ 計算機は**道具**である。
  - 計算機が物理を解くわけではない。
  - 道具も使い方を誤ると、大怪我をする。
- ・ (とはいえ)非常に強力かつ便利
  - 実現不可能な実験をシミュレーションできる。
  - 解析的に解けない式(微分方程式など)を数値的に解ける。
- ・ (でも)計算機を単純に信用しないこと
  - 計算機の出力は多くの場合、一見もつともらしい。
  - 計算機で計算したことと、結果の正しさは無関係。
    - ・ 計算機への入力(プログラム)が間違っている。
    - ・ 数値計算は基本的に誤差を含む  
(計算機は正確には計算しない!!!)
  - 結果を、紙と鉛筆・自分の頭で正しいか判断できる素養が必要

# 要望・連絡先

- ・ **質問などは、演習中に積極的に行うこと**
- ・ 演習中は自由に席を移動したり、話をしたりしてよいので、以下のことを推奨する
  - 友人(歩くマニュアル)に質問する
  - 困っている人を助ける
  - 問題をどう解決すればよいか議論する
- ・ 演習後に質問がある場合は、教員かTAに**電子メール**で質問するか、居室を訪ねること。電子メールの場合は、自身の学籍番号と氏名を記載し、質問の内容を明瞭かつ簡素に記述する。電子メールアドレスと居室は以下の通り。
  - 若狭智嗣                      [wakasa@phys.kyushu-u.ac.jp](mailto:wakasa@phys.kyushu-u.ac.jp)  
(理学部2号館1階2169)
  - 西村信哉                      [nobuya@phys.kyushu-u.ac.jp](mailto:nobuya@phys.kyushu-u.ac.jp)  
(六本松キャンパス)

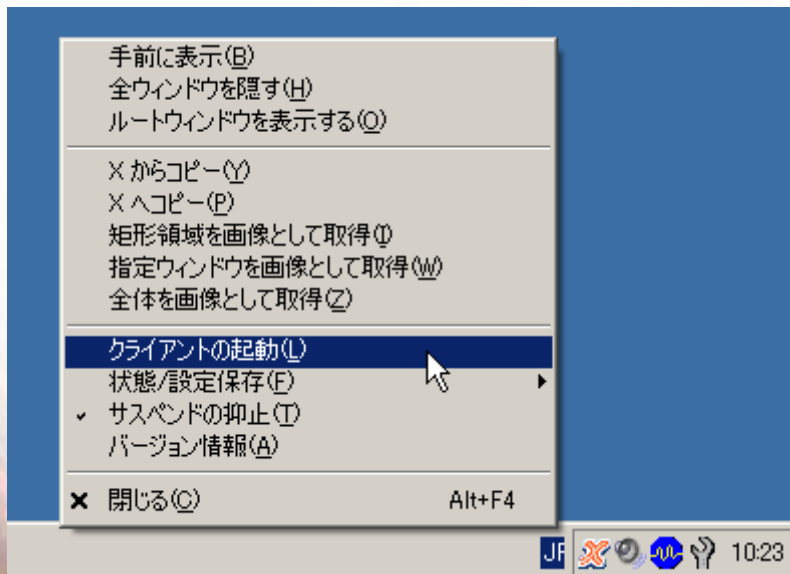
# 参考文献

- ・ 講義資料はある程度自己完結的に作成してあるが、発展した内容を学習したいものは、以下の文献などを参照
- ・ **教育用システム**
  - 「教育用システム利用の手引き」九州大学情報基盤センター
- ・ **FORTRAN**
  - 「FORTRAN90入門」新井親夫 著(森北出版)
- ・ **物理学における計算機の利用**
  - 「計算物理」早野龍五・高橋忠幸 著(共立出版)
- ・ **数値計算法**
  - 「計算物理 I」夏目雄平・小川建吾 著(朝倉書店)
  - 「数値計算の常識」伊理正夫・藤野和建 著(共立出版)



# Xでの接続方法 (ASTEC-Xの使い方)

- Windowsで、  
    **スタート → プログラム → ASTEC-X → ASTEC-X**  
として、ASTEC-Xを起動する
- タスクトレイ内のASTEC-Xのアイコン (**X文字**) 上で**右クリック**して、「**クライアントの起動**」を選択
- クライアントの起動ウインドウで、**ユーザー名とパスワード**を入力して、**OKボタン**を押す



# Xでの接続方法(終了方法)

- ・ Xでの接続が完了すると、**kterm**と呼ばれる端末ウィンドウが開く。このウィンドウから、**UNIXに対してコマンドを入力して作業する事ができる。**
- ・ **ここで、**  
    **% kterm & (リターン・キーを押す)**  
    **とすると、新しいktermが開く。試してみよ。**
- ・ UNIXでの作業が終了したら**ログアウト**(PCとUNIXの接続を解除)する。  
    **具体的には、ktermで**  
        **% logout (リターン・キーを押す)**  
    **とする。すると、ktermのウィンドウは消える。複数のktermを開いている場合は、各々ログアウトする必要がある。**
- ・ ログアウトが完了したら、タスクトレイ内のASTEC-Xのアイコン(**X文字**)上で**右クリック**して、「**閉じる**」を選択して、ASTEC-Xを終了する。

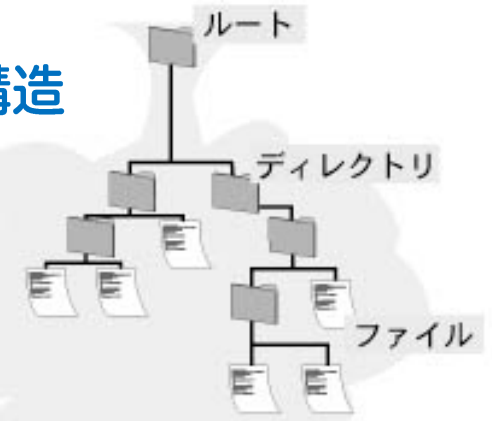
# 第3章 UNIXシステムとプログラミング言語

- ・ 第3章では、
  - ・ UNIXシステムの使い方(主にファイル操作)
  - ・ エディタ(mule)の使い方について説明している。次章(次回)以降のFORTRANによるプログラムの基礎となるので、よく練習すること。
- ・ **全ての項目を練習する必要は無い**(必要な時に、この資料を見ればよい)ので、**よく使いそうなものや、動作がよく理解できないもの**を中心に実際に練習すること。
- ・ ファイルは、演習の例題が置いてあるディレクトリ  
[/home/teacher/z6wt01in/SAMPLE](#)  
に幾つかファイルがあるので、適宜コピーしたり、削除したり、編集したりすること。

# ディレクトリとツリー構造

- ・ **ディレクトリ**
  - ウィンドウズの「フォルダ」に相当
  - ディレクトリ(フォルダ)の中にファイルを納めたり、さらにディレクトリ(フォルダ)を作成出来る
    - ・ **ディレクトリ(フォルダ)に階層構造が出来る**
- ・ **階層構造の最上位のディレクトリを「ルート」という**
  - ウィンドウズの”C:”(ドライブ番号)に相当
  - **木の根(root)にたとえている**
- ・ **階層構造とツリー構造**
  - 木は根から幹が伸び、枝分かれして、葉をつける
  - 根を「ルート」、枝を「ディレクトリ(フォルダ)」、葉を「ファイル」と考えると、階層構造と木の構造は似ている
    - ・ **ツリー構造と呼ぶ**

計算機の構造



木の構造



# ディレクトリとファイル

- ・ ファイルを扱う(コピーや編集)には、以下の指定が必要
  - ファイル名(アルファベット・数字・\_(アンダースコア))
  - ファイルのある場所(ディレクトリ)
- ・ ディレクトリは“指定しなければ”現在の作業ディレクトリ(pwdで表示されるもの)が仮定される。
  - 例えば、ユーザー名がsc106000の場合、

```
% pwd
```

```
/home/06nen/sc106000
```

```
% cp test1 test2
```

は以下と同じ

```
% cp /home/06nen/sc106000/test1 /home/06nen/sc106000/test2
```

# 相対パスと絶対パス

- ディレクトリの指定の仕方は以下の2種類
  - 絶対パス： / (ルートディレクトリ) から完全に記述
  - 相対パス：現在のディレクトリからの相対位置を記述

- 例

```
% pwd  
/home/06nen/sc106000/dir1
```

に

```
/home/06nen/sc106000/dir2/test2
```

をコピーしたい場合

```
% cp /home/06nen/sc106000/dir2/test2 test2 (絶対パス指定)
```

```
% cp ../dir2/test2 test2 (相対パス指定)
```

- ディレクトリの指定の仕方

- .. : ひとつ上のディレクトリ (親ディレクトリ)

- . : 現在のディレクトリ (作業ディレクトリ)

- % cp /home/06nen/sc106000/dir2/test2 .

は

```
% cp /home/06nen/sc106000/dir2/test2 test2
```

と同じ

# 演習 1

- ・ ログインした時にいるディレクトリを「ホームディレクトリ」という。ホームディレクトリは一般に  
/home/06nen/sc106000
- ・ ホームディレクトリに移動  
% cd
- ・ 現在いるディレクトリの表示  
% pwd  
/home/06nen/sc106000
- ・ 新しいディレクトリ（ウィンドウズのフォルダに対応）  
DIR1 を作成  
% mkdir DIR1
- ・ ディレクトリが本当に作成されているか確認  
% ls (LiSt の意味)  
DIR1/ (最後の / がディレクトリを表す)

# 演習2

- ・ ディレクトリを移動 (cd)

```
% cd DIR1
% pwd
/home/06nen/sc106000/DIR1
```

- ・ ファイルをコピーする (cp)

```
% cp /home/teacher/z6wt01in/SAMPLE/circle.f .
% ls
circle.f
```

スペースの後に  
ピリオド

- ・ ファイルの内容の表示 # 1 (cat)

```
% cat circle.f
```

行数が多いと表示が流れるので、  
kterm のスクロールバーを利用する

- ・ ファイルの内容の表示 # 2 (less)

```
% less circle.f
```

・ 矢印キーで上下に移動  
・ "q" キーを押すと終了

# 演習3

- ・ 名前を変える (mv)

```
% mv circle.f maru.f  
% ls  
maru.f
```

- ・ ファイルを消す (rm)

```
% rm maru.f  
% ls  
%
```

(ディレクトリが空なので何も表示されない)

- ・ 親ディレクトリに移動

```
% cd ..  
% pwd  
/home/06nen/sc106000
```

- ・ ディレクトリを消す (rmdir)

```
% rmdir DIR1
```

# 注意事項

- ・ 各人のホームディレクトリで、  
    % ls -la (リターンを押す)  
とした時に表示される.(ピリオド)で始まるファイル、例えば  
    .cshrc  
などは**重要なファイルであるので、消さない(rmしない)こと。**
- ・ 各人のホームディレクトリで  
    % rm . (リターンを押す)  
    % rm .\* (リターンを押す)  
などと**決してしないこと(如何に危険かは、資料に書いてある)。**

# 続きーファイルをコピーして中身を読むー

- ファイル “gauss. f” を各人のディレクトリにコピーする

```
% cp /home/teacher/z6wt01in/SAMPLE/gauss. f .
```

スペース

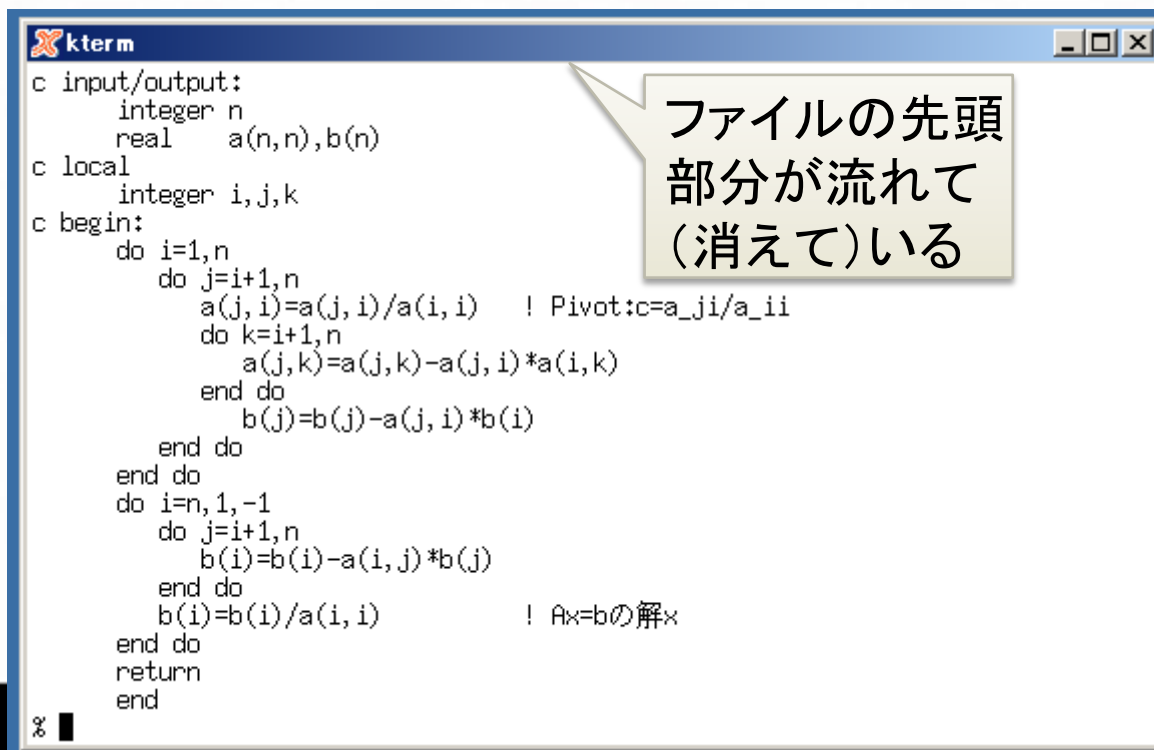
ピリオド

- ファイルの中身を読む（内容を画面に出力する）  
コマンドは “cat” なので、cat コマンドを用いる

```
% cat gauss. f
```

- 1 ページ（ktermの行数）  
に収まらない場合、  
スクロールしてしまう

どうするか？

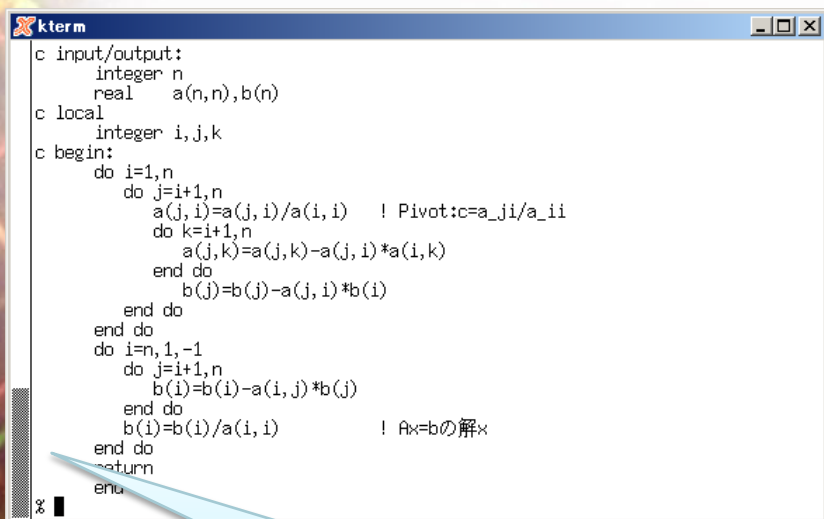


```
kterm
c input/output:
  integer n
  real    a(n,n),b(n)
c local
  integer i,j,k
c begin:
  do i=1,n
    do j=i+1,n
      a(j,i)=a(j,i)/a(i,i)  ! Pivot:c=a_ji/a_ii
      do k=i+1,n
        a(j,k)=a(j,k)-a(j,i)*a(i,k)
      end do
      b(j)=b(j)-a(j,i)*b(i)
    end do
  end do
  do i=n,1,-1
    do j=i+1,n
      b(i)=b(i)-a(i,j)*b(j)
    end do
    b(i)=b(i)/a(i,i)      ! Ax=bの解x
  end do
  return
end
```

ファイルの先頭  
部分が流れて  
(消えて)いる

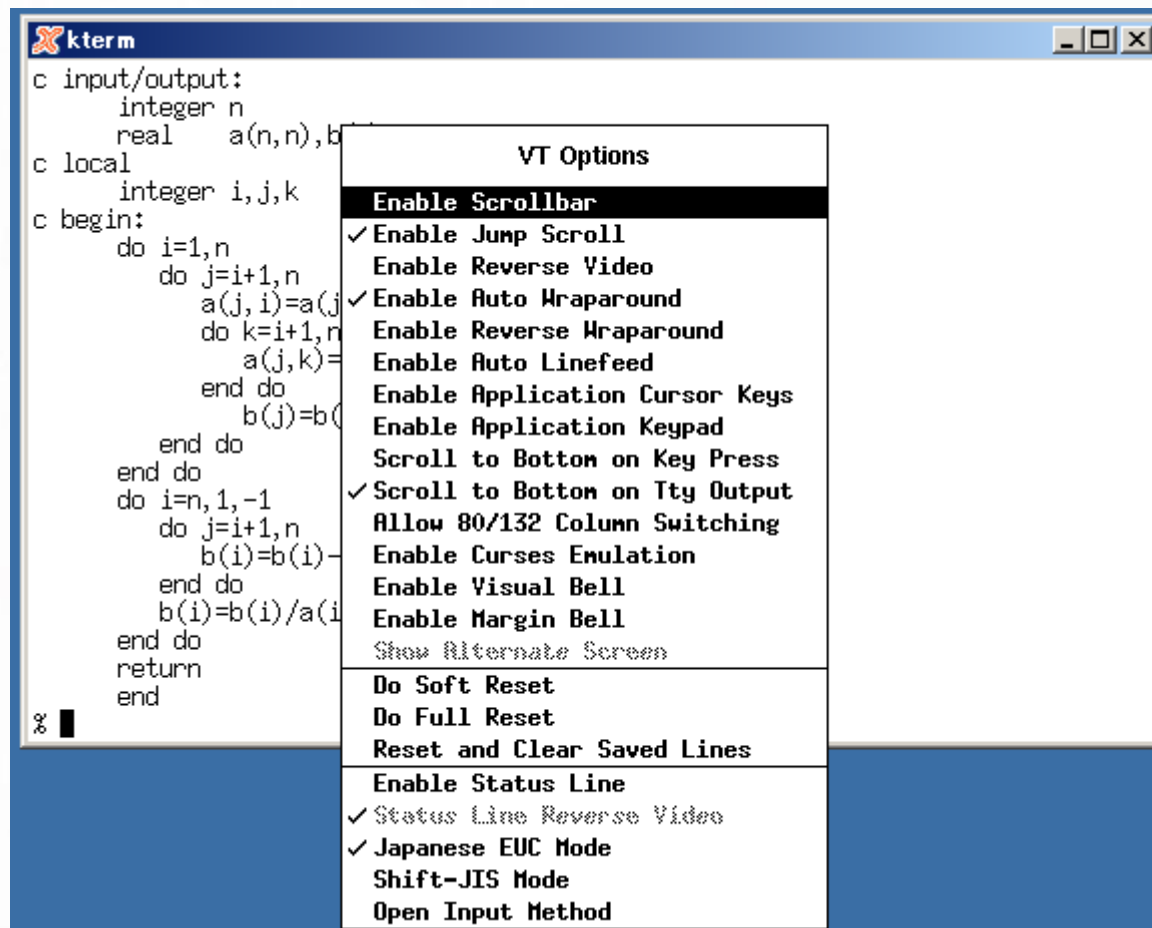
# kterm にスクロール・バーをつけよう

- ・ kterm 内の白い部分(文字の無い部分)にマウスを持っていく
- ・ “ctrl”キーを押しながら、マウスの中ボタンをクリック
- ・ “VT options”というプルダウン・メニューが現れる
- ・ 一番上の  
”Enable Scrollbar”を選択
- ・ スクロールバーが現れる



```
c input/output:
integer n
real a(n,n),b(n)
c local
integer i,j,k
c begin:
do i=1,n
do j=i+1,n
a(j,i)=a(j,i)/a(i,i) ! Pivot:c=a_ji/a_ii
do k=i+1,n
a(j,k)=a(j,k)-a(j,i)*a(i,k)
end do
b(j)=b(j)-a(j,i)*b(i)
end do
end do
do i=n,1,-1
do j=i+1,n
b(i)=b(i)-a(i,j)*b(j)
end do
b(i)=b(i)/a(i,i) ! Ax=bの解x
end do
return
end
```

スクロールバー



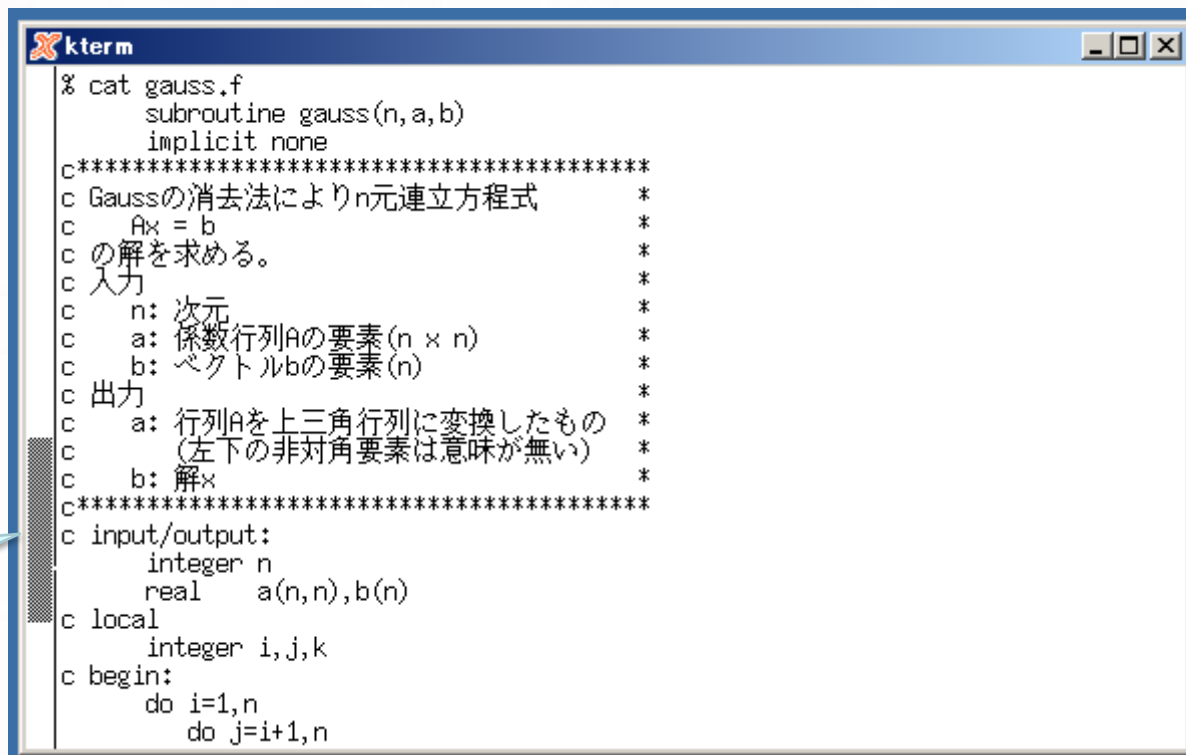
```
c input/output:
integer n
real a(n,n),b(n)
c local
integer i,j,k
c begin:
do i=1,n
do j=i+1,n
a(j,i)=a(j,i)/a(i,i)
do k=i+1,n
a(j,k)=
end do
b(j)=b(
end do
end do
do i=n,1,-1
do j=i+1,n
b(i)=b(i)-
end do
b(i)=b(i)/a(i,i)
end do
return
end
```

VT Options	
<input type="checkbox"/>	Enable Scrollbar
<input checked="" type="checkbox"/>	Enable Jump Scroll
<input type="checkbox"/>	Enable Reverse Video
<input checked="" type="checkbox"/>	Enable Auto Wraparound
<input type="checkbox"/>	Enable Reverse Wraparound
<input type="checkbox"/>	Enable Auto Linefeed
<input type="checkbox"/>	Enable Application Cursor Keys
<input type="checkbox"/>	Enable Application Keypad
<input type="checkbox"/>	Scroll to Bottom on Key Press
<input checked="" type="checkbox"/>	Scroll to Bottom on Tty Output
<input type="checkbox"/>	Allow 80/132 Column Switching
<input type="checkbox"/>	Enable Curses Emulation
<input type="checkbox"/>	Enable Visual Bell
<input type="checkbox"/>	Enable Margin Bell
<input type="checkbox"/>	Show Alternate Screen
Do Soft Reset	
Do Full Reset	
Reset and Clear Saved Lines	
Enable Status Line	
<input checked="" type="checkbox"/>	Status Line Reverse Video
<input checked="" type="checkbox"/>	Japanese EUC Mode
<input type="checkbox"/>	Shift-JIS Mode
<input type="checkbox"/>	Open Input Method

# スクロールバーの使い方

- ・ スクロールバーのグレーの部分は、マウスを持って行き、中ボタンを押している間つかむ事が出来る
- ・ つかみながら、マウスをスクロールバー上を上下すると、流れて画面から消えてしまった部分を見ることができる

マウスの中ボタンを  
押しながら、  
上下する



```
kterm
% cat gauss.f
  subroutine gauss(n,a,b)
    implicit none
c*****
c Gaussの消去法によりn元連立方程式
c   Ax = b
c の解を求める。
c 入力
c   n: 次元
c   a: 係数行列Aの要素(n x n)
c   b: ベクトルbの要素(n)
c 出力
c   a: 行列Aを上三角行列に変換したもの
c     (左下の非対角要素は意味が無い)
c   b: 解x
c*****
c input/output:
c   integer n
c   real    a(n,n),b(n)
c local
c   integer i,j,k
c begin:
c   do i=1,n
c     do j=i+1,n
```

# ファイルの中身の見方ーless コマンドー

- ・ ファイルの中身を確認する便利なコマンドに“less”がある。
- ・ 使い方は、  
    % less gauss.f
- ・ 主なコマンドー一覧
  - “q” キー           less コマンド終了  
                          (ktermのプロンプトに戻る)
  - スペースキー       1 ページ分先に行く
  - “b” キー           1 ページ分元に戻る
  - “↓” キー           1 行分先に行く
  - “↑” キー           1 行分元に戻る

# 単位の認定方法

- ・ 単位は、
  - ・ 数回の演習レポート
  - ・ 最後の課題レポートを評価して決める。課題レポート提出は単位認定の必須条件である。演習レポートは必須ではないが、提出することが薦められる。
- ・ 演習レポートの問題は、講義ノート中に「演習」とあるものの中から選んで出題する。
- ・ 課題レポートは、
  - ・ 物理学に関係がある問題に取り組んでいること
  - ・ 数値計算や数値シミュレーションなどが行われていること
  - ・ 結果や考察において、計算機を使って描いたグラフが使われていること
  - ・ レポートとしての体裁(目的・手法・結果・考察の記述)が整っていることが満たされていれば、どのような問題に取り組んでもよい。問題の例は講義ノート中に「課題」として挙げてある。論文作成には、LaTeXやWordなどの論文清書ツールを使うことが推奨される(必須ではない)

# お知らせ

- ・ 来週(10月9日)の数値計算法は**休講**  
(海外出張中の為)です。
- ・ 次回は10月16日です。
- ・ それまでに、時間を見つけてUnixシステムになれておいて下さい
  - ファイル操作の復習
  - Mule(エディタ)での編集