第 2 版 まえがき

本利用の手引は、九州大学大型計算機センターで利用できるプログラム・ライブラリの中で、応用プログラムに関するガイドブックです。前回 1976 年に発行された改訂第 2 版掲載分については、現在のシステムにあわせて使用方法を書き改め、更に第 2 版では現在サポート中の応用プログラムの紹介を行うなど、全面的な改訂をおこなっています。

「改訂第 1 版 まえがき」の指摘通り、 1976 年時点から「計算に関する需要は、今後質量ともに非常に急激に増加して」ゆきました。センターでサポートするライブラリの中身も、当時から大きく変わってきています。数値計算や統計解析に関するライブラリの優れたものは統合、整理され、汎用パッケージとして提供されました。また、文書整形システムや数式処理システムが研究室や自宅に居ながら利用できる環境になりました。ざっと、 17 年前から主流になったライブラリと、消えていったライブラリを眺めてみると、主流となったライブラリの大きな特徴は、計算機環境を選ばないということです。例えば、統計解析システム SAS はパーソナル・コンピュータでもワークステーションでもセンターの汎用機でも動き、必要なデータは完全に互換が保たれています。これに対し、特定の計算機上の特定の OS でしか動作しないライブラリは、計算機環境の分散とともに、その機動性のなさが欠点となり、最低限必要な出力データの互換性すら保証できず、消えていく運命となりました。

また、研究室単位への計算環境の分散と同じくして、大規模数値計算の需要が急激に増加しています。スーパーコンピュータの急速な大容量化、高速化によって、配列がたくさん取れるようになりました。配列がたくさん取れれば、より「連続」に近づくことができます。「連続に近いもの」がコンピュータ内で表現されれば、研究者はその結果から、より多くの意味ある結果を導き出すことができます。この「速い」と「でかい」は計算機の世界では良いこととされています。皆さんも、何の罪もない一昔前のコンピュータを前に「遅いぞ！」と罵声を浴びせる知り合いの姿を見たことがあると思います。

プログラム言語の世界では、センター開設当時からサポートされている Fortran は、幾つかの改訂を経て、現在は C の挑戦を受けながらも、科学技術計算の主役の座を譲っていません。そして、その配列をこねくりまわす演算との相性のよさを武器に次の世紀をむかえるまで、 Fortran がスーパーコンピュータの主力言語として頑張ることでしょう。センターに最も貢献したライブラリとして、ここに心から感謝する次第です。

この利用の手引に関する意見・質問等は下記へお知らせ下さい．

- 配布に関する問い合わせ先：共同利用掛（内線 2505）
- 内容に関する問い合わせ先：ライブラリ室（内線 2509）
  e-mail : f70029a@kyu-cc.cc.kyushu-u.ac.jp

『利用の手引・ライブラリ編（第 2 版）』 編集作業は，以下の者が担当しました．

研究開発部	: 佐藤 周行，木實 新一，渡部 善隆，畑 三千代，境 理恵子
共同利用掛	: 垣内 良子
システム運用掛	: 浦川 伸治
システム管理掛	: 山嵜 信広，肥田木 直子
ネットワーク掛	: 多久島 智

改訂第1版 まえがき

九州大学大型計算機センターは、大学その他の研究機関における学術研究の環境整備のために、日本学術会議の勧告のもとに、各地に設置された全国共同利用施設の一つであります。設置以来、多くの方々に利用されて来たのでありますが、計算に関する需要は、今後質量ともに非常に急激に増加してゆくと思われます。したがって利用者の計算依頼をいかに迅速に効率よく、しかも公平に処理するかということがセンターの大きな課題であります。

そこでセンターとしては

1. できるだけ多くの人が、気軽に利用できるようにすること。
2. 計算機利用の質および量の変化に即応できること。
3. 限られた職員と予算で、その機能を十分発揮すること。

を目標としております。

しかし、そのためには利用者各位のご理解、ご協力がぜひ必要であります。センターは "全国の研究者のための施設" であるという認識は、運営にたずさわる者だけでなくて、利用者にもひとしく望まれるところであります。

「利用の手引」は以上の趣旨によって定められたセンター利用のための手続きと計算利用方法をまとめたもので、昭和 44 年 3 月 1 日に初版[1]を発行しました。これはその後における全国共同利用センター発行のこの種刊行物の原型となったものであると、いささか自負しておるものであります。

その後、数回の改訂を行なって現在に至っておりますが、今回、これまでのシステムの変更、利用技術のレベルアップなどのより改善された利用方法を整理して、次のように技術的なテーマ別の「専門編」として改編することになりました。旧版以上のお役に立つことを念願するものであります。

1. 手続き編
2. 基本編
3. 制御文編
4. ファイル編
5. ライブラリ編
6. TSS 編
7. XY プロッタ、紙テープ編

この「利用の手引」作成にあたり、企画および執筆の任にあたられたセンター職員の方々の労を多とすると共に、資料の整備、校正に多大のご協力をいただきました福岡女子大学の国宗真教授、九州大学工学部の塩川浩三氏に厚くお礼申し上げます。

昭和 47 年 5 月

九州大学大型計算機センター
広報教育委員会

[1] 九州大学大型計算機センターには、この初版が保管されていません。貴重な資料ですので、もし利用者の方で 1969 年発行の初版をお持ちの方がおられましたら、是非センターに寄贈していただくようお願いします。

OS について

- 九州大学大型計算機センターの計算機システムは、現在ベクトル演算機構を持つ高速演算向きの Fujitsu VP–2600/10 と、汎用大型計算機である Fujitsu M–1800/20 および並列計算機 CM5 から構成されています。計算機システムでは、 MSP/EX と UXP/M の２種類のオペレーティングシステムを運用しています。

- MSP とは Multidimensional System Products の略で、計算機システムを効率よく運転するための OS （オペレーティングシステム）です。具体的には、システムの監視、ジョブ制御などの機能を提供します。

  – オペレーティングシステムとは、ソフトウェアの一種で、コンピュータを構成しているハードウェアと、操作する人との間に立ち、人とハードウェアの間での意志のやり取りを仲介する役目を果たすものです。

- UXP とは UNIX Product の略で、 UNIX のオペレーティングシステムです。

表記について

- ライブラリ作成者の所属は 登録当時 の所属です。所属は現在変動がありますのでご注意下さい.

- データベース作成、提供者は、現在の所属です。

ライブラリの使用について

- ライブラリの利用及び計算結果に関する責任は利用者が負います。

- ライブラリを利用した研究結果を公表する場合、指定された条件に従い、プログラム名、作成者名等を明記して下さい。

- プログラムの著作権者の承認を得ずして他のシステムへの移植、第三者への提供を行なうことは禁じられています。

記事に関する注意点

- ソフトウェア、ハードウェア、及び参考文献は、全て 1993 年 12 月末日 現在の環境です。その後の変更は、適宜センターニュース、広報等でお知らせしています。
- 応用プログラムで、使用方法を詳述した文献が既存のものは、本記事で機能の紹介のみ行なっています。使用方法は参考文献を参照下さい。また、使用方法が前版にしか掲載されていなかった一部のライブラリについては、本手引で詳しい使用方法を掲載しています。
- UXP 関係の参考文献で度々登場する『UXP/M for the Working Scientists』は、 1993 年に行なわれた「UXP 講習会」の資料です。 1993 年 12 月末日段階でまだ publish されておりません。それまでは、 TEX の dvi ファイルが UXP の /usr/local/doc/uxpguide.dvi にありますので参照下さい。センターのプリンターへの出力方法は広報の Vol.27, No.1, pp.9 ~ を御覧下さい。

センター発行の利用の手引について

次の手引が既刊です。ご利用下さい。入手方法は、各連絡所を通して共同利用掛に申し込み下さい。

基本編	TSS 編
バッチジョブ編	ネットワーク編
MSP コマンド編	MSP:TCP/IP 編
ライブラリ編	

表紙の写真	1968 年 6 月 2 日，大型計算機センターに墜落炎上する米軍機.『写真集 九州大学史 1911–1986』(九州大学出版会) より転載
illustration	*Shoudai, Harumi* and *Hidaki, Naoko*

1

プログラミング言語

プログラム言語名	特徴	MSP	UXP	CM5
C	ANSI 規格準拠	▲ [1]	○	
C/VP	VP 用 C		○	
C++	AT&T 版		○	
C*	並列計算用 C			○
Fortran77 EX	Fortran77 の拡張版	○	○	
Fortran77 EX/VP	VP 用 Fortran	○	○	
CM Fortran	並列計算用 Fortran			○
Ratfor	Fortran ベースの言語		○	
Assembler	アセンブリ言語	○		
Cobol	ANSI Cobol 1985	○		
Lisp	Common Lisp 準拠	○	○	
Pascal	ISO 版準拠	▲ [2]	▲ [3]	
PL/I	日本語データ処理機能をサポート	○		
Prolog	DEC10 Prolog 準拠	○	○	

▲ [1] : 外部名が 7 文字を越えてはならないなど UNIX 系からの移植が困難
▲ [2] : 制限事項が多い
▲ [3] : p2c によって C 言語に一度変換して処理を行なっている

一昔前の一部のセンターが計算機資源を完全に独占していた環境から，パーソナルコンピュータ (PC)，ワークステーション (WS) の出現とともに，各研究室単位への仕事の分散化が進んでいる．「大型計算機」で行なわれる仕事は，PC, WS で実行不可能な「大規模」なデータ処理中心になりつつある．従って，今後のプログラム言語は，OS に依存しない移植性の高いものが要求される．

なお，プログラム言語の紹介は『コンピュータソフトウェア事典』（丸善）によった．

1.1 Assembler

【概略】
コンピュータが解釈可能な機械語 (machine language) は 0 と 1 で表現された 2 進法で定義された命令である．アセンブリ言語は，機械語を人間に理解しやすいように解釈する目的で 1950 年代前半に開発された記号言語 (mnemonic code)．アセンブリ言語は，機械語と 1 対 1 の対応関係を持ち，命令コードと memory のアドレスを記号で記述できる．

アセンブラはアセンブリ言語を機械語に翻訳するプログラムのこと．アセンブリ言語の考案によって，それまで機械語レベルで行なっていたプログラミングはかなり楽になり，早く正確なプログラミングが可能となった．しかし，コンピュータの知識のない人がいきなりアセンブリ言語を習得し，プログラムを作成することは容易でない．このため，人間の使う言語に似せたプログラム記述用言語（いわゆる高水準言語）がアセンブリ言語の後に開発されることになる．

【処理環境】
- MSP/M1800
  command : `ASM`
  cataloged procedure : `ASM`

```
*  EXAMPLE OF FORMAT
PROGRAM1 START , FIRST CONTROL SECTION
R1       EQU   1
BASE     EQU   13
     BALR  BASE, 0    SET BASE ADDRESS
     USING *, BASE    ADDRESSABILITY
LOOP1    L      R1, SUM   LOAD SUM
         A      R1, DATA1 COUNT UP
                :
```

assembly language source

【提供】
- 富士通株式会社

【参考文献】
- 「FACOM OS IV アセンブラ文法書」(70SP-5651), 富士通株式会社.
- 「FACOM OS IV/F4 MSP アセンブラ使用手引書」(79SP-3400), 富士通株式会社.
- 「利用の手引・ MSP コマンド編」九州大学大型計算機センター・ライブラリ室, 1993.
- 「利用の手引・バッチジョブ編」九州大学大型計算機センター・ライブラリ室, 1992.
- `HELP ASM`

1.2 C, C/VP

【概略】
UXP, MSP ともに ANSI(American National Standards Institute) 規格準拠 C 言語．

C は D.M.Ritche によって 1972 年に開発されたプログラミング言語．M.Richards が設計した言語 BCPL をもとに K.Thompson が 1970 年に設計した言語 B の更なる発展型である．C は当時開発中であった UNIX の開発を円滑に進めるために導入された．従って UNIX のほとんどは C で記述されている．現在 UNIX の世界を中心として，標準言語として広く流通している．

MSP は標準機能に加えて，日本語処理やアドレス拡張などの機能が追加されている．ただし MSP の場合外部名が 7 文字を越えて指定できないので UNIX で流通している C のプログラムを MSP でコンパイルしようとしてもほとんどこの制限に引っかかり，翻訳できないので注意．C/VP は VP 向けの C 言語．UXP のみのサポート．

【処理環境】

- MSP/M1800
  command : `CC, FCC`
- UXP/M1800, UXP/VP2600
  command : `cc(/usr/ccs/bin/cc), vcc`

```
#include <stdio.h>
#include <math.h>

main()
{
  double x;
  for(x=-2;x<=10;x+=2)
  {
   int n;
   double y=frexp(x,&n);
   printf("%3g=% f*2^%d\n",x,y,n);
  }
}
```

C source

【注意事項】

- C/VP によってベクトル化される `for` ループは，以下の条件を満たす必要がある（かなり厳しい：Fortran の DO ループに素直に読みかえられるのが対象）．

  1. 一般に変数 `N`，`M`，`I` に対して以下のように書ける．変数 `M` は変わってはいけない．

```
for(i = N; i < M; i += I) {
      ......
}
```

2. 実行開始時に繰り返し回数が確定していること．途中の飛び出しは１回のみ可．
3. ループ途中への飛び込みは不可．

【提供】

- 富士通株式会社

【参考文献】

- 「C 言語文法書」(99SP-8050)，富士通株式会社．
- 「OS IV C 言語使用手引書 V12」(70SP-5631), 富士通株式会社.
- 「UXP/M C 言語使用手引書 V11L10 用」(94SP-5050), 富士通株式会社.
- 「OS IV C 言語メッセージ説明書 V12」(70SP-5640), 富士通株式会社.
- 「利用の手引・ MSP コマンド編」九州大学大型計算機センター・ライブラリ室, 1993.
- 「UXP/M for the Working Scientists」佐藤 周行, 1993.
- 「Unix 環境でスーパーコンピューティングをする人のためのガイド」佐藤 周行, 九州大学大型計算機センター広報, Vol.26, No.4, pp.452–484, 1993.
- man cc で参照可能 (UXP).

1.3 C++

【概略】

AT&T 版をサポート．他のシステム（ex. GNU C++）などと言語仕様，ランタイムルーチンの振舞いが微妙に異なるので注意が必要．

C++ は AT&T ベル研究所の B.Stroustrup により 1983 年に開発された言語．仕様は C 言語をほぼ包含したものである．C 言語に比べて新たに，タイプチェック機能，データ抽象化機能，演算子の overloading，オブジェクト指向のプログラミングが行なえる機能などが追加されている．

【処理環境】

- UXP/M1800
  command : `CC(/usr/uxp/CC)`

```
#include <stream.h>

main(int argc,char *argv[])
{
  int iend=atoi(argv[1]);
  int i,sum=0;

  for(i=1;i<=iend;i++)
  {
    sum+=i;
  }
  cont << "SUM="  << sum "\n";
}
```

C++ source

【提供】

- 富士通株式会社

【参考文献】

- 「The C++ Programming Language」B.Stroustrup, Addison Wesley, 1989.
- 「UXP/M for the Working Scientists」佐藤 周行, 1993.
- `man CC`

1.4 C*

【概略】

並列計算機 CM-5 のために開発された並列 C 言語．ANSI C の全ての機能を包含する．並列処理機能として，並列変数の形の定義，各仮想プロセッサ管理，並列データの参照があげられる．

【処理環境】

- CM-5
  command : cs

```
#include <stdlib.h>

main()
{
  shape [8192]S;
  int:S source, sum;

  with(S)
  {
     source = prand() %% 100;

     sum = 0;

     [source]sum += 1;
  }
}
```

C* source

【提供】

- Thinking Machines Corporation

【参考文献】

- 「C* User's Guide」Thinking Machines Corporation.

1.5 Cobol85

【概略】

Cobol(Common business oriented language) は 1959 年アメリカ国防省が主催した CODASYL(The Conference On Data Systems Languages) 委員会での議論をもとに開発された事務処理用の言語．MSP 上の Cobol は ANSI Cobol 1985 の機能単位の最高水準と，その他いくつかの拡張機能を包含した Cobol 85 の文法に拠る．大量で複雑なデータ構造を扱うために，大容量ファイルの検索や更新，併合機能や文字列操作機能，報告書作成機能などの特徴を持っている．

【処理環境】

- MSP/M1800
  command : `COBOL`
  cataloged procedure : `COBOL`

```
DATA DIVISION.
FILE SECTION.
FD    IN-FILE
      RECORD IS VARYING IN SIZE.
01    IN-REC.
      02 C-FIELD.
        03 WIN      PIC X(8).
        03 FILLER   PIC XX.
          04 I1IN   PIC X.
          04 I2IN   PIC X(20).
        03 FILLER   PIC X(100).
      02 IN-MSG
          :
```

Cobol source

【提供】

- 富士通株式会社

【参考文献】

- 「FACOM OS IV COBOL85 使用手引書」(70SP-5812), 富士通株式会社．
- 「FACOM OS IV COBOL85 文法書」(70SP-5801), 富士通株式会社．
- 「利用の手引・ MSP コマンド編」九州大学大型計算機センター・ライブラリ室, 1993.
- `HELP COBOL`

1.6 Fortran77 EX, Fortran77 EX/VP

【概略】

Fortran77/EX は，Fortran77 の拡張言語．次期言語である Fortran90 への移行段階に位置するプログラミング言語処理体系．Fortran77 EX/VP は VP2600 上で高速実行のための自動ベクトル化コンパイラ．実行はバッチのみに限られる．Fortran(Formular Translation) は 1957 年に J.W.Backus らによって開発された科学技術数値計算言語．当時開発中であったスーパーコンピュータ IBM 704 専用の高水準言語として設計された．その後，言語の処理系，設計の優秀さから IBM 社の他の機種および他社の機種にも採用され，さまざまな高水準言語の規範となった，Fortran の成功を直接の基礎とした言語に Algol, Cobol, PL/I, Basic などがある．

【処理環境】

- MSP/M1800, MSP/VP2600
  command : `FORT`, cataloged procedure : `FORT`
- UXP/M1800, UXP/VP2600
  command : `fort(/usr/local/bin/fort)`, `frt(/usr/uxp/frt)`

```
      PROGRAM TEST1
      PARAMETER(MAXN=14)
      REAL*8 M(MAXN)
      WRITE(6,*) 'N='
      READ(5,*) N       ! THIS IS COMMENT
C
      DO 10 I=1,MAXN
        K=I*2
        J=K-I-4
   10 CONTINUE
          :
```

Fortran77/EX source

【提供】

- 富士通株式会社

【注意事項】

- Fortran77/EX は，他の Fortran77 コンパイラと基本的に互換である．ただし，文法チェックが非常に厳しい．従って，パソコン，ワークステーションで一見エラーなしで動作していたプログラムが，センターの Fortran77/EX ではエラーメッセージを出して実行が中断される場合がある．この場合，ほとんどのケースが利用者のプログラムのバグである．移行の際は十分なチェックをすること．逆に言えば，Fortran77/EX を使って，自分のプログラムのチェックができる．

【Fortran77/EX の特徴】

- Fortran90 の一部先行採用
  - 文字 ! によって，行の途中から注釈が書ける
  - 暗黙の型宣言が抑止できる．
  - 英小文字およびアンダースコアー（下線）が使用可能
  - 名前（変数名や定数名など）の最大長は 31 文字

```
program main       ! コメントです
 implicit none
 real a,b,c_ext
 real x12345678
end
```

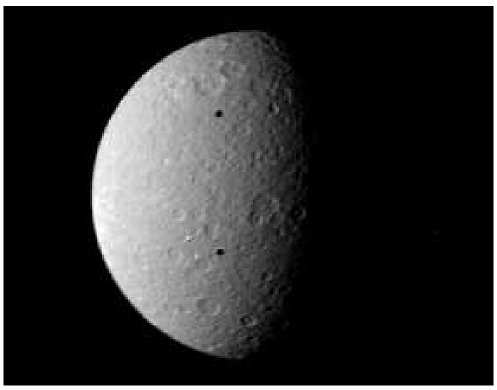
- 1 文の継続可能行数は 99 行，関数やサブルーチンの引数の最大個数は 1000 個．

```
  function longlong(L1,L2,L3,L4,h)
  longlong= (-3*h*L1 - 2*L3)*L4**3/(18*h**6*L1**3*L3) +
&  L2**2*L3**2*(L2*L3 - L1*L4)/(6*h**6*L1**6) +
                :
         継続が 99 行まで可能
                :
&  (-6*h*L1*L2**2*L3 - 10*L2**2*L3**2 + 3*h*L1**2*L2*L4 +
&  8*L1*L2*L3*L4 - L1**2*L4**2)*Log(-L2 + L1*L4/L3)/(3*h**6)
  end
```

【参考文献】

- 「OS IV/MSP FORTRAN77 EX 使用手引書」(79SP-5031)，富士通株式会社．
- 「OS IV/MSP FORTRAN77 EX/VP 使用手引書 V12 用」(79SP-5041)，富士通株式会社
- 「OS IV FORTRAN77 EX メッセージ説明書 V12 用」(70SP-5321)，富士通株式会社．
- 「FORTRAN 新コンパイラの公開について」 竹生 政資, 九州大学大型計算機センター広報, Vol.25, No.1, pp.55–60, 1992.
- 「MSP/FORTRAN を VP で実行するには」肥田木 直子，渡部 善隆，九州大学大型計算機センター広報, Vol.26, No.4, pp.397–451, 1993
- 「Unix 環境でスーパーコンピューティングをする人のためのガイド」佐藤 周行，九州大学大型計算機センター広報, Vol.26, No.4, pp.452–484, 1993.

- 「MSP/FORTRAN 利用法」 畑 三千代，渡部 善隆，国宗 真，肥田木 直子, 九州大学大型計算機センター広報, Vol.26, No.3, pp.232–293, 1993.
- 「利用の手引・バッチジョブ編」九州大学大型計算機センター・ライブラリ室, 1992.
- 「UXP/M for the Working Scientists」佐藤 周行, 1993.
- 「利用の手引・ MSP コマンド編」九州大学大型計算機センター・ライブラリ室, 1993.
- `man frt`(UXP)
- `HELP FORT`(MSP)
- 「利用の手引・ TSS 編」九州大学大型計算機センター・ネットワーク室, 1993.



1.7 CM Fortran

【概略】

並列計算機 CM-5 のために開発された並列 Fortran 言語．Fortran77 に Fortran90 の配列処理部分を追加したもの．全ての Fortran77 の操作，固有関数を採用している．並列計算用の配列のための機能として，プロセッサごとに配列を分割し，並列のアクセスを行なうための CM 配列が用意されている．

【処理環境】

- CM-5
  command : `cmf`

```
      PROGRAM TEST
      REAL C(0:100),F(0:100)
      PARAMETER(SCALE=9.0/5.0)
      PARAMETER(OFFSET=32.0)

C     Compute values of C and F:
      C=[0:100]
      F=SCALE*C+OFFSET

C     Output results:
      PRINT 10,(C(I),F(I),I=0,100)
   10 FORMAT(F9.3,3X,F9.3)
      END
```

CM Fortran source

【提供】

- Thinking Machines Corporation

【参考文献】

- 「CM Fortran Programming Guide」Thinking Machines Corporation.

1.8 Lisp

【概略】
Lisp 処理系の国際標準となりつつある Common Lisp の言語使用に準拠した実用的な知識システム開発言語．UTILisp を包含．Lisp は 1950 年代後半に，MIT の J.McCarthy によって考案された．Lisp の最大の特徴は，言語それ自体が独立したものではなく，対話型の処理系に組み込まれている点である．プログラムの作成，テスト，修正，コンパイル，実行に至るまでプログラム開発の全ての側面において処理系と対話するための言語が Lisp である．

【処理環境】

- MSP/M1800
  command : `LISP`
- UXP/M1800
  command : `lisp(/usr/uxp/lisp)`

```
: (defun sum (l)
:   (if (null l)
:     0
:     (+ (car l) (sum (cdr l)))))
SUM
: (sum '(1 2 3 4))
10
```

Lisp source

【提供】

- 富士通株式会社

【参考文献】

- 「富士通 LISP 説明書」(99SP-4001)，富士通株式会社．
- `man lisp`(UXP)
- `HELP LISP`(MSP)
- 「利用の手引・MSP コマンド編」九州大学大型計算機センター・ライブラリ室, 1993.
- 「UXP/M for the Working Scientists」佐藤 周行, 1993.

1.9 Pascal

【概略】
ISO 版 Pascal 規格案に準拠．Pascal は N.Wirth により 1975 年に開発されたプログラミング教育用の言語．記述様式は Algol 60 を引継ぎ，構造化されたプログラムを作成するのに適切な言語仕様になっている．UXP は p2c コマンドによって，Pascal ソースを C ソースに変換したのち処理を行なっている．

【処理環境】

- MSP/M1800
  command : `PASCAL`
  cataloged procedure : `PASCAL`
- UXP/M1800
  command : `pc(/usr/local/bin/pc)`

```
program test;
var a,b : integer;
ratio : real;
begin
 a:=9;
 b:=7;
  ratio:=a/b;
  write('Ratio=',ratio);
end.
```

Pascal source

【提供】

- 富士通株式会社

【参考文献】

- 「FACOM OS IV MSP PASCAL 使用手引書」(78SP-5551), 富士通株式会社．
- 「FACOM OS IV PASCAL 文法書」(64SP-3340), 富士通株式会社．
- 「利用の手引・MSP コマンド編」九州大学大型計算機センター・ライブラリ室, 1993.
- 「UXP/M for the Working Scientists」佐藤 周行, 1993.
- HELP PASCAL

1.10 PL/I

【概略】

PL/I は Programming Language one の簡略形であり，1965 年に IBM によって開発された言語．科学技術計算や事務計算を含んだ広範囲のアプリケーションに使用できるようなプログラミング言語を目的としている．多種のデータ構造やその上での操作，例外処理やリアルタイム処理を記述する機能など豊富な機能を備える．

【処理環境】

- MSP/M1800
  command : `PLI`
  cataloged procedure : `PLI`

```
TEST : PROCEDURE OPTIONS(MAIN) ;
       DECLARE X ;
       ON ENDFILE(SYSIN) GO TO OWARI ;
YOMU : GET LIST(X) ;
       PUT LIST(X) ;
       IF X>=0 THEN X=SQRT(X) ;
               ELSE X=SQRT(-X) ;
       PUT LIST(X) SKIP(2) ;
       GO TO YOMU ;
OWARI : END TEST ;
```

PL/I source

【提供】

- 富士通株式会社

【参考文献】

- 「FACOM OS IV MSP PL/I 使用手引書」(78SP-5412), 富士通株式会社．
- 「OS IV PL/I 文法書」(70SP-5404), 富士通株式会社．
- 「FACOM OS IV TESTPLI 使用説明書」(70SP-5420), 富士通株式会社．
- 「利用の手引・ MSP コマンド編」九州大学大型計算機センター・ライブラリ室, 1993.
- HELP PLI

1.11 Prolog

【概略】

Prolog(Programming in logic) は 1972 年に開発されたホーン論理に基づくプログラミング言語．DEC-10 Prolog に準拠し，いくつかの機能拡張を行なっている．Prolog は Lisp と並び，文字列の検索や書換に便利であり，知識表現や推論のシステムを作るのに容易なため，人工知能言語であると一般にいわれている．

【処理環境】

- MSP/M1800
  command : `PROLOG`
- UXP/M1800
  command : `prolog(/usr/uxp/prolog)`

```
love(kaihatsu,gyomu).
love(gyomu,jimu).
friend(X,Y) :- love(X,Y).
friend(X,Y) :- love(X,Z),friend(Z,Y).

| ?-friend(kaihatsu,jimu). ↵
yes
| ?-friend(jimu,gyoumu). ↵
yes
| ?-halt. ↵
```

Prolog source

【提供】

- 富士通株式会社

【参考文献】

- 「富士通 PROLOG 手引書」(99SP4060), 富士通株式会社．
- 「利用の手引・MSP コマンド編」九州大学大型計算機センター・ライブラリ室, 1993.
- 「UXP/M for the Working Scientists」佐藤 周行, 1993.
- `man prolog`(UXP)
- `HELP PROLOG`(MSP)

1.12 Ratfor

【概略】
Ratfort(Rational fortran) は AT&T ベル研究所で開発された Fortran に基づくプログラム言語．プリプロセッサ方式であり， Ratfort 語で書かれたプログラムをプリプロセッサによって Fortran 語のプログラムに翻訳し，更に Fortran コンパイラで翻訳，実行を行なう．

【処理環境】

- UXP(M1800)

【処理環境】

- UXP/M1800
  command : `ratfor(/usr/local/bin/ratfor)`

```
#program test
defin(eof,11111)
integer r
   100 format(i5)
for(read(5,*) n;n^=eof ; read(5,*)n){
   if(n<=1) { #illegal input ?
     write(6,*) n
    }
   else if(n==2) {    #n=2 ?
    write(6,*) n
    }
        :
```

Ratfor source

【提供】

- AT&T

【参考文献】

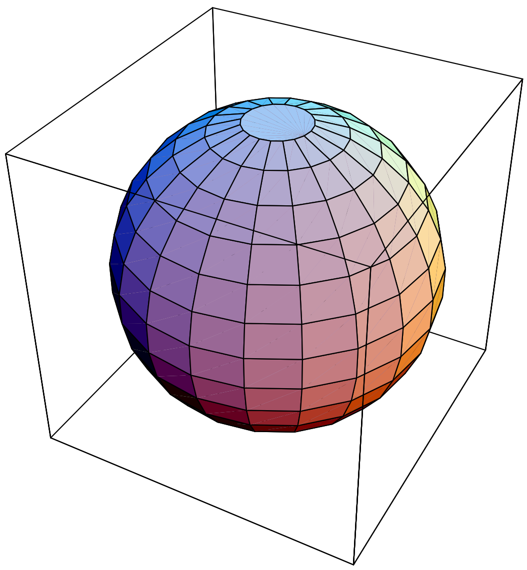
- 「UXP/M for the Working Scientists」佐藤 周行, 1993.

2

プログラム言語処理支援

プログラミング言語処理支援として，次のソフトウェアがある．

プログラム名	特徴	MSP	UXP
Analyzer	Fortran 動作状況解析	○	○
Tuner	Fortran チューニング支援	○	○
f2c	Fortran ソースを C に変換		○
p2c	Pascal ソースを C に変換		○
Debugger	Fortran, C のデバッグ支援	○	
sdb	シンボリックデバッガ		○
Wisepack	Fortran プログラムの開発・保守支援	○	
lex	簡単な字句解析プログラムの生成		○
yacc	構文解析プログラム生成ツール		○
bison	GNU Project parser generator		○
m4	マクロ・プロセッサ		○



2.1 Analyzer

【概略】

Fortran プログラムの動作情報を収集および分析して，チューニング作業（ベクトル計算機上での実行性能の改善作業）を支援するツール．

一般にプログラムの中で最も実行時間を費やす部分は，プログラムの特定の場所に偏る．チューニング作業の大きな目的は，この実行時間が多くかかる部分を改善することにある．アナライザはその支援として，実行頻度，実行回数，実行時間，コスト値などのプログラム動作情報を提供する．また，TSS でのチューニング支援ツール Tuner に渡す情報を採取する．

【処理環境】

- MSP/M1800, MSP/VP2600
  cataloged procedure : `ANALYZE`
- UXP/M1800, UXP/VP2600
  command : `afrt(/usr/uxp/afrt)`

```
vectorize - total list ------------------------------------------------------------------

name       ex-count     v-cost     %     s-cost     %     v-leng  v-rate      v-effect
MAIN             *1   *2206017  54.6   *2206017   4.3         *5    *0.0    *1.0-  1.0
SETTBL           *1   *1210793  30.0  *33097172  65.1       *100   *99.8   *22.2- 34.0
SETLST           *1    *526458  13.0  *13082014  25.7       *100   *99.4   *20.5- 30.2
CF22M            *1     *42546   1.1   *1177955   2.3        *99   *99.8   *22.4- 34.6
CF23M            *1     *39020   1.0   *1114065   2.2        *99   *99.9   *23.0- 35.9
SFOOM            *1      *5460   0.1    *131016   0.3        *99   *99.2   *19.9- 29.0
CF2              *1      *3030   0.1     *16358   0.0        *84   *84.4    *5.2-  5.6
FFT2             *1      *2845   0.1      *4583   0.0        *67   *39.3    *1.6-  1.6
(total)   ---------   *4037726 100.0  *50868886 100.0  ---------   *95.3   *11.4- 13.8

vectorize - routine list ----------------------------------------------------------------

routine    ex-count     v-cost     %     s-cost     %     v-leng  v-rate      v-effect
MAIN             *1   *2206017  54.6   *2206017   4.3         *5    *0.0    *1.0-  1.0

isn       ex-count   true    v-cost    s-cost v o c---------1---------2---------3------
00000001                          IMPLICIT REAL*8 (A-H,O-Z)
00000002                          PARAMETER (NMAX=5000,MMAX=12)
00000003                          COMPLEX*16 C(NMAX),OMEGK(NMAX,MMAX,2),X(NMAX)
00000004                          COMPLEX*16 C1(NMAX),C2(NMAX)
00000005                          REAL*8 ER(NMAX)
00000006                          INTEGER LIST(NMAX*MMAX)
                                          :
```

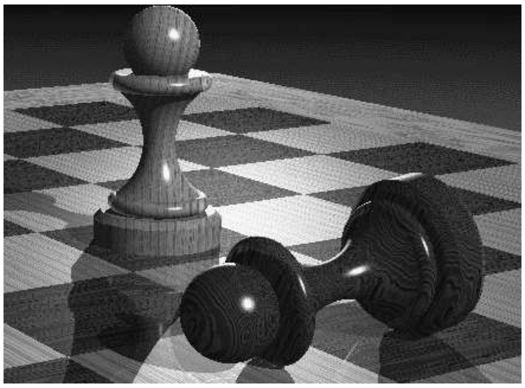
Analyzer による Fortran プログラム解析情報

【提供】

- 富士通株式会社

【参考文献】

- 「FACOM OS IV/MSP アナライザ使用手引書(FORTRAN,VP 用) V10L20 用」(79SP-5080),富士通株式会社.
- 「利用の手引・バッチジョブ編」九州大学大型計算機センター・ライブラリ室, 1992.
- 「MSP/FORTRAN を VP で実行するには」 肥田木 直子,渡部 善隆,九州大学大型計算機センター広報, Vol.26, No.4, pp.397–451, 1993
- 「Unix 環境でスーパーコンピューティングをする人のためのガイド」佐藤 周行,九州大学大型計算機センター広報, Vol.26, No.4, pp.452–484, 1993.
- man afrt(UXP)



2.2 bison

【概略】

構文解析プログラムツール．yacc よりも強力な機能を有する．

Bison is a parser generator in the style of yacc(1). It should be upwardly compatible with input files designed for yacc. Input files should follow the yacc convention of ending in ".y". Unlike yacc, the generated files do not have fixed names, but instead use the prefix of the input file.

For instance, a grammar description file named parse.y would-produce the generated parser in a file named `parse.tab.c`, instead of yacc's `y.tab.c`.

【処理環境】

- UXP/M1800
  command : `bison(/usr/local/bin/bison)`

【提供】

- GNU

【参考文献】

- 「UXP/M for the Working Scientists」佐藤 周行, 1993.
- `nroff -man /usr/local/man/man1/bison.1|more`

2.3 Debugger

【概略】
Fortran 言語，C 言語で作成したロードモジュールを対象にデバッグを行う．MSP の DEBUGGER コマンドを使用するためには，あらかじめデバッガ用の情報を付加したロードモジュールを作成する必要がある．

また，FORT コマンドに GODBG オプションを指定することで editor 内からデバッガを起動することもできる．

【処理環境】

- MSP/M1800
  command : DEBUGGER, DEB

```
DBG TEST.LOAD(EX) ↵
 DEBUG/I
B 200 IN(SUB) ↵
 DEBUG/I
C ↵
 'SUB'の NUM  :  200 で中断しました．
 DEBUG
```

debugger を用いたデバッグ例

【提供】

- 富士通株式会社

【参考文献】

- 「OS IV デバッガ使用手引書 FORTRAN,C 言語用」(70SP-6430) 富士通株式会社．
- 「デバッガ機能の紹介」渡部 善隆，平尾 耕二，九州大学大型計算機センター広報, Vol.26, No.1, pp.1–47, 1993.

2.4 f2c

【概略】

Fortran のソースプログラム (~.f) を C(~.c) に変換する。変換したファイルを使って実行ファイルを作る場合，リンクの際にオプション

```
-L/usr/local/lib -lF77 -lI77
```

を指定すること。

Fortran 用のライブラリ (SSL2 など) を使ったものは，リンク時にそれも指定すること．

【処理環境】

- UXP/M1800
  command : `f2c(/usr/local/bin/f2c)`

【使用例】

- `Fortran` ソースファイル `a.f` を `C` ソースプログラム `a.c` に変換する．さらに，実行ファイルを作成する．

```
kyu-cc% f2c a.f ↵
kyu-cc% cc a.c -L/usr/local/lib -lF77 -lI7 ↵
```

【提供】

- AT&T

【参考文献】

- 「UXP/M for the Working Scientists」佐藤 周行, 1993.

2.5　lex

【概略】
テキストの簡単な字句解析を行うプログラムを作成する。入力ファイル file (省略時は標準入力) は，検索の対象である文字列と式，および文字列が見つかったときに実行する C プログラムのテキストを含む．

lex は `lex.yy.c` というファイルを生成する．`lex.yy.c` をコンパイルし lex ライブラリとリンクすると，このプログラムは指定した文字列が検出された場合を除き，入力を出力に複写する．文字列が検出された場合は，対応するプログラム・テキストを実行する．一致した実際の文字列は，外部文字配列 yytext に入る．

【処理環境】

- UXP/M1800
  command : `lex(/usr/ccs/bin/lex)`

【提供】

- 富士通株式会社

【参考文献】

- `man lex`
- 「UXP/M for the Working Scientists」佐藤 周行, 1993.

2.6 m4

【概略】
C 言語，アセンブラ言語，あるいはその他の言語のフロント・エンドとして使用するためのマクロ・プロセッサ．引数として指定されたファイルは，その順に処理される．ファイルが指定されていない場合，あるいはファイル名が - である場合には標準入力が読み取られる．処理されたテキストは標準出力に書き込まれる．

【処理環境】

- UXP/M1800
  command : `m4(/usr/ccs/bin/m4)`

【提供】

- 富士通株式会社

【参考文献】

- `man m4`
- 「UXP/M for the Working Scientists」佐藤 周行, 1993.

2.7　p2c

【概略】
UXP では Pascal コンパイラが提供されていないので Pascal のソースファイルは C などにいったん変換する必要がある．

p2c は Pascal のソースプログラム (～.p) を C(～.c) に変換する．変換したファイルを使って実行ファイルを作る場合，リンクの際にオプション

`/usr/local/lib/libp2c.a`

を指定すること．

【処理環境】

- UXP/M1800
  command : `p2c(/usr/local/bin/p2c)`

【使用例】

- `Pascal` ソースファイル `a.p` を `C` ソースプログラム `a.c` に変換する．さらに，実行ファイルを作成する．

```
kyu-cc% p2c a.p ↵
kyu-cc% cc a.c /usr/local/lib/libp2c.a ↵
```

【提供】

- Dave Gillespie

【参考文献】

- 「UXP/M for the Working Scientists」佐藤 周行, 1993.

2.8　sdb

【概略】

sdb(Symbolic DeBugger) は C, Fortran，アセンブラ用のシンボリックデバッガ．sdb は実行可能ファイルとコア・ファイルを調べるのに利用できる．また制御された実行環境において，実行中のプロセスを調べることも可能．各ファイルはデバッグ用のフラグ (-g) を立ててコンパイルすること．

【処理環境】

- UXP/M1800
  command : `sdb(/usr/ccs/bin/sdb)`

【提供】

- 富士通株式会社

【参考文献】

- 「UXP/M for the Working Scientists」佐藤 周行, 1993.
- `man sdb`

2.9 Tuner

【概略】
チューナは Fortran77 EX プログラムをベクトル計算機 (VP2600) 上でより高速に動作させるためのチューニング作業を支援するシステム．Analyzer 解析情報ファイル内の項目を表示し，チューニングに有効となるガイダンスメッセージを表示する．

【処理環境】

- MSP/M1800
  command : `TUNER`
- UXP/M1800
  command : `tuner(/usr/uxp/tuner)`

【注意事項】

- Tuner でチューニング作業を行なうためには，あらかじめ Analyzer によって，解析用のデータを収集しておく必要がある．

【提供】

- 富士通株式会社

【参考文献】

- 「OS IV/MSP チューナ使用手引書」(79SP-5041) 富士通株式会社．
- 「チューナの公開について」 九州大学大型計算機センター広報, Vol.25, No.3, pp.228–230, 1992.
- 「アナライザの公開について」九州大学大型計算機センター広報, Vol.25, No.3, pp.231–232, 1992.
- `man tuner`(UXP)
- 「利用の手引・MSP コマンド編」九州大学大型計算機センター・ライブラリ室, 1993.
- 「UXP/M for the Working Scientists」佐藤 周行, 1993.

2.10 yacc

【概略】
文脈自由文法を，LALR(1) 構文解析アルゴリズムを実行するオートマトンのためのテーブルに変換する．出力ファイル `y.tab.c` は，プログラム yyparse を作成するために C コンパイラでコンパイルする必要がある．このプログラムは，字句解析プログラム yylex ならびに main とエラー処理ルーチン yyerror とともにロードする必要がある．これらのルーチンは，user が供給する．yacc が使用できる字句解析プログラムの作成には，lex が便利．

【処理環境】

- UXP/M1800
  command : `yacc(/usr/ccs/bin/yacc)`

【提供】

- 富士通株式会社

【参考文献】

- 「UXP/M for the Working Scientists」佐藤 周行, 1993.
- `man yacc`

2.11 Wisepack

【概略】

Fortran77 プログラムを静的解析し，プログラムの開発保守にかかわる作業を支援するシステム．解析機能，ドキュメント生成機能及びファイル操作機能を有する．

【処理環境】

- MSP/M1800
  command : `WISEPACK`
  cataloged procedure : `WISEPACK`

【提供】

- 富士通株式会社

Wisepack 起動メニュー

【参考文献】

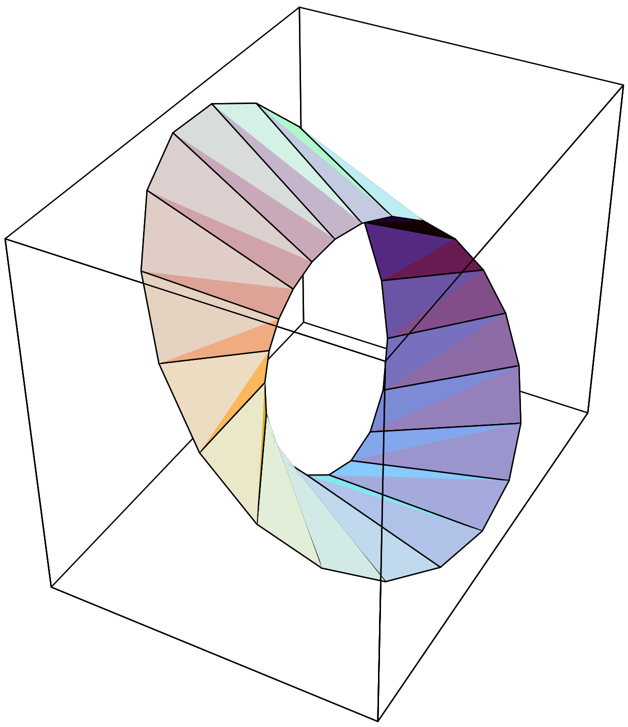
- 「FACOM OS IV / F4 MSP WISEPACK 説明書」(78AR-3050)，富士通株式会社．
- 「FORTRAN プログラム開発・保守支援パッケージ WISEPACK の公開について」九州大学大型計算機センター広報, Vol.21, No.2, pp.144–146, 1988.
- 「利用の手引・ MSP コマンド編」九州大学大型計算機センター・ライブラリ室, 1993.
- 「利用の手引・バッチジョブ編」九州大学大型計算機センター・ライブラリ室, 1992.

3

数式処理

UXP の数式処理システムはバックエンドで動いている．そのため，UXP のファイルを用いるためにはバックエンドのワークステーションと UXP との間でファイル転送をおこなう必要がある．MSP の Reduce は M1800 上の動作．

ソフトウェア名	特徴	MSP	UXP
Mathematica	Graphics 表示機能を持つ		○
Reduce	日本語対応版	○	○



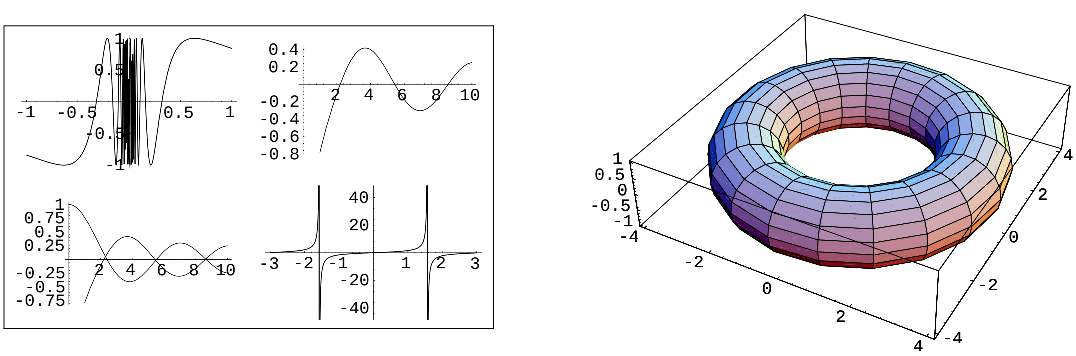
3.1　Mathematica

【概略】

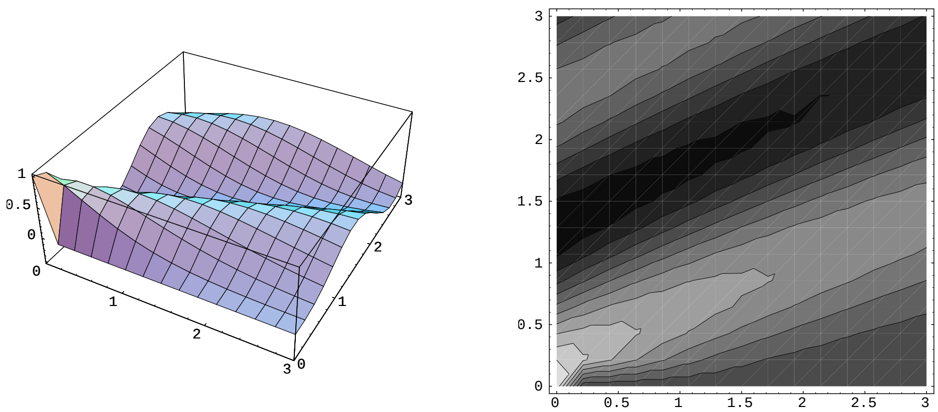
Mathematica は Stephen Wolfram により開発された汎用数学処理システム．高度な計算機能，数式処理機能，2 次元，3 次元グラフィックス機能を有する．また，数式は Fortran, C, TEX ソースとして出力可能．グラフィックス出力は PostScript コードで出力される．グラフィックスは X Window システムと Tektronix のグラフィックシステムに対応している。 X-server の名前を指定するか，Tek.m を読み込むことで対応できる．

【処理環境】

- UXP/M1800(バックエンド SUN4 上の動作)
  command : `math(/usr/local/bin/math)`, `rmath(/usr/local/bin/rmath)`



Mathematica graphics 1



Mathematica graphics 2

【使用例】

- Tektronix グラフィックスを用いる .

```
kyu-cc% math ↵
wait a moment ........

Input Your X-server host-name:
Input  <<Tek.m   ,when your terminal is Tektro4014/4010 Terminal.

Mathematica 2.0 for SPARC
Copyright 1988-91 Wolfram Research, Inc.
 -- Terminal graphics initialized --

In[1]:= <<Tek.m ↵
 -- Tektronix graphics initialized --
In[2]:=
```

【提供】

- Wolfram Research, Inc.

【参考文献】

- 「汎用数式処理ツール Mathematica の紹介」尾崎 敬二，九州大学大型計算機センター広報, Vol.24, No.5, pp.501–522, 1991.
- 「問題解決ツールとしての数式処理ツール Mathematica」尾崎 敬二，九州大学大型計算機センター広報, Vol.26, No.1, pp.48–63, 1993.
- 「Mathematica: a system for doing mathematics by computer, 2nd ed.」Stephen Wolfram, Addison Wesley, 1991.
- 「UXP/M for the Working Scientists」佐藤 周行, 1993.

3.2 Reduce

【概略】
Reduce は Anthony C.Hearn が量子電磁力学における計算のため Lisp を用いて 1968 年に作成した数式処理システム．システムと機能が簡素であり，演算速度も速い．UXP の Reduce はバックエンドで動いているため，UXP のファイルを Reduce で用いるためにはバックエンドと UXP との間でファイル転送をする必要がある．

【処理環境】

- MSP/M1800
  command : `REDUCE`, cataloged procedure : `REDUCE33`
- UXP/M1800(バックエンドの SUN4 上の動作)
  command : `reduce(/usr/local/bin/reduce)`

【使用例】

- `MSP/Reduce` には`'LIB.REDUCE33.XMPLIB'` に豊富な入門用使用例が用意されている．このデータセットは，`REDUCE` コマンドを実行させると，`DD` 名 `XMPLIB` で自動的に組み込まれるので，例のように簡単に実行でき，`Reduce` の利用方法を知ることができる．

```
 READY
REDUCE  ↵
 ** REDUCE 3.3 BEGIN **
 SLISP : 4686424 BYTES
 REDUCE 3.3, 15-JAN-88...
 1:
IN XMPLIB(メンバ名);  ↵
（例題実行開始）
```

  メンバ名としては `ALGINT, ANUM, BFLOAT, COMPLEX, EXCALC, EZGCD, FACT, GENTRN, GROEBNER, INT, LESS1 ~ LESS7, SOLVE, SPDE, TEST` が使用できる．

【提供】

- 富士通株式会社 (MSP)
- Forbs Co.(UXP)

【参考文献】

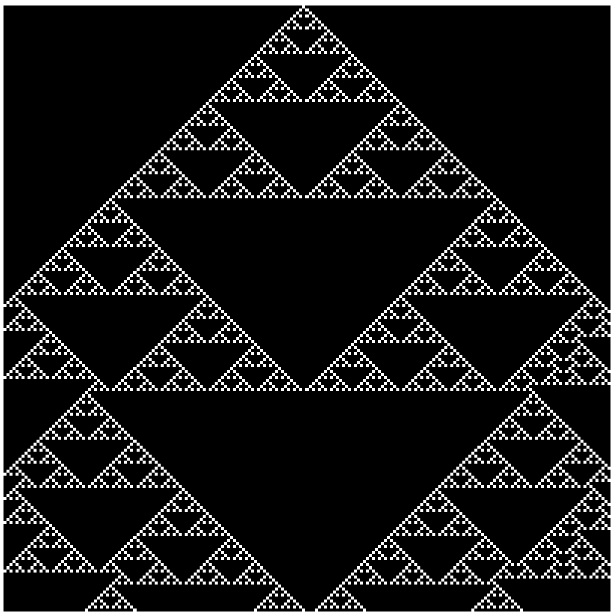
- 「REDUCE ユーザーズマニュアル（日本語版）」A.C. ハーン著, マグロウヒルブック株式会社, 1988.
- 「利用の手引・MSP コマンド編」九州大学大型計算機センター・ライブラリ室, 1993.
- 「UXP/M for the Working Scientists」佐藤 周行, 1993.

4

数値計算サブルーチン I

数値計算パッケージとして以下のライブラリを提供している．各ライブラリともサブルーチン群であり． SSL2/VP, NUMPAC/VP は，VP の性能を引き出すように特別にチューニングされている．

パッケージ名	機能	MSP	UXP
BLAS	線形代数用サブルーチン群		○
EISPACK	固有値関連サブルーチン群		○
LINPACK	線形代数用サブルーチン群		○
NUMPAC	名古屋大学提供総合数値計算パッケージ	○	○
NUMPAC/VP	NUMPAC の VP 版	○	○
SSL II	富士通株式会社提供	○	○
SSL II/VP	SSL II の VP 版	○	○



One dimensional Cellular Automata part 1

4.1 BLAS

【概略】
BLAS は線形代数用のサブルーチン群．次の分類によって構成される．

- ベクトル × ベクトルのサブルーチン群 (Level 1)
  setup Givens rotation, swap x into y, sum of absolute values, etc.
- 配列 × ベクトルのサブルーチン群 (Level 2)
  Matrix vector multiply, Rank one(two) update to a matrix, etc.
- 配列 × 配列のサブルーチン群 (Level 3)
  Solving triangular matrix with many right-hand sides, etc.

【処理環境】

- UXP/M1800

【使用例】

- BLAS Level1 ルーチンの組み込み．

```
kyu-cc% fort a.f -lblas ↵
```

- BLAS Level2 ルーチンの組み込み．

```
kyu-cc% fort a.f -lblas2 ↵
```

- BLAS Level3 ルーチンの組み込み．

```
kyu-cc% fort a.f -lblas3 ↵
```

【提供】

- Dongarra Group

【参考文献】

- 「UXP/M for the Working Scientists」佐藤 周行, 1993.

4.2 EISPACK

【概略】
Eispack は J.Dongarra のグループが開発した固有値関連のルーチン群．倍長精度対応の Eispack が使用可能．

【処理環境】

- UXP/M1800

【使用例】

- EISPACK ルーチンを組み込む．

```
kyu-cc% fort a.f -leispack ⏎
```

【提供】

- Dongarra Group

【参考文献】

- 「UXP/M for the Working Scientists」佐藤 周行, 1993.

4.3　LINPACK

【概略】
LINPACK は Jack Dongarra, Jim Bunch, Cleve Moler and Pete Stewart のグループが作成した線形代数に関するサブルーチンパッケージ．

【処理環境】

- UXP/M1800

【使用例】

- `LINPACK` のルーチンを使った `Fortran` プログラム `a.f` で実行ファイルを作る．

```
kyu-cc% fort a.f -llinpack ↵
```

【提供】

- Dongarra Group

【参考文献】

- 「UXP/M for the Working Scientists」佐藤 周行, 1993.

4.4 NUMPAC, NUMPAC/VP

【概略】
NUMPAC(Nagoya University Mathematical PACkage) は名古屋大学大型計算機センター研究開発部及び名古屋大学工学部数値解析研究グループを主として開発された数値計算パッケージ．線形代数から特殊関数に至る広汎な領域を包含する．精度，速度とも高い性能を有する．NUMPAC/VP は VP 用の NUMPAC．

【処理環境】

- MSP/M1800, MSP/VP2600
- UXP/M1800, UXP/VP2600

【使用例】

- MSP の TSS で NUMPAC を組み込んだ Fortran プログラムを実行する．
  VP 版は'LIB.NUMVP.LOAD' であるが，TSS での実行は出来ない．

```
 READY
FORT TEST.FORT LIB('LIB.NUMPAC.LOAD') ↵
```

- UXP で NUMPAC を組み込んだ Fortran プログラムより実行ファイルを作成する．

```
kyu-cc% fort a.f -lnumpac ↵
```

【提供】

- 名古屋大学大型計算機センター

【サブルーチンの機能】
各サブルーチン名は目的別にまとめ，代表する副プログラム名のみを記述している．単精度/倍精度等の型の区別や，具体的な使用方法は参考文献を参照されたい．参考文献は MSP の MANUAL コマンドで出力可能．

基本行列演算

ADDMMV	行列の加減算
MDETS	行列式の計算
MNORMS	行列の正規化
MNRMBS	バンド行列の正規化
MNRSPS	対称対称行列の正規化
MULMMV	行列の乗算
MULMVV	行列とベクトルの乗算

連立一次方程式

BUNCBS	Bunch 法による対称バンド行列係数連立一次方程式の解法
BUNCHS	Bunch 法による対称行列係数連立一次方程式の解法
CGHTCS	共役勾配法による対称正値連立一次方程式の解法（圧縮モード）
CHLBDC	Cholesky 法, 改訂 Cholesky 法による Hermite 対称正値連立一次方程式の解法 (バンド行列)
CHLVBS	Cholesky 法による対称正値連立一次方程式の解法（可変帯幅バンド行列，圧縮表現）
CHOLCS	Cholesky 法, 改訂 Cholesky 法による対称正値連立一次方程式の解法 (密行列, 圧縮表現)
CHOLFC	Cholesky 法, 改訂 Cholesky 法による Hermite 対称正値連立一次方程式の解法（密行列）
CHOLFS	Cholesky 法, 改訂 Cholesky 法による対称正値連立一次方程式の解法（密行列）
CHOLSK	Cholesky 法による対称正値連立一次方程式の解法
GAUELS	LU 分解による連立一次方程式の解法
GSORSS	SOR 分解による疎行列の連立一次方程式の解法（圧縮表現）
LAPLBS	二次元 Laplace 方程式の解法
LEQBDS	Gauss の消去法によるバンド行列係数連立一次方程式の解法
LEQLSS	Householder 変換による一般連立一次方程式の最小二乗解および最小ノルム解
LEQLUS	LU 分解による連立一次方程式解法
LSMNS	特異値分解による一般連立一次方程式の解法最小二乗最小ノルム解
PRCGFS	前処理付き共役勾配法による対称正値連立一次方程式の解法
PRCGSS	前処理付き共役勾配法による対称正値疎行列の連立一次方程式の解法（圧縮表現）
TRDSPS	対称正値三項方程式の解法
TRIDGS	三項方程式の解法

行列の逆転

GINVS	特異値分解による一般化逆行列
MINVS	行列の逆転
MINVSP	対称正値行列の逆転

代数方程式，非線形方程式

BROYDS	Broyden の方法による非線形連立一次方程式の解法
FLPOWS	Davidon-Fletcher-Powell 法による関数の最小化
GJMNKS	Garside-Jarrat-Mack 法による実係数代数方程式の解法
MINSXS	Simplex 法による関数の最小化
NOLEQS	非線形方程式の解法
NOLLS1	Quasi-Newton 法による非線形最小二乗法サブルーチン
POLEQC	複素係数代数方程式の解法
POLESB	静電場モデルによる複素係数代数方程式の解法
QUADRC	複素係数低次数方程式の解法
QUADRS	実係数低次代数方程式の解法
RTFNDS	非線形方程式の解法

線形計画法

LIPS	Criss-Cross 法を用いた線形計画問題解法ルーチン
SIMPLX	Simplex 法を用いた線形計画法

数列と級数の加速

ACCELS	数列または級数の収束の加速

固有値解析

CGHBSS	Householder-Bisection 法による $Ax = \lambda Bx$ 型の固有値解析（Hermite 行列）
CGHQIS	Householder-QR-Inverse Iteration 法による $Ax = \lambda Bx$ 型の固有値解析（Hermite 行列）
CGHQRS	Householder-QR 法による $Ax = \lambda Bx$ 型の固有値解析（Hermite 行列）
CGKLZS	LZ 法による $Ax = \lambda Bx$ 型の固有値解析（複素行列）
CHEQIS	QR 法および逆反復法による複素行列の固有値解析
CHEQRS	QR 法による複素行列の固有値解析
CHOBSS	Householder-Bisection 法による Hermite 行列の固有値解析
CHOQRS	Householder-QR 法による Hermite 行列の固有値解析
CHQRIS	Householder-QR- 逆反復法による Hermite 行列の固有値解析
GHBSVS	Householder-Bisection 法による $Ax = \lambda Bx$ 型の固有値解析
GHQRIS	Householder-QR- 逆反復法による $Ax = \lambda Bx$ 型の固有値解析
GHQRVS	Householder-QR 法による $Ax = \lambda Bx$ 型の固有値解析
HEQRVS	Double QR 法による実非対称行列の固有値解析
HOBSVS	Householder-Bisection 法による実対称行列の固有値解析
HOQRVS	Householder-QR 法による実対称行列の固有値解析
HQRIIS	Householder-QR- 逆反復法による実対称行列の固有値解析
JACOBS	Threshold Jacobi 法による実対称行列の固有値解析
JENNFS	Jennings の同時反復法による実対称行列の固有値解析
NGHOUS	準直接法による $Ax = \lambda Bx$ 型の固有値解析
NGJENS	Jennings 法による $Ax = \lambda Bx$ 型の固有値解析
NSHOUS	準直接法による $Ax = \lambda x$ 型の固有値解析
NSJENS	Jennings 法による $Ax = \lambda x$ 型の固有値解析
RHBSVS	Rutishauser-Bisection 法による対称バンド行列の固有値解析
RHQRVS	Rutishauser-QR 法による実対称バンド行列の固有値解析
SVDS	特異値分解

補間，平滑化

AGFBS	Briggs の方法による不規則分布データの格子化
CFS1A	Spline による曲線のあてはめ
CFS2A	Spline による曲面のあてはめ
DCOMD1	複合多項式による曲線のあてはめ
DSCI1A	１変数スプライン補間
DSCI1D	２変数スプライン補間
HERM31	区分的 Hermite 補間による曲線のあてはめ
HERM32	区分的 Hermite 補間による曲面のあてはめ
LSAICS	直交多項式による最小二乗近似
LSANLS	非線形最小二乗法による曲線のあてはめ
TETPCK	不規則分布３変数関数データに対する C^k 級補間法 $(0 \leq k \leq 1)$
TRIPCK	不規則分布２変数関数データに対する C^k 級補間法 $(0 \leq k \leq 3)$

表関数

BERNO	Bernoulli 数
BETNO	β 数
EULNO	Euler 数
FCTRL	階数の計算
GAMCO	$1/\Gamma(x)$ の Taylor 級数展開係数
HARMS	$\phi(n) = \sum_{k=1}^{n}(1/k)$
ZETNO	$\zeta(n) = \sum_{k=1}^{\infty}(1/k^n)$

フーリエ解析

BITREV	ビット逆転によるデータの並べ替え
DRCH1S	Chebyshev 級数の導関数，不定積分
FCHB1S	Chebyshev 多項式による関数の Fourier 展開
FCOSCS	開区間 $(0,\pi)$ で与えられた関数の級数展開
FCOSMS	中点公式にもとづく高速 cosine, sine 変換
FCOSTS	台形公式にもとづく高速 cosine, sine 変換
FFT2DC	複素高速 Fourier 変換 (2, 3 次元)
FFT2DR	実高速 Fourier 解析，合成 (2, 3 次元)
FFTC	複素高速 Fourier 解析
FFTR	実高速 Fourier 解析
FFTRI	実高速 Fourier 合成
FFTS	複素高速 Fourier 変換
FT235C	サンプル数が $2^K3^L5^M$ の形の高速 Fourier 解析
TRIGQP	２進逆順に並べられた三角関数表
VCHB1S	Chebyshev 級数
VCHB2S	第 2 種 Chebyshev 級数
VCOSS	cosine 級数, sine 級数

数値微積分

AQCHYS	Caucy 主値積分に対する自動積分
AQCOSS	振動する関数の半無限自動積分
AQCPACK	複素数値関数の自動数値積分
AQDCCS	Clenshaw-Curtis 法による自動積分
AQIOSC	複素数値関数の無限振動積分の自動積分
AQIOSS	半無限区間振動積分の自動積分
AQMDS	等差数列的に標本数を増す補間型積分にもとづく自動多重積分
AQNDS	自動多重数値積分
AQNN5S	Newton-Cotes 5,7,9 点則にもとづく適応型自動積分
AQOSCS	有限 Fourier 積分
DEFINS	二重指数関数型公式による有限区間積分
GASNS	Gauss 型数値積分
HINFAS	二重指数関数型公式による半無限区間積分
INFINS	二重指数関数型公式による無限区間積分
MQFSRS	完全対称則による多重数値積分
MQNCDS	Newton-Cotes 則の直積による多重数値積分
MQPRRS	直積型公式による多重数値積分
QDAPBS	等差数列的標本点を増す補間型積分法
ROMBGS	Romberg 積分
TNCOTS	数値積分公式のための重率と分点の値
TRAPZS	台形則による無限区間積分

常微分方程式

ODEBSS	有理補外法による連立 1 階常微分方程式の初期値問題の解法
RK4S	４次古典的 Runge-Kutta 法による連立 1 階常微分方程式の初期値問題の解法
RKF4AS	Runge-Kutta-Fehlberg 4 次法による連立 1 階常微分方程式の初期値問題の解法
RKM4AS	Runge-Kutta-Merluzzi 4 次法による連立 1 階常微分方程式の初期値問題の解法

初等関数

ALANGV	Langevin 関数
ALOG1	$\log(1+x)$
ASINH	逆双曲線関数
CABS1	$\|\|z\|\|_1 = \|x\| + \|y\|, \quad z = x + iy$
COMB	二項係数
EXP1	$e^x - 1$
FASTEE	e の高速高精度計算
FASTPI	π の高速高精度計算
SINHP	引数 $\frac{\pi}{2}x$ に対する三角関数

特殊関数

ABRM0	Abramowitz 関数（0 ～ 2 次）
ABRMW	整数次 Abramowitz 関数
ACND	累積正規分布関数と余関数の逆関数
AERF	誤差関数および余関数の逆関数
AICGAM	不完全ガンマ関数
BETIC	不完全ベータ積分
BLAS	Blasius 方程式の解と導関数
CELI1	第 1, 2 種完全楕円積分
CGAMMA	複素変数のガンマ関数
CLASN	Clausen の積分
CND	累積正規分布関数と余関数
DAWSN	Dawson の積分
DEBYE	Debye の関数
DIGAM	Digamma 関数
DILOG	Dilogarithm
ERFC1	余誤差関数の積分
EXI	指数積分
FRESS	Fresnel 正弦，余弦積分
HYPGM	超幾何級数と合流型超幾何級数
ICEILS	第 1，2 種不完全楕円積分
JACELS	Jacobian 楕円関数
PN	ルジャンドル多項式およびルジャンドル陪多項式
QN	第 2 種ルジャンドル関数およびルジャンドル陪関数
QNOME	楕円 θ 関数の Nome
RGAMA	ガンマ関数の逆数
SI	正弦積分，余弦積分
SPENC	Spence 関数
TMFRM	Thomas-Fermi 方程式の解とその導関数
ZETA	Riemann Zeta 関数

直交多項式

PLEGE	ルジャンドル多項式 $P_n(X)$
PLEGA	ルジャンドル陪関数 $P_n^m(X)$
PLEGN	規格化ルジャンドル陪関数 $\bar{P_n^m}(X)$
PCHB1	第 1 種チェビシェフ多項式 $T_n(X)$
PCHB2	第 2 種チェビシェフ多項式 $U_n(X)$
PLAGU	ラゲール多項式 $L_n(X)$
PLAGG	一般ラゲール多項式 $L_n^{(a)}(X)$
PHERM	エルミート多項式 $H_n(X)$

ベッセル関数

AI	Airy 関数と導関数
BER0	Kelvin 関数 (0, 1 次)
BESJFC	複素変数の非整数次第 1 種 Bessel 関数
BESJNC	複素変数の整数次第 1 種 Bessel 関数
BESKNC	複素変数の整数次第 2 種変形 Bessel 関数
BESYNC	複素変数の整数次第 2 種 Bessel 関数
BH0	Struve 関数 (0, 1 次)
BI0	変形 Bessel 関数 (0, 1 次)
BI0I0	変形 Bessel 関数の積分
BI0ML0	変形 Bessel 関数と変形 Struve 関数の差 (0, 1 次)
BIF	非整数次第 1 種変形 Bessel 関数
BIN	整数次変形 Bessel 関数
BJ0	Bessel 関数 (0, 1 次)
BJ0I0	Bessel 関数の積分
BJF	非整数次第 1 種 Bessel 関数
BJN	整数次 Bessel 関数
BKF	非整数次第 2 種変形 Bessel 関数
BL0	変形 Struve 関数 (0, 1 次)
BYF	非整数次第 2 種 Bessel 関数
J0Y0S	Bessel 関数 (0, 1 次)
SI0	変形球 Bessel 関数 (0, 1 次)
SIK	整数次変形球 Bessel 関数
SJ0	球 Bessel 関数 (0, 1 次)
SJN	整数次球 Bessel 関数
ZBJ0	$J_0 \sim J_{15}$ の正の零点
ZBJ0S	Bessel 関数 J_0, J_1 の零点および導関数
ZBJN	$J_0 \sim J_{15}$ の正の零点

その他

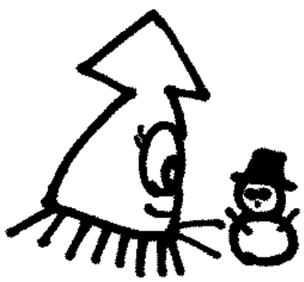
SETPACK	集合演算プログラムパッケージ
SORTPACK	スカラー，ベクトルデータの内部ソーティング
BITLOGIC	4-Byte データ間のビットごとの論理演算
IBITCT	4-Byte データの 2 進表示での 1 のビット数の数え上げ
IBITRV	4-Byte データのビットパターンの逆順並べ換え
IGCD	二つの整数の最大公約数
PRIME	素数表の作成
PRMFAC	整数の素因数分解
RANDOM	一様乱数の生成
ROUND	実数の 0 捨 1 入

【注意事項】

- NUMPAC を使用して得られた結果を公表する場合は，NUMPAC 及び使用プログラム名について言及すること．

【参考文献】

- 「ライブラリ・プログラム利用の手引（数値計算編：NUMPAC ）」名古屋大学大型計算機センター, 1989．
- 「UXP/M for the Working Scientists」佐藤 周行, 1993.



4.5　SSL II, SSL II/VP

【概略】
SSL II(Scientific Subroutine Library II) は富士通株式会社提供の数値計算サブルーチンライブラリ．アルゴリズムの面では精度的に安定な手法を採用し，プログラミングの面では速度向上に重点がおかれている．使用言語は Fortran を用い，コンパイラの最適化機能を最大限に受けられるようコーティングされている．マニュアルには各サブルーチンの手法概要が記され，例題も詳しい．富士通の計算機マニュアルとしては出色の出来である．MSP の SSL II は LINK 時に自動的に組み込まれるので，利用者はプログラム内でサブルーチンを CALL するだけでよい．SSL II/VP は SSL II の VP 版．

【処理環境】

- MSP/M1800, MSP/VP2600
- UXP/M1800, MSP/VP2600

【使用例】

- `UXP` で `SSL II` ルーチンの組み込みをおこなう．`MSP` では自動的に組み込まれる．

```
kyu-cc% fort a.f -lssl2 ↵
```

【提供】

- 富士通株式会社

【サブルーチンの機能】
各サブルーチンは一部を除き，単精度と倍精度がある．4 倍精度はサポートしていない．倍精度は単精度サブルーチンの頭に`'D'` がつく．具体的な使用方法はマニュアルを参照されたい．

行列格納モードの変換

CGSM	行列格納モードの変換（一般モード → 対称行列）
CSGM	行列格納モードの変換（対称行列 → 一般モード）
CGSBM	行列格納モードの変換（一般モード → 対称バンド行列）
CSBGM	行列格納モードの変換（対称バンド行列 → 一般モード）
CSSBM	行列格納モードの変換（対称行列 → 対称バンド行列）
CSBSM	行列格納モードの変換（対称バンド行列 → 対称行列）
VMGGM	行列の積（実行列）［拡張機能］

行列操作

AGGM	行列の和（実行列）
SGGM	行列の差（実行列）
MGGM	行列の積（実行列）
MGSM	行列の積（実行列・実対称行列）
ASSM	行列の和（実対称行列）
SSSM	行列の差（実対称行列）
MSSM	行列の積（実対称行列）
MSGM	行列の積（実対称行列・実行列）
MAV	実行列と実ベクトルの積
MCV	複素行列と複素ベクトルの積
MSV	実対称行列と実ベクトルの積
MSBV	実対称バンド行列と実ベクトルの積
MBV	実バンド行列と実ベクトルの積

連立一次方程式

LAX	実行列の連立一次方程式（クラウト法）
LCX	複素行列の連立一次方程式（クラウト法）
LSX	正値対称行列の連立一次方程式（変形コレスキー法）
LSIX	実対称行列の連立一次方程式（ブロック対角ピボッティング手法）
LSBX	正値対称バンド行列の連立一次方程式（変形コレスキー法）
LSBIX	実対称バンド行列の連立一次方程式（ブロック対角ピボッティング手法）
LBX1	実バンド行列の連立一次方程式（ガウス消去法）
LSTX	正値対称 3 項行列の連立一次方程式（変形コレスキー法）
LTX	実 3 項行列の連立一次方程式（ガウス消去法）
LAXR	実行列の連立一次方程式の解の反復改良
LCXR	複素行列の連立一次方程式の解の反復改良
LSXR	正値対称行列の連立一次方程式の解の反復改良
LSIXR	実対称行列の連立一次方程式の解の反復改良
LSBXR	正値対称バンド行列の連立一次方程式の解の反復改良
LBX1R	実バンド行列の連立一次方程式の解の反復改良
VLSX	正値対称行列の連立一次方程式（変形コレスキー法）[拡張機能]
VLTX	実 3 項行列の連立一次方程式（サイクリック・リダクション法）[拡張機能]
VLTX1	定数型実 3 項行列の連立一次方程式 [拡張機能]
VLTX2	定数型実 3 項行列の連立一次方程式 [拡張機能]
VLTX3	定数型実 3 項行列の連立一次方程式 [拡張機能]
VLAX	実行列の連立一次方程式（ブロッキング LU 分解法）[拡張機能]

逆行列

LUIV	LU 分解された実行列の逆行列
CLUIV	LU 分解された複素行列の逆行列
LDIV	LDL^T 分解された正値対称行列の逆行列
VLUIV	LU 分解された実行列の逆行列 [拡張機能]

行列の三角分解

ALU	実行列の LU 分解（クラウト法）
CLU	複素行列の LU 分解（クラウト法）
SLDL	正値対称行列の LDL^T 分解（変形コレスキー法）
SMDM	実対称行列の MDM^T 分解（ブロック対角ピボッティング手法）
SBDL	正値対称バンド行列の LDL^T 分解（変形コレスキー法）
SBMDM	実対称バンド行列の MDM^T 分解（ブロック対角ピボッティング手法）
BLU1	実バンド行列の LU 分解（ガウス消去法）
VSLDL	正値対称行列の LDL^T 分解 [拡張機能]
VALU	実行列の LU 分解（ブロッキング LU 分解法）[拡張機能]

三角分解された連立一次方程式

LUX	LU 分解された実行列の連立一次方程式
CLUX	LU 分解された複素行列の連立一次方程式
LDLX	LDL^T 分解された正値対称行列の連立一次方程式
MDMX	MDM^T 分解された実対称行列の連立一次方程式
BDLX	LDL^T 分解された正値対称バンド行列の連立一次方程式
BMDMX	MDM^T 分解された実対称バンド行列の連立一次方程式
BLUX1	LU 分解された実バンド行列の連立一次方程式
VLDLX	LDL^T 分解された正値対称行列の連立一次方程式 [拡張機能]

最小二乗解

LAXL	実行列の最小二乗解（ハウスホルダー変換）
LAXLR	実行列の最小二乗解の反復改良
LAXLM	実行列の最小二乗最小ノルム解（特異値分解法）
GINV	実行列の一般逆行列（特異値分解法）
ASVD1	実行列の特異値分解（ハウスホルダー法，QR 法）

固有値，固有ベクトル

EIG1	実行列の固有値及び固有ベクトル（２段 QR 法）
CEIG2	複素行列の固有値及び固有ベクトル（QR 法）
SEIG1	実対称行列の固有値及び固有ベクトル（QL 法）
SEIG2	実対称行列の固有値及び固有ベクトル（バイセクション法, 逆反復法）
HEIG2	エルミート行列の固有値及び固有ベクトル（バイセクション法，逆反復法）
BSEG	実対称バンド行列の固有値及び固有ベクトル
BSEGJ	実対称バンド行列の固有値及び固有ベクトル（ジェニングス法）
TEIG1	実対称３重対角行列の固有値及び固有ベクトル（QL 法）
TEIG2	実対称３重対角行列の固有値及び固有ベクトル（バイセクション法, 逆反復法）
GSEG2	実対称行列の一般固有値及び固有ベクトル（バイセクション法，逆反復法）
GBSEG	実対称バンド行列の一般固有値及び固有ベクトル（ジェニングス法）
VSEG2	実対称行列の固有値・固有ベクトル [拡張機能]
VGSG2	実対称行列の一般固有値・固有ベクトル [拡張機能]

固有値

HSQR	実ヘッセンベルグ行列の固有値（２段 QR 法）
CHSQR	複素ヘッセンベルグ行列の固有値（QR 法）
TRQL	実対称３重対角行列の固有値（QL 法）
BSCT1	実対称３重対角行列の固有値（バイセクション法）

固有ベクトル

HVEC	実ヘッセンベルグ行列の固有ベクトル（逆反復法）
CHVEC	複素ヘッセンベルグ行列の固有ベクトル（逆反復法）
BSVEC	実対称バンド行列の固有ベクトル（逆反復法）

その他の固有値計算

BLNC	実行列の平衡化
CBLNC	複素行列の平衡化
HES1	実行列の実ヘッセンベルグ行列への変換（ハウスホルダー法）
CHES2	複素行列の複素ヘッセンベルグ行列への変換（安定化基本相似変換）
TRID1	実対称行列の実対称３重対角行列への変換（ハウスホルダー法）
TRIDH	エルミート行列の実対称３重対角行列への変換（ハウスホルダー法）
BTRID	実対称バンド行列の実対称３重対角行列への変換
HBK1	実行列の固有ベクトルへの逆変換と正規化
CHBK2	複素行列の固有ベクトルへの逆変換
TRBK	実対称行列の固有ベクトルへの逆変換
TRBKH	エルミート行列の固有ベクトルへの逆変換
NRML	実行列の固有ベクトルの正規化
CNRML	複素行列の固有ベクトルの正規化
GSCHL	一般形から標準形への変換（実対称行列の一般固有値問題）
GSBK	一般形の固有ベクトルへの逆変換（実対称行列の一般固有値問題）

非線形計算

RQDR	実係数２次方程式
CQDR	複素係数２次方程式
LOWP	実係数低次代数方程式（５次以下）
RJETR	実係数高次代数方程式（ジェンキンス・トラウプの方法）
CJART	複素係数高次代数方程式（ヤラット法）
TSD1	実超越方程式 $f(x) = 0$ （ブレント法）
TSDM	実超越方程式 $f(x) = 0$ （マラー法）
CTSDM	複素超越方程式 $f(z) = 0$ （マラー法）
NOLBR	連立非線形方程式（ブレント法）

極値問題

LMINF	１変数関数の極小化（微係数不要，２次補間法）
LMING	１変数関数の極小化（微係数要，３次補間法）
MINF1	多変数関数の極小化（微係数不要，改訂準ニュートン法）
MING1	多変数関数の極小化（微係数要，準ニュートン法）
NOLF1	関数二乗和の極小化（微係数不要，改訂マルカート法）
NOLG1	関数二乗和の極小化（微係数要，改訂マルカート法）
LPRS1	線形計画問題（改訂シンプレックス法）
NLPG1	非線形計画問題（微係数要，パウエル法）

補間

AKLAG	エイトケン・ラグランジュ補間
AKHER	エイトケン・エルミート補間
SPLV	３次 spline 補間式による補間, 数値微分
BIF1	B-spline 補間式 (I) による補間，数値微分，数値積分
BIF2	B-spline 補間式 (II) による補間，数値微分，数値積分
BIF3	B-spline 補間式 (III) による補間，数値微分，数値積分
BIF4	B-spline 補間式 (IV) による補間，数値微分，数値積分
BIFD1	B-spline ２次元補間式 (I-I) による補間，数値微分，数値積分
BIFD3	B-spline ２次元補間式 (III-III) による補間，数値微分，数値積分
AKMID	２次元準エルミート補間式による補間
INSPL	３次 spline 補間式
AKMIN	準エルミート補間式
BIC1	B-spline 補間式 (I)
BIC2	B-spline 補間式 (II)
BIC3	B-spline 補間式 (III)
BIC4	B-spline 補間式 (IV)
BICD1	B-spline ２次元補間式 (I-I)
BICD3	B-spline ２次元補間式 (III-III)

近似

LESQ1	最小二乗近似多項式

平滑化

SMLE1	最小二乗近似多項式による平滑化（等間隔離散点）
SMLE2	最小二乗近似多項式による平滑化（不等間隔離散点）
BSF1	B-spline 平滑化式による平滑化，数値微分，数値積分
BSC1	B-spline 平滑化式（固定節点）
BSC2	B-spline 平滑化式（節点追加方式）
BSFD1	B-spline ２次元平滑化式による平滑化，数値微分，数値積分
BSCD2	B-spline ２次元平滑化式（節点追加方式）

級数

FCOSF	偶関数の cosine 級数展開（関数入力，高速 cosine 変換）
ECOSP	cosine 級数の求和
FSINF	奇関数の sine 級数展開（関数入力，高速 sine 変換）
ESINP	sine 級数の求和
FCHEB	実関数のチェビシェフ級数展開（関数入力，高速 cosine 変換）
ECHEB	チェビシェフ級数の求和
GCHEB	チェビシェフ級数の導関数，数値微分
ICHEB	チェビシェフ級数の不定積分

変換

FCOST	離散型 cosine 変換（台形公式，２基底 FFT）
FCOSM	離散型 cosine 変換（中点公式，２基底 FFT）
FSINT	離散型 sine 変換（台形公式，２基底 FFT）
FSINM	離散型 sine 変換（中点公式，２基底 FFT）
RFT	離散型実フーリエ変換
CFTM	多次元離散型複素フーリエ変換（混合基底 FFT）
CFT	多次元離散型複素フーリエ変換（８，２基底 FFT）
CFTN	離散型複素フーリエ変換（８，２基底 FFT，逆順出力）
CFTR	離散型複素フーリエ変換（８，２基底 FFT，逆順入力）
PNR	ビット逆転によるデータの置換
LAPS1	ラプラス変換（複素右半平面で正則な有理関数）
LAPS2	ラプラス変換（一般の有理関数）
LAPS3	ラプラス変換（一般関数）
HRWIZ	Hurwitz 多項式の判定
VCOS1	離散型 cosine 変換（２基底 FFT）［拡張機能］
VSIN1	離散型 sine 変換（２基底 FFT）［拡張機能］
VRFT1	離散型実フーリエ変換（性能優先型，２基底 FFT）［拡張機能］
VRFT2	離散型実フーリエ変換（メモリー節約型，２基底 FFT）［拡張機能］
VCFT1	離散型複素フーリエ変換（性能優先型，２基底 FFT）［拡張機能］
VCFT2	離散型複素フーリエ変換（メモリー節約型，２基底 FFT）［拡張機能］

数値微分

SPLV	３次 spline 補間式による補間, 数値微分
BIF1	B-spline 補間式 (I) による補間，数値微分，数値積分
BIF2	B-spline 補間式 (II) による補間，数値微分，数値積分
BIF3	B-spline 補間式 (III) による補間，数値微分，数値積分
BSF1	B-spline 平滑化式による平滑化，数値微分，数値積分
BIFD1	B-spline ２次元補間式 (I-I) による補間，数値微分，数値積分
BIFD3	B-spline ２次元補間式 (III-III) による補間，数値微分，数値積分
BSFD1	B-spline ２次元平滑化式による平滑化，数値微分，数値積分
GCHEB	チェビシェフ級数の導関数，数値微分

数値積分

SIMP1	１次元有限区間積分（等間隔離散点入力，シンプソン則）
TRAP	１次元有限区間積分（不等間隔離散点入力，台形則）
BIF1	B-spline 補間式 (I) による補間，数値微分，数値積分
BIF2	B-spline 補間式 (II) による補間，数値微分，数値積分
BIF3	B-spline 補間式 (III) による補間，数値微分，数値積分
BSF1	B-spline 平滑化式による平滑化，数値微分，数値積分
BIFD1	B-spline ２次元補間式 (I-I) による補間，数値微分，数値積分
BIFD3	B-spline ２次元補間式 (III-III) による補間，数値微分，数値積分
BSFD1	B-spline ２次元平滑化式による平滑化，数値微分，数値積分
SIMP2	１次元有限区間積分（関数入力，適応型シンプソン則）
AQN9	１次元有限区間積分（関数入力，適応型ニュートン・コーツ９点則）
AQC8	１次元有限区間積分（関数入力，クレンショー・カーチス型積分法）
AQE	１次元有限区間積分（関数入力，二重指数関数型積分公式）
AQEH	１次元半無限区間積分（関数入力，二重指数関数型積分公式）
AQEI	１次元全無限区間積分（関数入力，二重指数関数型積分公式）
AQMC8	多次元有限領域積分（関数入力，クレンショー・カーチス型積分法）
AQME	多次元積分（関数入力，二重指数関数型積分公式）

微分方程式

RKG	連立１階常微分方程式（ルンゲ・クッタ・ギル法）
HAMNG	連立１階常微分方程式（ハミング法）
ODRK1	連立１階常微分方程式（ルンゲ・クッタ・ヴァーナー法）
ODAM	連立１階常微分方程式（アダムス法）
ODGE	スティフ連立１階常微分方程式（ギア法）

特殊関数

CELI	第 1 種完全楕円積分 $K(x)$
CELI2	第 2 種完全楕円積分 $E(x)$
EXPI	指数積分 $E_i(x), \bar{E}_i(x)$
SINI	正弦積分 $S_i(x)$
COSI	余弦積分 $C_i(x)$
SFRI	正弦フレネル積分 $S(x)$
CFRI	余弦フレネル積分 $C(x)$
IGAM1	第 1 種不完全ガンマ関数 $\gamma(\nu, x)$
IGAM2	第 2 種不完全ガンマ関数 $\Gamma(\nu, x)$
IERF	逆誤差関数 $erf^{-1}(x)$
IERFC	逆余誤差関数 $erfc^{-1}(x)$
BJ0	第 1 種 0 次ベッセル関数 $J_0(x)$
BJ1	第 1 種 1 次ベッセル関数 $J_1(x)$
BY0	第 2 種 0 次ベッセル関数 $Y_0(x)$
BY1	第 2 種 1 次ベッセル関数 $Y_1(x)$
BI0	第 1 種 0 次変形ベッセル関数 $I_0(x)$
BI1	第 1 種 1 次変形ベッセル関数 $I_1(x)$
BK0	第 2 種 0 次変形ベッセル関数 $K_0(x)$
BK1	第 2 種 1 次変形ベッセル関数 $K_1(x)$
BJN	第 1 種整数次ベッセル関数 $J_n(x)$
BYN	第 2 種整数次ベッセル関数 $Y_n(x)$
BIN	第 1 種整数次変形ベッセル関数 $I_n(x)$
BKN	第 2 種整数次変形ベッセル関数 $K_n(x)$
CBIN	複素変数第 1 種整数次変形ベッセル関数 $I_n(z)$
CBKN	複素変数第 2 種整数次変形ベッセル関数 $K_n(z)$
CBJN	複素変数第 1 種整数次ベッセル関数 $J_n(z)$
CBYN	複素変数第 2 種整数次ベッセル関数 $Y_n(z)$
BJR	第 1 種実数次ベッセル関数 $J_\nu(x)$
BYR	第 2 種実数次ベッセル関数 $Y_\nu(x)$
BIR	第 1 種実数次変形ベッセル関数 $I_\nu(x)$
BKR	第 2 種実数次変形ベッセル関数 $K_\nu(x)$
CBJR	複素変数第 1 種実数次ベッセル関数 $J_\nu(z)$
NDF	正規分布関数 $\phi(x)$
NDFC	余正規分布関数 $\psi(x)$
INDF	逆正規分布関数 $\phi^{-1}(x)$
INDFC	逆余正規分布関数 $\psi^{-1}(x)$

疑似乱数

RANU2	一様乱数 $(0,1)$ の生成
RANU3	一様乱数 $(0,1)$ の生成（シャフル型）
RANN1	正規乱数の生成（高速型）
RANN2	正規乱数の生成
RANE2	指数乱数の生成
RANP2	ポアソン乱数の生成
RANB2	二項乱数の生成
RATF1	一様乱数 $(0,1)$ の頻度テスト
RATR1	一様乱数 $(0,1)$ の上昇・下降連テスト

【注意事項】

- SSL II のソースは拡張機能分を除き公開されている．MSP の LIBCOPY コマンドを用いて利用者のデータセットにコピーが可能であるが，九州大学大型計算機センター以外の使用は禁止されている．コピー方法は『利用の手引・ MSP コマンド編』参照．

【参考文献】

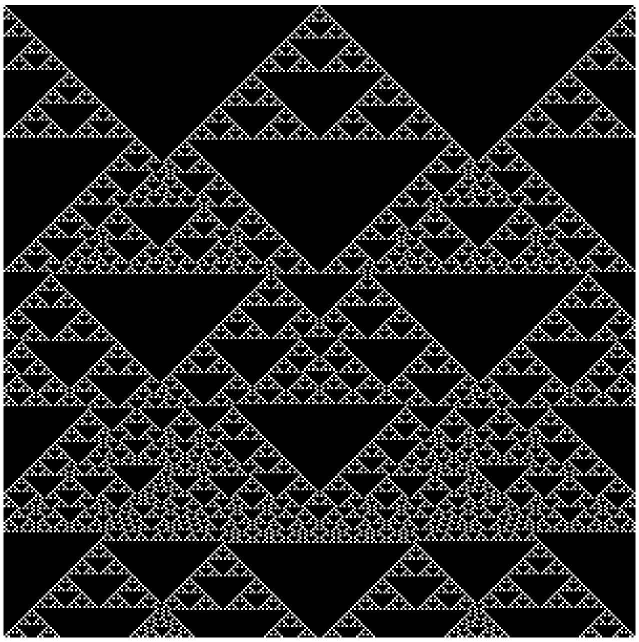
- 「SSL II 使用手引書（科学用サブルーチンライブラリ）」（99SP–0050）, 富士通株式会社．
- 「SSL II 拡張機能使用手引書（科学用サブルーチンライブラリ）」（99SP–4070）, 富士通株式会社．
- 「UXP/M for the Working Scientists」佐藤 周行, 1993.

5

数値計算サブルーチン II

九州大学大型計算機センターの開発課題として作成，登録しているライブラリの中で，数値計算ライブラリ関係の紹介と利用の手引をあげる．

ライブラリ名	機能	MSP	UXP
CGJQ	Gauss-Jacobi 積分公式の分点，重み算定	○	
CGLQ	Gauss-Laguerre 積分公式の分点，重み算定	○	
DIFF1S	解析関数の数値微分（単精度）	○	
DIFF1D	解析関数の数値微分（倍精度）	○	
MINMAX	線形方程式のミニマックス解	○	
POWELL	n 変数関数の極小値	○	
PRESNL	一般化されたフレネル積分	○	



One dimensional Cellular Automata part 2

5.1 CGJQ

Gauss-Jacobi 積分公式の係数 (Coefficients of Gauss-Jacobi Qauadrature Formula)

プログラム形式	サブルーチン副プログラム
作成者 作成年月日	塩川 浩三 (九州大学工学部) 1971 年 5 月 1 日
ソースの公表 使用言語	'QS.LIBSC(D1CGJQ)' (MSP) Fortran
使用 OS	MSP(M1800, VP2600)

【 概要 】

1. 目的
   $(1-x)^r(1+x)^s$ を重み関数とする Gauss-Jacobi 積分公式

$$\int_{-1}^{1}(1-x)^r(1+x)^s f(x)dx = 2^{r+s+1}B(r+1,s+1)\sum_{k=1}^{n}H_k f(a_k)$$

   の分点 a_k, 重み H_k を求める．$B(r,s)$ はベータ関数．

2. 計算法
   分点 a_k は n 次の Jacobi Polynomial $G_n^{r,s}(x)$ の零点であることを利用し，適当な近似値を Newton の方法により修正して求める．重み H_k は次式で与えられる．

$$H_k = \frac{4^{n-1}B(r+n,s+n)B(r+s+n,n)(r+n+2n-1)}{B(r+1,s+1)G_{n-1}^{r,s}(a_k)\frac{d}{dx}G_n^{r,s}(a_k)}$$

【 使用方法 】

1. 呼出し方法
   `CALL CGJQ(N,R,S,A,H,B,C,O,ILL)`

2. パラメータ

N	:	整数型定数または変数名．分点の数を与える．
R	:	倍精度実数型定数または変数名． 重み関数の $(1-x)$ のベキを与える．$R \geq -0.999$ とすること．
S	:	倍精度実数型定数または変数名． 重み関数の $(1+x)$ のベキを与える．$S \geq -0.999$ とすること．
A	:	倍精度実数型配列名．大きさ N 以上の一次元配列． 分点が値の大きい方から順にセットされて戻る．$A(K) > A(L) \quad (K < L)$.
H	:	倍精度実数型配列名．大きさ N 以上の一次元配列． 分点に対応する重みがセットされて戻る．
B, C	:	倍精度実数型配列名．作業領域として用いる大きさ N 以上の一次元配列．
O	:	実数型定数または変数名．分点の収束判定値を与える．

$O \leq 0.0$ であれば 10^{-16} として計算する．

ILL : 整数型変数名．サブルーチン CALL 時に $ILL < 0$ であれば分点の近似値が $A(K)$ に与えられているとして計算を行なう．
RETURN 時はエラー処理のパラメータ:
ILL=0 – 正しく処理されている
ILL>0 – 異常終了

3. 制限
$N \geq 1,\ R \geq -0.999,\ S \geq -0.999$ であれば計算を行うが，N があまり大きいとオーバーフローを起こす．たとえば，$N = 100, R = -0.998$，$S = 30$ でオーバーフローが起きた．

4. エラー処理
次の場合は ILL パラメータにそれぞれの値をセットして戻る．
   - $N \leq 0$ のとき $ILL = 1000$
   - $R < -0.999$ または $S < -0.999$ のとき $ILL = 2000$
   - a_k が 40 回反復修正しても収束しないとき $ILL = k$
   - $\sum_{k=1}^{n} A(K)$ が理論値 $\dfrac{N(S-R)}{2N+S+R}$ と異なるとき $ILL = 3000$

   いずれもエラーメッセージは出さない．

5. 使用ルーチン
DABS

【注意事項】

- R, S があまり大きいとき，N が小さい場合，また大きすぎる場合に正しい解が得られないことが多い（この場合，$ILL = 3000$ になることが多い）．次の場合，いずれも $ILL = 3000$ となった．
  $R = -0.998$，$S = 30$，$N = 50$
  $R = -0.998$，$S = 50$，$N = 2$
  $R = -0.9$，$S = 20$，$N = 50$
  $R = -0.9$，$S = 50$，$N = 2$

【参考文献】

- A.H.Stroud and D.Secrest, Gaussian Quadrature Formulas, Prentice-Hall，Inc．

5.2 CGLQ

Gauss-Laguerre 積分公式の係数 (Coefficients of Gauss-Laguerre Quadrature Formula)

プログラム形式	サブルーチン副プログラム
作成者 作成年月日	塩川 浩三 (九州大学工学部) 1971 年 5 月 1 日
ソースの公表 使用言語	'QS.LIBSC(D1CGLQ)' (MSP) Fortran
使用 OS	MSP(M1800, VP2600)

【 概要 】

1. 目的
   $x^s e^{-x}$ を重み関数とする Gauss-Laguerre 積分公式

$$\int_0^\infty x^s e^{-x} f(x)dx \approx \Gamma(S+1)\sum_{k=1}^{n} H_k f(a_k)$$

   の分点 a_k と重み H_k を求める．

2. 計算方法
   分点 a_k は n 次の Laguerre Polynomial $L_n^s(x)$ の零点であることを利用し，適当な近似値を Newton の方法により修正して求める．重み H_k は次式を用いて求める．

$$H_k = \frac{\Gamma(n+s+1)\ a_k}{n!\ \Gamma(s+1)\{(n+s)L_{n-1}^s(a_k)\}^2}$$

【 使用方法 】

1. 呼出し方法
   CALL CGLQ(N,S,A,H,O,ILL)

2. パラメータ

   N ：整数型定数または変数名．分点の数を与える．
   S ：倍精度実数型定数または変数名．重み関数のベキを与える．
   　　$S \geq -0.999$ とすること．
   A ：倍精度実数型配列名．大きさ N 以上の一次元配列．
   　　分点が値の小さいものから順にセットされて戻る． $A(K) < A(L) \quad (K < L)$.
   H ：倍精度実数型配列名．大きさ N 以上の一次元配列．
   　　分点に対応する重みがセットされて戻る．
   O 　実数型定数または変数名．分点の収束判定値を入れておく．
   　　$0 \leq 0.0$ であれば 10^{-16} として計算する．
   ILL：整数型変数名．サブルーチン CALL 時に $1 \leq ILL < N$ であれば
   　　分点 a_k $(k \geq ILL)$ の近似値が $A(K)$ に与えられているとする．
   　　ILL が負であれば全ての分点の近似値が与えられたとして計算する．
   　　RETURN 時はエラー処理のパラメータをセットして戻る．
   　　　ILL=0 – 正しく処理されている
   　　　ILL>0 – 異常終了 (エラー処理参照)

3. 制限
   $N \geq 1,\ R \geq -0.999$ であれば計算を行うが，N があまり大きいとオーバーフローを起こす．たとえば，$N = 100,\ S = -0.998$ でオーバーフローが起きた．

4. エラー処理
   次の場合は ILL パラメータにそれぞれの値をセットして戻る．
   - $N \leq 0$ のとき $ILL = 1000$
   - $S < -0.999$ のとき $ILL = 2000$
   - a_k が 40 回反復修正しても収束しないとき $ILL = k$
   - $\displaystyle\sum_{k=1}^{n} A(K)$ が理論値 $N(N+S)$ と異なるとき $ILL = 3000$

5. 使用ルーチン
   DABS

【注意事項】

- S が大きいときは，N が大きい場合または小さすぎる場合に正しい解が得られない．たとえば次の場合，$ILL = 3000$ となった．
  $S = 20.0$，$N = 2$，$N = 50$
  $S = 50.0$，$N = 5$，$N = 40$
- $\displaystyle\int_0^\infty x^\alpha e^{-x^2} f(x)dx = \frac{1}{2}\int_0^\infty t^{\frac{\alpha-1}{2}} e^{-t} f(t^{\frac{1}{2}})dt$ であるから，このサブルーチンを使って Gauss-Hermite 積分公式の分点，重みが計算できる．

【参考文献】

- A.H.Stroud and D.Secrest, Gaussian Quadrature Formulas, Prentice-Hall，Inc．

5.3 DIFF1S, DIFF1D

解析関数の数値微分 (Numerical Differentiation of Analytic Functions)

プログラム形式	サブルーチン副プログラム
作成者 作成年月日	角 秀吉 (久留米高専) 1973 年 3 月 10 日
ソースの公表 使用言語	'QS.LIBSC(D4DIFF1S)' (MSP) 'QS.LIBSC(D4DIFF1D)' (MSP) Fortran
使用 OS	MSP(M1800, VP2600)

【 概要 】

1. 目的
   解析関数 $f(z)$ の第 n 次導関数 $f^{(n)}(a)$ の計算．

2. 計算方法

$$f^{(n)} = \frac{n!}{2\pi i}\int_C \frac{f(z)}{(z-a)^{n+1}}dz = \frac{n!}{2\pi r^n}\int_0^{2\pi} f(a+re^{it})e^{-int}dt$$

   であるから，拡張された台形公式に適当な重みを掛け加えた方法による．すなわち微分する点 a を中心としていくつかの同心円周上に格子点をとる方法である．

【 使用方法 】

1. 呼出方法
   `CALL DIFF1S(F,NORDER,LCC,ZPOINT,RADIUS,ZVALUE,EPS)` (単精度)
   `CALL DIFF1D(F,NORDER,LCC,ZPOINT,RADIUS,ZVALUE,EPS)` (倍精度)

2. パラメータ

F	:	単精度複素数型関数名． 被微分関数 $f(z)$ を文関数または関数サブプログラムで与える． 関数サブプログラムで与えた場合，メインプログラムでこの サブプログラム名を EXTERNAL 宣言しておかなければならない．
NORDER	:	整数型変数名または整定数． 第 n 次導関数 $f^{(n)}(a)$ の階数 n を表す．
LCC	:	整数型変数名または整定数．微分する点 a を中心とする同心円の個数．
ZPOINT	:	複素数型変数名または複素定数．第 n 次導関数 $f^{(n)}(a)$ の点 a を表す．
RADIUS	:	実数型変数名または実定数．同心円の最大円の半径を表す．
ZVALUE	:	複素数型変数名．第 n 次導関数 $f^{(n)}(a)$ の値がセットされる．
EPS	:	実数型変数名または実数型定数．相対誤差の収束判定値を与える．

3. 制限事項

   (a) LCC の値は $1, 2, 3$ の 3 種類

(b) LCC=1 のとき $1 \leq \mathrm{NORDER} \leq 25$
LCC=2 のとき $1 \leq \mathrm{NORDER} \leq 24$
LCC=3 のとき $1 \leq \mathrm{NORDER} \leq 12$

(c) EPS>0

(d) RADIUS< $R \equiv$ 微分する点と特異点との最短距離

(e) $f(z)$ は解析関数に限る．

4. エラー処理

- LCC の値が $1, 2, 3$ でないとき，次のようなメッセージを出力し何も計算しない．

```
SUBROUTINE<DIFF1*>ILL CONDITION;0<LCC<4
```

- NORDER ≤ 0 であれば，次のようなメッセージを出力して何も計算しない．

```
SUBROUTINE<DIFF1*>ILL CONDITION;NORDER>0
```

- EPS ≤ 0 であれば次のような値が与えられたものとして計算する．

DIFF1S : EPS=10^{-8}
DIFF1D : EPS=10^{-18}

5. 使用ルーチン

DIFF1S : FLOAT, CMPLX, CEXP, CABS, MIN0
DIFF1D : DFLOAT, DCMPLX, CDEXP, CDABS, SNGL, MIN0

6. 精度
精度はパラメータ F, RADIUS, ZPOINT に関係しているが，大体の目安として RADIUS を適切に取れば DIFF1S が 6 桁位， DIFF1D が 16 桁位である．

【注意事項】

- LCC の値は 1 が標準値であろう．
- 微分する点 a と $f(z)$ の特異点との最短距離を R とするとき，RADIUS $\leq \frac{2}{3}R$ であって，関数 $f(z)$ の複雑さ，点 a の近傍の状態を考慮して RADIUS を決定しなければならない．
- 本サブルーチンを用いて得られた結果を公表する場合は，プログラム名と作成者名を明記すること．
- 倍精度ルーチン DIFF1D を用いるとき F, ZPOINT, RADIUS, ZVALUE, EPS は倍精度宣言しなければならない．
- パラメータ EPS の内容は演算後こわされるがパラメータはそのまま保存される．

	e^z	$e^{\sin z+z^8}$	$\sqrt{4+\sin x}$	$\dfrac{1}{3-z}$	$e^{\frac{1}{3-z}}$	$\dfrac{e^z}{\cos^3 z+\sin^3 z}$
RADIUS	1.20	0.40	1.20	0.80	0.80	0.32
DIFF1S						
(real)	0.0	3.0×10^{-8}	$3,0\times10^{-8}$	0.0	2.5×10^{-8}	0.0
(imaginary)	1.2×10^{-8}	2.1×10^{-9}	3.7×10^{-9}	5.8×10^{-9}	2.7×10^{-9}	3.4×10^{-9}
DIFF1D						
(real)	3.5×10^{-18}	6.5×10^{-18}	2.6×10^{-18}	2.1×10^{-18}	4.3×10^{-18}	1.7×10^{-18}
(imaginary)	8.3×10^{-19}	8.8×10^{-19}	8.2×10^{-19}	4.9×10^{-19}	1.1×10^{-18}	6.2×10^{-19}
DIFF1S						
(real)	1.8×10^{-7}	3.0×10^{-8}	5.1×10^{-7}	2.6×10^{-7}	4.8×10^{-8}	9.3×10^{-8}
(imaginary)	2.2×10^{-7}	6.5×10^{-7}	1.6×10^{-7}	7.2×10^{-7}	1.2×10^{-7}	1.7×10^{-6}
DIFF1D						
(real)	2.2×10^{-17}	9.5×10^{-17}	1.1×10^{-15}	9.7×10^{-18}	8.9×10^{-18}	6.8×10^{-18}
(imaginary)	6.9×10^{-18}	1.3×10^{-16}	8.3×10^{-17}	2.1×10^{-17}	7.5×10^{-18}	6.0×10^{-17}

$a=0$, LCC$=1$ 計算結果．上が階数１，下が階数５．数値は相対誤差をあらわす．

【参考文献】

- J.N.Lyness, Differentiation Formulas for Analytic Functions, Math.Comp.Vol.22, 1968, pp.352-362.
- 角秀吉, 解析関数の数値微分公式 I, 久留米高専研究報告, 第 18 号, 1971, pp.37–45.
- 角秀吉, 解析関数の数値微分公式 II, 久留米高専研究報告, 第 19 号, 1972, pp.27–36.

5.4 MINMAX

線形方程式のミニマックス解 (Minimax Solution to Overdetermined Linear Systems)

プログラム形式	サブルーチン副プログラム
作成者 作成年月日	須永 照雄 (九州大学工学部) 1971 年 11 月 27 日
ソースの公表 使用言語	'QS.LIBSC(H1MINMAX)'(MSP) Fortran
使用 OS	MSP(M1800, VP2600)

【 概要 】

1. 目的
   線形方程式
   $$\sum_{j=1}^{n} a_{ij}x_j = d_i, \qquad (i = 1, 2, \cdots, m)$$
   において $m > n$ とすると，残差
   $$\gamma_i(\mathbf{x}) \equiv \sum_{j=1}^{n} a_{ij}x_j - d_i, \qquad (i = 1, 2, \cdots, m)$$
   は全てゼロとできない．そこで，残差の最大 $\max_i |\gamma_i(\mathbf{x})|$ を最小とする解 $\hat{\mathbf{x}}$ 即ち
   $$\max_i |\gamma_i(\hat{\mathbf{x}})| \le \max_i |\gamma_i(\mathbf{x})|, \quad \forall \mathbf{x} \in \mathbf{R}^n$$
   なる $\hat{\mathbf{x}} \in \mathbf{R}^n$ を求める．

2. 計算方法
   これは本質的には LP [2] の問題であるが，次の手順で解けることが証明されている（参考文献参照）．以下 $A_i = (a_{i1}\ a_{i2}\ \cdots\ a_{in}) \in \mathbf{R}^n$ とおく．

   - $m = n + 1$ のとき
     (a) $\sum_{i=1}^{n+1} \lambda_i A_i = 0$ を解く．
     (b) $\varepsilon \equiv -\sum_{i=1}^{n+1} \lambda_i d_i \ / \ \sum_{i=1}^{n+1} |\lambda_i|.$
     (c) $A_i\mathbf{x} = d_i + \varepsilon S_i, \qquad (i = 1, 2, \cdots, n+1)$ を解く．
     但し $S_i \equiv sgn(\lambda_i).$
   - $m > n + 1$ のとき
     最初の $n + 1$ 個の方程式につき $m = n + 1$ のときの手順で $\mathbf{x}$ を求める．
     (a) $|\gamma_\alpha(\mathbf{x})| > |\varepsilon|$ のうち最大のものを α とおく．もしなければ Stop.
     (b) $A_\alpha = \sum_{i=1}^{n+1} \rho_i A_i$ を解く．

[2] Linear Programming： 限られた制約条件より目的に合った最良の解を選ぶ数理計画法の一種．特に LP 問題は，制約条件が線形不等式，目的関数が線形式で表現される問題を指す．

(c) $\dfrac{\sigma_\alpha \times S \times \rho_i}{\lambda_i}$ のうち最大の i を β とする．但し $\sigma_\alpha = sgn(\gamma_\alpha(\mathbf{x}))$, $S = sgn(\varepsilon)$.

(d) $A_1, \cdots, A_{\beta-1}, A_{\beta+1}, \cdots, A_{n+1}, A_\alpha$ が新しい reference set となる．これを新しい $n+1$ の方程式として再度 $m = n+1$ の時の手順で解く．以上の手順を繰り返す．

【 使用方法 】

1. 呼び出し方法
   CALL MINMAX(A,K,M,N,P,NN,X,KROW,EPS,ILL)

2. パラメータ

A　　: 実数型配列名．$A(K,L)$ なる二次元配列．
　　　但し $K \geq M,\ L \geq N+1$ であること (図 1 参照)．
K　　: 整数型変数名または整定数．配列 A の行数を与える．$K \geq M$ ．
M　　: 整数型変数名または整定数．方程式の数を与える．$N+1 \leq M \leq 400$.
N　　: 整数型変数名または整定数．変数の個数を与える．$1 \leq N \leq 80$.
P　　: 実数型配列名．$P(NN,L')$ なる二次元配列．
　　　サブルーチンの中で作業用の領域として用いる．
　　　$NN \geq N+1,\ L' \geq N+1$ であること (図 2 参照)．
NN　: 整数型変数名または整定数．配列 P の行数を与える．$NN \geq N+1$.
X　　: 実数型配列名．$X(N)$ からなる一次元配列．
　　　計算結果のミニマックス解が与えられる．
KROW : 整数型配列名．$KROW(N+1)$ なる一次元配列．
　　　reference set の行番号が与えられる．
EPS　: 実数型変数名．残差の最大値 $|\varepsilon|$ が与えられる．
ILL　: 整数型変数名．サブルーチンから戻った時の状態がセットされる．
　　　0 — 正常.
　　　3000 — 次のいずれかの状態のとき．
　　　　$M \leq 1,\ M > 400,\ N \leq 0,\ N > 80, M \leq N,\ K < M,\ NN < N.$

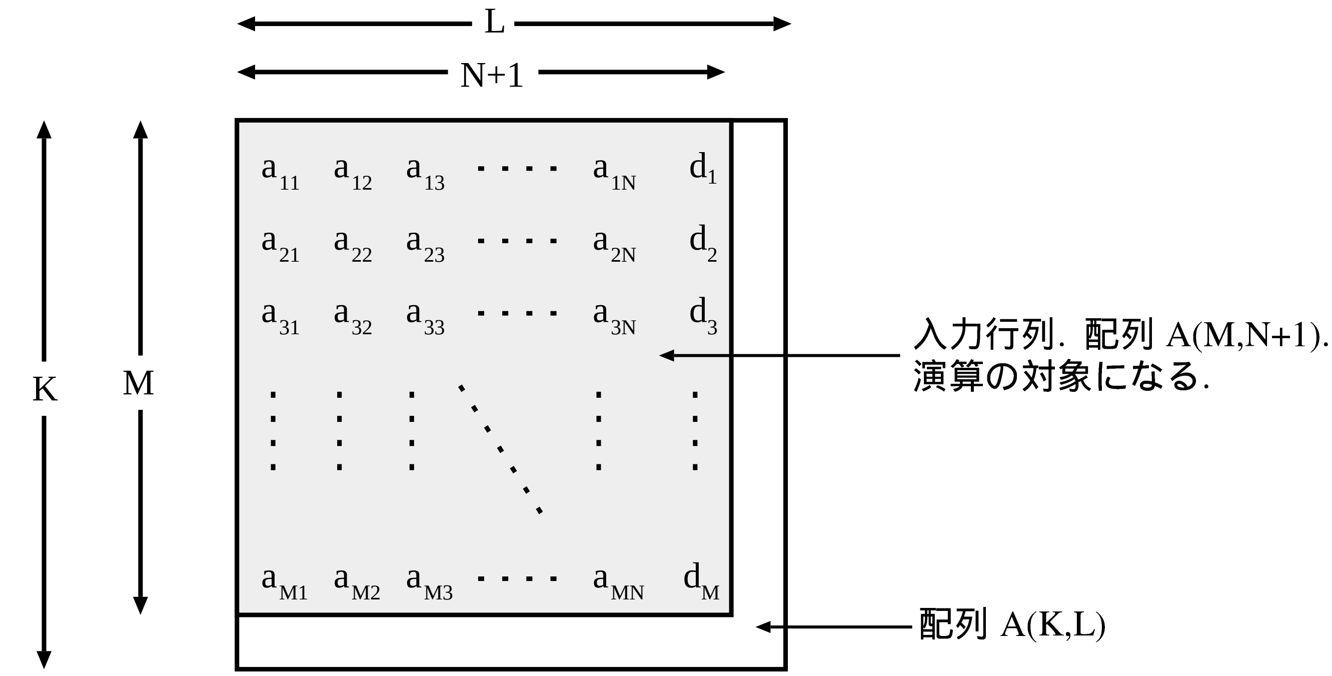
3. 使用ルーチン
   このサブルーチンでは，以下のサブルーチン，基本外部関数を使用している．
   - MINVS(SSL)
   - SIGN, ABS

【使用例】

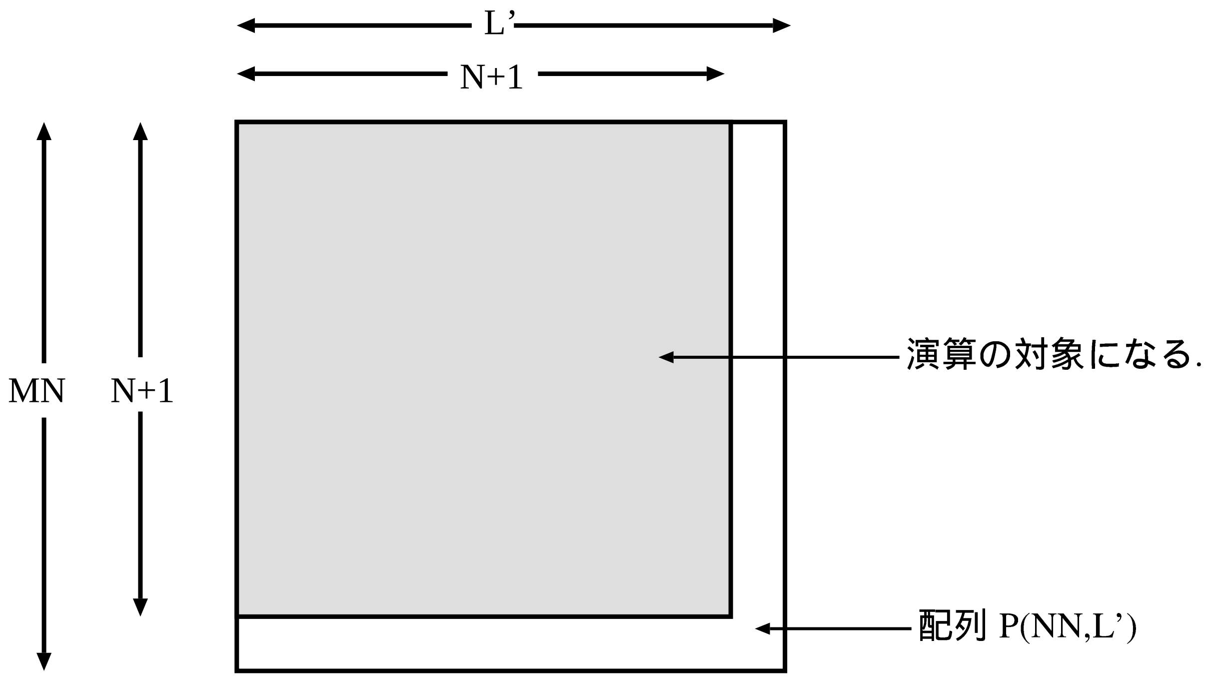
- 線形方程式

$$\begin{cases} 2x_1 + 3x_3 = 3 \\ 3x_1 + x_2 = -1 \\ x_1 + 2x_2 + x_3 = 2 \\ -x_1 + 4x_2 - 3x_3 = 6 \\ 3x_1 + x_2 + 2x_3 = -3 \end{cases}$$

のミニマックス解を求める．



方程式に対応して決まる行列 A



作業用領域としての行列 P

```
C     TEST PROGRAM MINMAX
       REAL A(5,4)/2.0,3.0,1.0,-1.0,3.0,0.0,1.0,2.0,4.0,1.0,
      &     3.0,0.0,1.0,-3.0,2.0,3.0,-1.0,2.0,6.0,-3.0/  ! 行列の定義
       REAL P(4,4),X(3),EPS
       INTEGER KROW(4),ILL

        CALL MINMAX(A,5,5,3,P,4,X,KROW,EPS,ILL)   ! MINMAX の call
         PRINT*,ILL
         PRINT*,EPS
         PRINT*,X(1),X(2),X(3)                    ! 解の出力
       END
```

【参考文献】

- R.H. Bartels & G.H.Golb, Stable Numerical Methods for Obtaining the Chebyshev Solution to an Overdetermined System of Equations, Communications of the ACM, Vol.11, No.6, June, 1968, pp.401–406.

5.5　POWELL

関数の極小値 (Minimization of a Function)

プログラム形式	サブルーチン副プログラム
作成者 作成年月日	竹田 宏 (九州大学教養部) 1971 年 7 月 1 日
ソースの公表 使用言語	`'QS.LIBSC(C7POWELL)'` (MSP) Fortran
使用 OS	MSP(M1800, VP2600)

【 概要 】

1. 目的
   n 個の変数 $x_1, x_2, \cdots, x_n$ の関数 $f(x_1, x_2, \cdots, x_n)$ の極小値と，その極小値を与える x_i の値を誤差 e_i $(i = 1, \cdots, n)$ の範囲で求める．

2. 計算方法
   いわゆる POWELL の方法によるが，これは概略次のような方法である．x_i を座標軸とする n 次元空間を考える．変数に初期値を与えると，この n 次元空間内に 1 点（出発点）が定まる．そこでまず x_1 だけを変化させて（すなわち x_1 軸に平行に動いて）極小点を探す．つぎに第 2，第 3 の軸に平行に動いて極小点を探す．このように，ある直線に沿って動いて極小点を求めることを「直線走査」とよぶことにし，n 個の各方向に 1 通り「直線走査」を行ったら 1 回の「走査」が終わったということにしよう．この「直線走査」はおよそ次のように行う．

   i 番目の方向に，関数が減少する向きに少しずつ移動して，3 つの点 a，b，c でそれぞれ関数値 $f_a,\ f_b,\ f_c$ を求める．$f_a,\ f_b,\ f_c$ を 2 次曲線で結んでみて，それが上に凸ならばさらに移動し，下に凸になるまで移動を続ける．このとき 1 回の移動の距離は $30e_i$ にしてある．下に凸になれば 2 次曲線で結ぶことによって極小の位置を予想する．この予想極小点の位置 d が a，c の範囲外にあればさらに移動を続け，d が a，c の範囲内にくるようにする．d と b との距離が条件 $|x_{ib} - x_{id}| < 0.2e_i$ を満たせば b 点を極小点とする．条件が満たされないときは，例えば $a < d < b$ ならば a，d，b をそれぞれ新しい a，b，c として上の条件が満たされるまで予想極小点を探し続ける．

   さて「走査」が 1 回終わると，出発点と終点を結ぶ直線の方向に少し動いてみて，見込みがありそうなときはこの方向に直線走査を行い，同時に前回の直線走査の中でこの方向に一番近い方向を，この方向に置き換えて次の走査を始める．以下これをくり返し，1 回の走査についての出発点 x' と終点 x との距離が，条件

   $$|x_i - x'_i| < 0.5e_i \quad (i = 1, \cdots, n)$$

   を満たしたとき一応極小点が見つかったとみなす．場合によっては（引数 NTEST=0 のとき）これで終わりであるが，NTEST=1 のときは，$x_i + 10e_i \quad (i = 1, \cdots, n)$ を新しい出発点として再び計算をやり直し，第 2 の極小点を求める．第 1，第 2 の極小点を結ぶ方向に直線走査を行い，第 3 の極小点を求め，これが，第 1，第 2 の極小点のどちらに対しても条件を満たしていれば，これを最後の結果とし，もしそうでなければ，この第 3 の極小点を出発点として再び始めからやり直す．

【 使用方法 】

1. 呼出し方法
   `CALL POWELL(N,X,FVAL,F,E,MAX,NWRITE,NTEST,IRANG,KRANG)`

2. パラメータ

N	:	整数型変数名または整定数で変数の数を入れる．$1 \le N \le 15$ とする．
X	:	実数型配列名．$X(I) \quad (I = 1\,,\,\cdots, N)$ に変数の初期値を入れておく． あとで極小点における変数の値が入る．$N = 1$ のときは単純変数でよい．
FAVL	:	関数副プログラム名．変数の値を与えられたとき関数の値を計算するための 関数副プログラムを利用者が用意しておかねばならない． また EXTERNAL の宣言が必要である．
F	:	実数型配列名．関数の極小値が入る．
E	:	実数型配列名．極小点における変数の値 x_i に許される誤差の値を $E(I)$ に入れておく，$N = 1$ のときは単純変数でも実定数でもよい．
MAX	:	整数型変数名または整定数．$0, 1, 2, 3$ のいずれかを与える． 0 – サブルーチン内ではエラーメッセージ以外は何も出力しない． 1 – 1 回の走査が終わるごとにその時点の X と F の値を出力する． 2 – 1 つの直線走査が終わるごとに結果を出力する． 3 – 関数の値を計算するごとに結果を出力する．
NTEST	:	整数型変数名または整定数．0 または 1 のいずれか． 1 のときは収束の判定をきびしくする． すなおな関数のときは 0，そうでないときは 1 にする.
IRANG	:	整数型変数名または整定数．変数 $X(I)$ の中に，変域が > 0 の範囲に 限られるものが含まれるときは，その個数を入れる．
KRANG	:	整数型配列名．KRANG(I) に，変域が > 0 に限られる変数番号を入れる． IRANG=0 のときは KRANG も 0 としてよい.

3. 出力データの形式
   途中の計算結果は次の形で出力する．

```
LFREQ=(走査の回数) J=(直線走査の方向) F=(関数値)
  各変数の値
```

まず `HAZIME` と印字し，出発点での結果を出力する．次に NWRITE=3 なら，関数の値を計算するたびに結果を出力する．1 つの方向の直線走査が終わると，NWRITE ≥ 2 なら

```
MINMUM ALONG THE j-TH DIRECTION
```

と出力し，またそのときの結果を書き出す．NWRITE=1 のときは 1 回の走査が終わったときだけ結果を出力する． 第 1 の極小点がみつかると，NTEST=0 なら

```
SAIGO NO KEKKA
```

と印字し，結果を印刷し，これで終わる．NTEST=1 なら

```
SYUREN NO TEST
```

と印字し，第 2 の極小点が見つかるまでは前と同じ形の印刷を続ける．第 2，第 3 の極小点がみつかって，収束の条件が満たされていれば

```
SAIGO NO KEKKA
```

という字と，最後の結果を出力するが，条件が満たさなければ

```
TEST DAME YARINAOSI
```

と出力する．

4. エラーメッセージ
   - `NFREQ IS OVER J=j`
     j 番目の方向に直線走査したとき，30 回移動しても極小点が見つからなかったとき，このメッセージを出す．しかし計算は実行するので，あとでさらに次のメッセージが出ていなければ気にする必要はない．
   - `NO . OF ITERATION EXCEEDS KFMAX`
     MAX 回走査をくり返しても極小点が見つからなかった場合である．出発点を変えるか，許容誤差を大きくして再計算せねばならぬが，このメッセージの前に特定の J の値に対して先のメッセージが何度も出ているときは，特に J 番目の変数の初期値と許容誤差に注意すべきである．再計算に際しては，なるべく NWRITE=3 にして出力リストから関数値の変わりぐあいが読み取れるようにした方がよい．
   - `F DOES NOT VARY`
     ある 1 回の走査の前後で関数値が全く変化しなかったとき．原因としては出発点が偶然にも極小点に一致していた場合と，極小が「なべ底型」の場合とが考えられる．NWRITE>0 で，このメッセージの前に結果が出力されており，それが LFREQ>1 になっているならば間違いなく後者である．LFREQ=1 のときどちらの場合かを見分けたければ，出発点を変えて再計算すればよい．また同じメッセージが出れば後者であり，前者ならばこんどは「最後の結果」が出るはずである．
5. 使用ルーチン
   MOVE

【使用例】

- $-x\exp(-x)$
  極小点 $x=1$，極小値 $F=-0.367879$ (exact solution)
  $x=3$ から出発し，誤差 $e=0.01$ として結果
  $x=1.0002$，$F=-0.367879$ を得た．

- 次の関数をテストした．

$$-\frac{1}{1+(x_1-x_2)^2}-\sin(\frac{\pi}{2}x_2x_3)-\exp\{-(\frac{x_1+x_3}{x_2}-2)^2\}$$

exact な極小点は $x_1=x_2=x_3=\sqrt{4m+1}\quad(m=0,1,2,\cdots)$，$F=-3$ である．
この関数は，$x_1=0$，$x_2=1$，$x_3=2$ から出発すると途中で異常な振る舞いを示すが，$NTEST=1$ とすることにより，次の結果が得られた．
$x_1=1.0001$，$x_2=1.0001$，$x_3=0.9998$，$F=-3.00000$
ただし誤差としては $e_1=e_2=e_3=0.003$ としている．

【参考文献】

- M.J.D.Powell, Computer J. 7, pp.155, 1964.

5.6 PRESNL

一般化されたフレネル積分 (Generaized Fresnel integral)

プログラム形式	サブルーチン副プログラム
作成者 作成年月日	田村 英之 (九州大学工学部) 1971 年 4 月 28 日
ソースの公表 使用言語	`'QS.LIBSC(C3PRESNL)'` (MSP) Fortran
使用 OS	MSP(M1800, VP2600)

【 概要 】

1. 目的

$$\int_0^x \exp(j\ t^p)dt = C_p(x) + jS_p(x), \quad j = \sqrt{-1}, \quad p = 1 \sim \infty$$

を一般化されたフレネル積分と呼ぶ（$p = 2$ が通常のフレネル積分であるが，定義の仕方で変数と積分値は定数倍だけ異なる）．この計算を，ベキ級数，漸近展開の最適併用条件により計算する．

2. 計算方法
$x \to 0$ ではベキ級数を，$x \to \infty$ では漸近展開を使い，両方法の誤差が等しくなる x で切りかえを行う．詳細は文献を参照すること．

【 使用方法 】

1. 呼出し方法
`CALL PRESNL(P,X,C,S)`

2. パラメータ

P : 指数 p $(1 \sim \infty)$ を与える倍精度実数型変数名または倍精度実定数．
X : 独立変数 $x > 0$ を与える倍精度実数型変数名または倍精度実定数．
C : 結果の積分値の実数部がセットされる．倍精度実数型変数名．
S : 結果の積分値の虚数部がセットされる．倍精度実数型変数名．

3. 制限
$1 < P < \infty, \qquad X \geq 0$

4. 使用ルーチン
組込み関数 : DABS，DLOG，DCOS，DSIN，DSQRT

5. 精度
P, X によって異なるが，倍精度の場合，10^{-10} 以上が保証される．20 で最も誤差が大であるが，$x^p \geq 32, x^p \geq 9$ では 10^{-15} 程度であろう．

【注意事項】

- プログラムの一部を変更することにより，計算機の語長に応じて精度最良の条件で計算ができる．
- 通常のフレネル積分の定義

$$\int_0^x \exp(j\frac{2}{\pi}t^2)dt = C(x) + jS(x)$$

による $C(x)$, $S(x)$ はそれぞれ，

$$C(x) = \sqrt{\frac{2}{\pi}}C_2(\sqrt{\frac{\pi}{2}}x), \qquad S(x) = \sqrt{\frac{2}{\pi}}S_2(\sqrt{\frac{\pi}{2}}x)$$

である．

【参考文献】

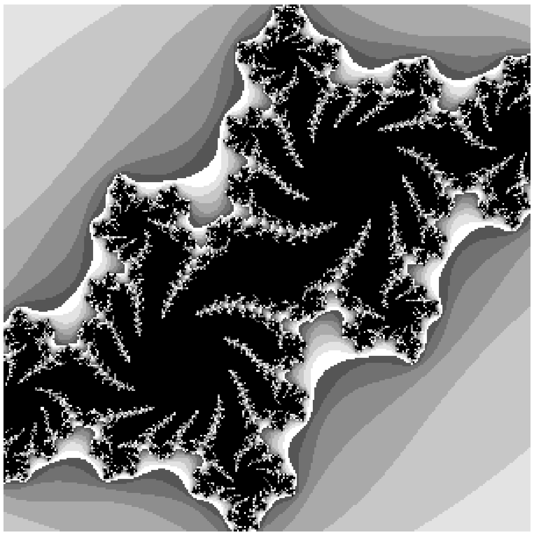
- 森口ほか, 数学公式 III, pp22, pp261, 岩波全書.
- 田村 英之, 九州大学工学集報， Vol.43, No.4， pp.550, 1970.

6

統計解析，数理計画問題

統計解析，数理計画問題，データ解析ソフトウェアは次の通り．

ライブラリ名	機能	MSP	UXP
AMPS	数理計画問題の最適化ソフト	○	
AMPS/VP	AMPS の VP 版	○	
ANALYST	富士通提供統計データ処理パッケージ	○	
BMDP	統計解析汎用プログラムパッケージ	○	
FACEST	因子分析 – 因子負荷行列の推定	○	
FACROT	因子分析 – 直交および斜交回転，評点推定	○	
PCAJCB	主成分分析 – Jacobi 法	○	
PCAQR	主成分分析 – QR 法	○	
S	AT&T 開発の汎用統計解析システム		○
SALS	最小二乗法標準プログラム	○	
SAS	汎用統計解析システム Graphics 機能付き	○	
SPSSX	社会科学のための統計パッケージ	○	



自己同型フラクタル I

6.1 AMPS, AMPS/VP

【概略】

AMPS(Advanced Mathematical Programming System) は，線形計画問題の他，混合整数問題などの数理計画問題の最適化を行なうソフトウェア．ユーザーサブルーチンをシステムに組み込んだり，ユーザープログラムからシステムを機能別に呼び出したりすることが可能．

AMPS/VP は AMPS の処理をベクトルプロセッサ VP2600 上で処理するためのソフト．AMPS/VP の処理はバッチのみ．

【処理環境】

- MSP/M1800, MSP/VP2600
  command : `AMPS`
  cataloged procedure : `AMPS`

【注意事項】

- AMPS, AMPS/VP は MPS/Xの後継にあたる．

【提供】

- 富士通株式会社

【参考文献】

- 「FACOM OS IV AMPS 説明書」（70SP-7190），富士通株式会社．
- 「AMPS, AMPS/VP の公開」九州大学大型計算機センター広報, Vol.26, No.4, pp.503–508.

6.2 ANALYST

【概略】
ANALYST(Analyzer for Statistical Data) はエンドユーザー志向，業務処理志向の統計データ処理パッケージ．処理モードとして MSP 上のバッチ処理モードと会話処理モードを備える．統計データの処理機能としては，統計情報の管理や加工・編集をはじめ，集計や帳簿作成などの統計グラフィックス機能までの幅広い機能を有する．

【処理環境】

- MSP/M1800
  command : `ANALYST`
  cataloged procedure : `ANALYST`

【提供】

- 富士通株式会社

【参考文献】

- 「FACOM ANALYST 解説書」(99SG-7401)，富士通株式会社．
- 「利用の手引・ MSP コマンド編」九州大学大型計算機センター・ライブラリ室, 1993.
- 「利用の手引・バッチジョブ編」九州大学大型計算機センター・ライブラリ室, 1992.

6.3 BMDP

【概略】
BMDP は主に生物医学の分野に適した統計解析システム． BMDP の処理を利用者の Fortran プログラムと結合しておこなうことも可能．処理は MSP のバッチのみ．

【処理環境】

- MSP/M1800
  cataloged procedure : BIMED, BIMEDT

【使用例】

- BMDP プログラム BMDP1D を用いて解析を行なう．解析データの入力方法は参考文献参照．

```
//A79999A JOB CLASS=A
//   EXEC BIMED,PROG=BMDP1D
//GO.SYSIN DD *
   （コントロール情報・解析データ）
//
```

【主な BMDP プログラム】

- Factor Analysis
- Life Tables and Survivor Functions
- Unbalanced Repeated Measures Models with Structured Covariance Matrices
- General Mixed Model Analysis of Variance Equal Cell Sizes
- Polychotomous Stepwise Logistic Regression
- Histograms and Univariate Plots
- Linear Scores for Preference Pairs
- Cluster Analysis of Variables

【提供】

- University of California

【参考文献】

- 「Statistical Software Manual 1990 Revision Vol.1, Vol.2」University of California Press.
- 「利用の手引・バッチジョブ編」九州大学大型計算機センター・ライブラリ室, 1992.

6.4 FACEST, FACROT

- 因子分析 – 因子負荷行列の推定 (Factor Analysis – Estimation of Loading Matrix)
- 因子分析 – 直交および斜交回転，因子評点の推定 (Factor Analysis – Orthogonal and Oblique Rotation, Estimation of Factor Scores)

プログラム形式	コンプリートプログラム
作成者 作成年月日	柳川 尭，工藤 奈津子 (九州大学理学部) 1976 年 8 月
ソースの公表	'QS.LIBSC(G6FACEST)' 'QS.LIBSC(G6FACROT)'(MSP)
使用言語	Fortran
使用 OS	MSP(M1800)

【 概要 】

(1) 目的

たとえば，児童の知能程度を解明する際，一連のテストがなされ，採点成績が示される．しかし，このテストの一つ一つの成績は，計数能力，読解力，推理力，注意力など，知能を表す各因子の絡み合い，さらにその時の調子などによるバラツキが付加され観測されるものである．もし知能を決定する内因因子を何らかの方法で取り出すことができるなら，そしてこれら各因子について各個体毎の因子評点がいかほどであるかを知ることができるなら，むしろその方が一連のテストの採点成績を知ることよりも，望ましいといえよう．因子分析法はこのような要望に応える目的で心理学者達によって開発された．このような考え方は他の分野にも適用されるものであり，今日，因子分析法は心理学者のみならず多くの分野で幅広く適用されている．

因子分析法では，因子負荷行列のいくつかの要素をゼロまたはある値に指定しておいて残りの要素を推定する検証的な場合(confirmatory factor analysis) と，因子負荷行列の要素に何の仮定もおかない探索的な場合 (exploratory factor analysis) の二つの方法が知られているが，本プログラムは後者，即ち探索的な場合のプログラムである．

因子分析法は後で述べるように，因子負荷行列の推定，回転 (直交，斜交) による因子の解釈および因子評点の推定という三段階より成っている．本プログラムでは，因子負荷行列の推定と，回転 (直交，斜交) および因子評点の推定を分け

- Factor Analysis – Estimation of Loading Matrix (実行形式プログラム名：FACEST)
- Factor Analysis – Orthogonal And Oblique Rotations, Estimation of Factor Scores (実行形式プログラム名：FACROT)

の完全に独立した二つのプログラムにおさめている．従って，因子分析をおこなうには，まず前者のプログラムで因子負荷行列の推定を行うことになる．探索的な因子分析法では推定が不可能な場合，不適当な解が得られる場合なども生じるので，この段階で得られる推定値については十分吟味しておくことが必要となる．ある場合には，特性値 (テスト項目) の加減，入れ換えや，共通因子数を設定し直しながら，何度も繰返し本プログラムを使用し，因子負荷行列の最適な推定値を求めることが必要となる．後者のプログラムではこのようにして得られた推定値を入力データとして利用し，回転 (直交，斜交) および因子評点の推定をおこなうこ

とにしている．

(2) 因子分析法の概説

因子分析法は実質科学の多くの分野に関連しているために，用語の不統一，混合が多い．従って，本プログラムで取り扱う因子分析法について，かなり詳しい概説を以下に与えておく．

因子分析法は，調べたい対象の因子構造を共分散行列 (covariance matrix)，または相関行列 (correlation matrix) の内部構造を調べることによって明らかにする．かかる方法に 2 つの接近法があり，1 つは主成分分析法 (principal component analysis)，他方は狭い意味での因子分析法と呼ばれている．本プログラムは後者，しかも最尤法 (maximum likelihood method，あるいは maximun likelihood factor analysis) と呼ばれている方法を採用している．最尤法は次のように行われる．

1. p 個の特性値 (p 変量) のデータ $(x_1, x_2, \cdots, x_p)$ を各要素が，平均 0，分散 1 を持つように規準化された p 次元確率変数 $(X_1, X_2, \cdots, X_p)$ の実現値とみなし，次の基本仮定をおく．

$$\begin{pmatrix} X_1 \\ X_2 \\ \vdots \\ X_p \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1k} \\ a_{21} & a_{22} & \cdots & a_{2k} \\ \vdots & \vdots & \ddots & \vdots \\ a_{p1} & a_{p2} & \cdots & a_{pk} \end{pmatrix} \begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_k \end{pmatrix} + \begin{pmatrix} \varepsilon_1 \\ \varepsilon_2 \\ \vdots \\ \varepsilon_p \end{pmatrix} \tag{1}$$

   ここに，k は指定された定数，f_i は $X_1, X_2, \cdots, X_p$ の 2 つ以上に含まれる共通因子 (common factor)(確率変数)，$(p \times k)$ 行列 $A = (a_{ij})$ は，因子負荷行列(factor loading matrix)(未知定数)，$\varepsilon = (\varepsilon_1, \varepsilon_2, \cdots, \varepsilon_p)^T$ は誤差項 (確率変数) である．

   さらに，$\varepsilon_1, \varepsilon_2, \cdots, \varepsilon_p$ は互いに独立で正規分布に従うこと，ε と $f = (f_1, f_2, \cdots, f_k)^T$ が独立であるという数学的仮定がおかれている．

2. 上のモデルを仮定した時，

$$E(f_i) = 0 \qquad Var(f_i) = 1 \qquad (i = 1, 2, \cdots, k)$$

$$E(\varepsilon_i) = 0 \qquad (i = 1, 2, \cdots, p)$$

   とすることができる．また，$X = (X_1, X_2, \cdots, X_p)^T$ の共分散行列 (これは，相関行列と等しくなる) を Σ, f および ε の分散共分散行列をそれぞれ Φ および Ψ (ただし，上のことより Φ の対角要素はすべて 1，Ψ は対角要素を ψ_i $(i = 1, 2, \cdots, p)$ とする対角行列である) とすると，関係式

$$\underbrace{\Sigma}_{p\times p} = \underbrace{A}_{p\times k} \underbrace{\Phi}_{k\times k} \underbrace{A^T}_{k\times p} + \underbrace{\Psi}_{p\times p} \tag{2}$$

   が成立する．$f_1, f_2, \cdots, f_k$ がお互いに無相関なとき，共通因子 $f_1, f_2, \cdots, f_k$ は直交因子 (orthogonal factor) と呼ばれている．このとき Φ は単位行列となり，(2) 式は $\Sigma = AA^T + \Psi$ となる．$f_1, f_2, \cdots, f_k$ が直交因子でないとき，いいかえれば $f_1, f_2, \cdots, f_k$ が相関関係を持つ時，これらは斜交因子(oblique factor) と呼ばれている．

因子分析法の第一段階は，n 個の観測値ベクトル

$$\tilde{x_1} = (x_{11}, x_{21}, \cdots, x_{p1})^T, \cdots, \tilde{x_n} = (x_{1n}, x_{2n}, \cdots, x_{pn})^T$$

が与えられたとき，Σ の推定値である標本相関行列 R を求め，関係式 (2) を利用して因子負荷行列 A (および Ψ) の妥当な推定値を求めることである．ところで，式 (2) の A と Φ は基本的には一意に定まらないという性質がある．それは，k 次元正則行列 T により関係式

$$A\Phi A^T = (AT)(T^{-1}\,\Phi\,T^{T^{-1}})(AT)^T$$

が成り立ち，上のようにして得られた A の推定値は，実は AT の推定値であるかもしれないからである．このことは，回転に関する因子の不定性とよばれている．従って，標本相関行列 R から A の推定値を定めるには，まず最初に (2) 式で Φ を単位行列とおいた最も簡単な式

$$\underbrace{\Sigma}_{p\times p} = \underbrace{A}_{p\times k}\underbrace{A^T}_{k\times p} + \underbrace{\Psi}_{p\times p} \tag{3}$$

を利用して A の推定値を求めることが第一の目的となる．(3) 式の対角線要素の間には

$$1 = \sum_{\alpha=1}^{k} {a_{i\alpha}}^2 + \psi_i \qquad (i = 1, 2, \cdots, p)$$

の関係が成り立つ．

$$h_i^2 = \sum_{\alpha=1}^{k} {a_{i\alpha}}^2, \qquad (i = 1, 2, \cdots, p)$$

は X_i の k 個の因子で説明される割合を表し，共有性 (communality) とよばれる．

本プログラムでは，上の段階を Factor Analysis – Estimation of Loading Matrix で行う．まず，入力データより相関行列を計算し (あるいは相関行列を入力する), 出力する．続いて，繰返し計算の回数，A の推定値である Loading Matrix，対角行列 Ψ の対角要素の推定値である PSI，および共有性 communality を出力する．また，指定によっては Loading Matrix をデータセットに出力させ，第 2 段階への入力とすることができる．

因子分析法の第 2 の段階は，回転に関する因子の不定性を利用して，上で得た因子負荷行列 A の推定値 $\hat{A}$ を実質科学の分野で具体的な解釈のつけやすい構造を持った因子負荷行列 $\hat{A}T$ に変換することである．共通因子が直交因子であるとき，このような変換は，直交回転(orthogonal rotation) と呼ばれ，斜交因子のときは，斜交回転(oblique rotation) と呼ばれる．T は変換行列(transformation matrix) と呼ばれている．直交回転を与える T は直交行列である．従って，共通因子の直交性はこの変換によって変わらず，式 (1) は，

$$\tilde{X} = AT(T^T f) + \varepsilon$$

と同等になる．よって，$G=AT,\ f^*=T^Tf$ とおけば，$\tilde{X}$ は規準化されているので，X と f^* の相関行列は G になる．かくして A の推定値 $\hat{A}$ を直交回転して得られた因子負荷行列 $\hat{G}$ と特性値 (テスト項目) との関係を調べれば因子の解釈が可能となり，各因子に適当な名前がつけられることになる．

本プログラムでは，直交回転は Factor Analysis – Orthogonal and Oblique Rotation, Estimation of Factor Scores の前半で行われる． Orthogonal Rotation が出力されたあと，繰返し計算の回数，Loading Matrix として $\hat{G}$, communality, および Transformation Matrix として T が出力される．斜交回転では，(1) 式は

$$\tilde{X}=AT(T^1f)+\varepsilon$$

と同等になる．よって $G=AT,\ f^*=T^{-1}f$ とおけば，$\tilde{X}$ と f^* の相関行列は

$$\begin{aligned} Corr.(\tilde{X},f^*) &= GE(f^*f^{*T}) \\ &= G(T^TT)^{-1} \end{aligned} \tag{4}$$

であたえられる．よって斜交回転では直交回転と異なり，$\tilde{X}$ と f^* の相関行列は変換された因子負荷行列 G と一致しない．これにより，斜交回転では，因子負荷行列という言葉は一般に使われず，(4) 上式で与えられる

$$Corr.(\tilde{X},f^*)=G(T^TT)^{-1}=A(T^T)^{-1}$$

を因子構造行列 (factor structure matrix)，$(T^T)^{-1}$ をそれへの変換行列，これまで因子負荷行列とよんだ $G=AT$ を因子パターン行列 (factor pattern matrix)，T をそれへの変換行列とよび，両者を区別する．斜交回転で因子の解釈を行うには，テスト項目と因子構造行列の対応を調べるほうが容易である．斜交回転では，因子構造，因子パターンの他に (4) 式の $E(f^*f^{*T})$，いいかえれば各因子の因子間相関行列 (factor correlation matrix), および各因子の因子軸間角度 (angle for each factor) を調べておくことも重要である．また，斜交回転では (3) 式は

$$\Sigma=G(T^TT)^{-1}G^T+\Psi$$

と同等である．上式の対角線要素の間には

$$1=\sum_{\alpha=1}^{k}{g_{i\alpha}}^2+\sum_{\alpha=1}^{k}\sum_{\beta=1}^{k}g_{i\alpha}\,g_{i\beta}\,t^{\alpha\beta}+\psi_i \quad (i=1,2,\cdots,p)$$

ただし，

$$G=(g_{ij}),\qquad (T^TT)^{-1}=(t^{ij})$$

の関係が成立する．従って，共有性は直接寄与 (direct contribution) とよばれる項 $\sum g_{i\alpha}^2$ と同時寄与 (joint contribution) とよばれる項

$$\sum_{\alpha=1}^{k}\sum_{\beta=1}^{k} g_{i\alpha}\, g_{i\beta}\, t^{\alpha\beta}$$

の和で表される．

本プログラムでは，斜交回転は直交回転と同じプログラムの中で行い，Oblique Rotation を出力した後，繰返し計算の回数，因子間相関行列，因子軸間角度行列，因子構造行列とそれへの変換行列，因子パターン行列とそれへの変換行列，直接寄与，同時寄与および共有性を出力する．

直交回転，斜交回転のいずれの結果を採用するかについては，実質科学より得られる共通因子に対する知識を計算結果の検討に基いて決定すればよい．

さて，このようにして内因因子の推定と解釈が行われると，次に問題になるのは，各個体(individual) がそれらの因子をどれほど所有しているかという各個体の 因子評点(factor scores) を推定することである．これが因子分析法の最終段階となる．本プログラムでは，因子評点の推定はそれぞれの回転の直後で行い，直交回転，斜交回転の各々の場合について，
因子評点係数行列(coefficient matrix of factor scores) および各個体の因子評点ベクトル(factor scores for individual) を出力している．

(3) 因子分析法の問題点

因子分析法は，主成分分析法とは異なり，かなり厳しい仮定を設けた上で内因因子を取り出し，その解釈を与え，各個体毎の因子評点がいかほどであるかの推定を積極的に与える方法である．従って，仮定が厳密に成立するとき，因子分析法はきわめて満足のいく結果をもたらすことになる．しかしながら，内因因子を探索的に探り出す因子分析の本来の目的を考えるならば，これらの仮定は，内因因子に関係して設立されるために多くの場合検証できないのが普通である．残念なことに仮定が成立しないとき，因子分析法は多くの致命的な問題を持つことが知られており，その多くは理論的にまだ解明されていない．これらの仮定を「仮説」におきかえ，因子分析法を仮説の検証として位置付け，使用する場合も多いようであるが，極めて細心の注意が必要である．因子分析を適用する場合は，まず最初に問題点が少ない主成分分析を行い，対象に対する十分な知見を得ておくことが必要である．多くの場合，因子分析を行うまでもなく，主成分分析法で十分満足のいく結果が得られるものである．因子分析では一度の計算で満足のいく結果を期待しない方がよい．たとえ満足のいく結果が得られたとしても，さらにテスト項目の加減，入れ換え，因子数の吟味など試行錯誤を反復し，積み重ねることによって初めて，良い結果が導かれるものである．次に本プログラムの関連を考慮しつつ，因子分析法の重要な問題点を 2 つ挙げておく．

1. 共通因子数 k の問題
   (3) 式により左辺の Σ は $(p\times p)$ 対称行列であるから，方程式の個数は $p(p+1)/2$ 個，これに対して未知数の個数は対角行列 Ψ に p 個，A に見かけ上 pk 個あるが，この A は回転の不定性より $(p \times k)$ 直交行列 T によって，AT ともおきかえられるので，これを差し引けば実際には，A は $pk - k(k-1)/2$ 個の未知母数がある．因子負荷行列 A が推定可能であるためには (3) 式で与えられる方程式の個数が未知母数の個数よりも多いことが必要となる．いいかえれば，p と k は次の不等式

$$\frac{p(p+1)}{2} - [\,p + (pk - \frac{k(k-1)}{2})\,] = \frac{(p-k)^2 - (p+k)}{2} \geq 0$$

を満たさなければならない．この不等式を満たす p と k の関係は次表で示される．

$k=$	1	3	6	10	15	21	$\cdots$
$p>$	3	6	10	15	21	28	$\cdots$

普通，観測特性 (テスト項目) の個数 p は共通因子数 k の少なくとも 2 倍ないし 3 倍は必要である．また，当然のことであるが，観測データの個数 (個体数) n は，p より大きくなければならない．

本プログラムでは，k の値は入力データとして与えるようになっているので，上の条件を満たすように k の値を指定することが最低限必要である．しかし，たとえ k が上の条件を満たしても，指定された k が「真の値」からかなり離れていると，計算が途中で続行できなくなったり，不適当な解が得られたりする事が知られているので，入力する k の値については，実質科学あるいは主成分分析の結果得られる知見などを利用してなるべく「真の値」に近いものを指定するよう心がけねばならない．

本プログラムでは，指定された k の値に対してモデル (1) の仮説検定を Bartlett によって与えられた χ^2 検定により危険率 5% で行っている．従って，出力で

```
NK IS REJECTED AT 5 PERCENT CRITICAL LEVEL
```

と印刷されたら共通因子数 k の値を $k-1$ または $k+1$ に設定し直し，再スタートする．最適な k はこれを繰り返すことによって定めればよい．

2. 不適当な解 (improper solution) の問題

不適当な解は，理論的および技術的な面より生ずることが知られている．理論的な面では，不適当な解は主モデル (1) 式および仮説が満たされない時生じる．特に，因子負荷行列 A が特殊な型をした行列であるとき，また $f_1, f_2, \cdots, f_k$ の中に特性値 (テスト項目) の 2 つ以上にわたって含まれるものがないとき，いいかえれば $f_1, f_2, \cdots, f_k$ の中に共通因子でないものが存在するとき不適当な解が得られることが知られている．これを防ぐには，テスト項目の加減，入れ換え等の工夫が必要である．

本プログラムでは，技術的な面で不適当な解が得られることがないように色々な工夫を凝らし細心の注意を払っている．しかし，不適当な解が得られたとき，それが理論的な理由によるものか，技術的な理由によるのか見分けるのは困難である．津村 (文献 [4]) は，不適当な解が得られたら因子分析をあきらめることを勧めている．

本プログラムでは，不適当な解が得られると，“IMPROPER SOLUTION” と出力し，これを与えるテスト項目を指定するようになっている．一般に，探索的な因子分析法の初期には不適当な解が得られることがかなり多いと予想されるので，たとえ出力で “IMPROPER SOLUTION” と出力されても，テスト項目の加減，入れ換え，または因子数の吟味などを行い，再スタートしてみるほうがよい．このような試行錯誤を繰り返しながら，なおかつ improper solution が得られるならば，因子分析はあきらめるべきである．

(4) 本プログラムの背景をなす理論について

各段階において，多様な理論と手法が多くの研究者達によって考案されている．このうち，本プログラムで採用した手法は以下のとおりである．個々の手法の詳しい説明や理論については引用した参考文献を見ていただきたい．

1. 因子負荷行列の推定
   因子負荷行列 A の推定は，Lawley によって与えられた最尤法 (maximum likelihood method) を採用した．この考え方については [2] に詳しい．また，推定値を求める繰返し法による計算は Jöreskog [5] によって与えられた収束のはやい方法を採用した．これについての簡単な説明は [2], pp.31 ~ 36 の訳者注で手際よく行われている．因子分析法の理論，および技術の両面の問題点は全てこの段階にあるので，因子負荷行列の推定理論をよく理解し，適用限界を洞察しておくことが因子分析の成功を導く秘訣であるといえる．

2. 因子の回転と解釈
   直交回転として，Kaiser によって提唱されたバリマックス法(Varimax method), 斜交回転として，カイザー・ディックマン法 (Keiser-Dickman method) を採用した．両者の計算は，繰返し法 (iteration method) によって行われるが，繰返しの打ち切り回数を 30 回，収束の判定精度を 1.0×10^{-4} に固定している．カイザー・ディックマン法では参考因子構造(reference factor structure) 行列とその変換行列が求められるが，本プログラムでは最終的にこれを根元因子構造 (primary factor structure)，根元因子パターン (primary factor pattern) に変換し，出力している．このことを明確にするために，出力では "`(PRIMARY)`" と出力している．また同時に，参考因子構造行列およびその変換行列も中間結果として "`(REFERENCE)`" として出力する．これらの方法の考え方および因子解釈の与え方などについては，文献 [1] に詳しい．なお，本プログラムは [1] のプログラムを部分的に参照した．

3. 因子評点の推定
   ここでは，Bartlett によって与えられた最小 2 乗法を適用して因子評点を推定する方法を採用した．これについては，文献 [1] および [2] に詳しい．なお，各分野での因子分析の適用例については [3] および [1] の巻末で挙げられた諸文献に詳しい．

【 使用方法 】
概要で述べたように，本プログラムは，次の 2 つの実行形式プログラムより成り立っている．

- Factor Analysis – Estimation of Loading Matrix
  実行形式プログラム FACEST
- Factor Analysis – Orthogonal and Oblique Rotations, Estimation of Factor Scores
  実行形式プログラム FACROT

それぞれのプログラムの使用法を以下に述べる．

FACEST

1. ジョブ制御文

```
//A79999A1 JOB CLASS=A
//  EXEC GO,PROG=FACEST
//GO.SYSGO DD *
     データ 1 行目
     データ 2 行目
     データ 3 行目
           :
//GO.FT01F001 DD DSN= データセット名  <-- 必要に応じ指定
//GO.FT02F001 DD DSN= データセット名  <-- 必要に応じ指定
//
```

2. 入力データ

(a) 一行目：FORMAT(3(1X,I1))
順番に IN, IP, IOUT を入力する．IN, IP, IOUT はそれぞれ対応する桁の値によって以下のような意味を持つ．

IN： 入力データの区別を表す (2 桁目)．

0 – 標本分散共分散行列を入力することを示す．
1 – 原データを入力することを示す．
2 – 原データをデータセットから入力することを示す．

IP： NP, NK と求められた因子負荷行列の出力を示す (4 桁目)．

0 – 結果を出力する．
2 – 結果を出力すると共にデータセットに出力することを示す．

IP=2 を指定した時，出力されるデータは FACROT の入力データとすることができる．なお，IP=2 と指定した時は機番 1 に対して

```
//GO.FT01F001 DD DSN=&WORK1,UNIT=WORK,DISP=(,PASS),
//    SPACE=(TRK,(10,10),RLSE)
```

または

```
//GO.FT01F001 DD DSN= データセット名,DISP=SHR
```

等の DD 文をデータの後に挿入しなければならない．

IOUT：原データを入力した場合，その原データをデータセットに出力するか否かを指定する．

0 – データセットに出力しない．
1 – データセットに出力する．

IOUT=1 と指定した場合データセットに出力された原データは，再び本プログラムの入力データとすることができる．(この時 IN=2, IOUT=0 とする) ま

たは，FACROT において因子評点を求める場合の入力データとすることもできる．なお，この時機番 2 に対して

```
//GO.FT02F001 DD DSN=&WORK2,UNIT=WORK,DISP=(,PASS),
//     SPACE=(TRK,(10,10),RLSE)
```

または，

```
//GO.FT02F001 DD DSN= データセット名,DISP=SHR
```

等の DD 文をを挿入しなければならない．

(b) 二行目：`FORMAT(2I4,I6,2F10.0)`
順番に NP, NK, ND, EPS, EX を入力する

NP： 変量 (テスト項目) の数 p を与える (1 ~ 4 桁).
$2 \leq NP \leq 50$.

NK： 共通因子数 k を与える (5 ~ 8 桁).
$1 \leq NK < NP$, $(NP - NK)^2 - (NP + NK) \geq 0$.

ND： IN=1 または 2 の時，入力するデータの組数 n を与える (9 ~ 14 桁).
$NP < ND \leq 999999$.

EPS：固有値，固有ベクトルを求める場合の収束判定値，逆行列を求める場合のピボットの零判定値を与える (15 ~ 24 桁)．
$0 < EPS < 1.0$. 通常は $10^{-4} \sim 10^{-5}$ 程度を与える．

EX： improper solution の境界を与える。即ち、残差分散 Ψ の対角要素 $\psi_i(i = 1, 2, \cdots, NP)$ を反復計算で求める場合，$\psi_i \leq EX$ となった場合 improper solution として処理する．$(\psi_i > 0)$. 標準的には EX=0.005 程度の値を与える (25 ~ 34 桁).

これらのデータを与える場合は，右寄せでなければならない．(Fortran のデータはブランクの桁を 0 とみなすので)

(c) 三行目以降：`FORMAT(8F10.0)`
一行目の IN の値に応じて次のようにデータを与える．

- IN=0 の場合
  標本分散共分散行列または標本相関行列 R の下三角行列を次のように与える．

$$\begin{pmatrix} r_{11} & & & & \\ r_{21} & r_{22} & & & \\ r_{31} & r_{32} & r_{33} & & \\ \vdots & \vdots & \vdots & \ddots & \\ r_{p1} & r_{p2} & r_{p3} & \cdots & r_{pp} \end{pmatrix}$$

  $(r_{ij},\ j = 1, 2, \cdots, i)$ を $i = 1, 2, \cdots, p$ に対して与える．一行には 10 桁ずつ 8 個まで与える．$i > 8$ の場合は数行にわたって同じ形式で与える．i が変わる時は前のデータが途中で終わっていても新しいデータから与える．

- IN=1 の場合
  原データを与える．p 個のテスト項目に対する原データを一組ずつ与える．i

組目のデータを X_{ij} $(j = 1, 2, \cdots, p)$ とする．一行には 10 桁ずつ 8 個のデータを与え，$p > 8$ の場合は一組のデータを数行にわたって同じ形式で与える．新しい組みのデータを与える時は，前のデータが行の途中で終わっていても必ず新しい行から与える．

3. 出力データ形式

(a) 入力された，あるいは原データからの計算された標本分散共分散行列（または相関行列）が出力される．`*** RESULTS OF THE CALCULATIONS ***` に引き続いて計算結果が出力される．

- proper solution が得られ，共通因子数 k が危険率 5% で棄却されなかった場合次のメッセージが印刷される．
  - `NK=K IS ACCEPTED AT 5 PERCENT CRITICAL LEVEL`
  - `AFTER`（繰返し計算回数）`ITERATIONS`
  - 残差分散行列 Ψ ((3) 式) の対角要素 ψ_i $(i = 1, \cdots, p)$ が出力される．
  - 因子負荷行列 A が印刷される．
  - 共有性 $C_i = \sum_{j=1}^{k} a_{ij}^2$ が印刷される．
- proper solution が得られたが，因子数 k が危険率 5% で棄却された場合

  `NK=K IS REJECTED AT 5 PERCENT CRITICAL LEVEL`

  `TAKE NK=NK-1 OR NK=NK+1 AND START AGAIN`

  のメッセージが印刷され，先の場合と同様結果が参考として印刷される．この場合は共通因子数 k を設定しなおして再度計算を行い最適な k を定めることが必要である．
- improper solution が得られた場合

  `*** IMPROPER SOLUTION ***` と出力した後これを与えるテスト項目の番号を出力する．この場合はテスト項目の加減，入れ替え，共通因子数の吟味などを行い再度計算することが必要である．

(b) データセット出力形式

- 入力データとして IP=2 を指定した場合，テスト項目数 p，共通因子数 k，因子負荷行列 $A = (a_{ij})$ $(p \times k)$ をバイナリー形式で出力する．出力の順序は p, k に引き続いて，$a_{11}, a_{12}, \cdots, a_{1k}, a_{21}, a_{22}, \cdots, a_{2k}, \cdots, a_{pk-1}, a_{pk}$ の順に出力される．このときのデータセット上の形式は DD 文の指定によって決まる．これは，FACROT の入力データとすることができる．ただし，データセットが保存されるか，あるいは同一ジョブの後続のジョブステップで FACROT が実行され，しかもデータセットが PASS されている必要がある．
- 入力データとして IN=1 かつ IOUT=1 を指定した場合，ND 組の原データをバイナリー形式でデータセットに出力する．出力の順序は原データの入力順序と全く同じである．これは，再び本プログラムの入力データとすることができる．この時 IN=2 かつ IOUT=0 と指定する．ただし，データセットは保存されていなければならない．また，FACROT で因子評点を求める場合の入力データとすることもできる．この時，データセットは保存されているか，あるいは同一ジョブの後続のジョブステップで FACROT が実行され，しかもデータセットが PASS されている必要がある．

4. 制限事項
   すでに述べたように
   テスト項目数 NP は $2 \le \mathrm{NP} \le 50$，共通因子数 NK は $1 \le \mathrm{NK} \le \mathrm{NP}$,
   $(\mathrm{NP}-\mathrm{NK})^2-(\mathrm{NP}+\mathrm{NK}) \ge 0$, データの組数 ND は $\mathrm{NP} \le \mathrm{ND} \le 999999$,
   収束判定値 EPS は $0 < \mathrm{EPS} < 1.0$, improper solution の境界 EX は $0 < \mathrm{EX} < 1.0$.

5. エラー処理

   (a) 制限条件をはずれた場合, すなわち
   $\mathrm{NP} \le 1$ または $\mathrm{NP} > 50$,
   $\mathrm{NK} \le 0$ または $\mathrm{NK} \ge \mathrm{NP}$ または $(\mathrm{NP}-\mathrm{NK})^2-(\mathrm{NP}+\mathrm{NK}) < 0$,
   $\mathrm{ND} \le \mathrm{NP}$(ただし，IN=0 の時は無関係),
   $\mathrm{EPS} \le 0.0$ または $\mathrm{EPS} \ge 1.0$,
   $\mathrm{EX} \le 0.0$ または $\mathrm{EX} \ge 1.0$
   のいずれかを与えた時は，次のエラーメッセージを印刷して何も計算を行わない．

```
STOP THIS CALCULATION BECAUSE INPUT DATA ERROR
** NP=     NK=     ND=     EPS=     EX=
```

   (b) 固有値，固有ベクトルの計算 (サブルーチン SHQS) または逆行列の計算 (サブルーチン MINVS) の中で解が求まらなかった時は，次のエラーメッセージを印刷して計算を打ち切る．

```
***  ILL CONDITION IN SHQS * ILL=    **
```

   この時は，1 つには標本分散共分散行列（または標本相関行列）に問題があると考えられる．もう 1 つには収束条件が厳しすぎたことも考えられる．

FACROT

1. ジョブ制御文

```
//A79999A1 JOB CLASS=A
//  EXEC GO,PROG=FACROT
//GO.SYSGO DD *
     データ 1 行目
     データ 2 行目
     データ 3 行目
          :
//GO.FT01F001 DD DSN=データセット名   <-- 必要に応じ指定
//GO.FT02F001 DD DSN=データセット名   <-- 必要に応じ指定
//
```

2. 入力データ形式

   (a) 一行目：FORMAT(4(1X,I1))
   順に IO, JO, INP1, INP2 を入力する．IO, JO, INP1, INP2 はそれぞれ対応する桁の値によって次のような意味を持つ．

IO： 中間結果の出力を指示する (2 桁目).

0 – 中間結果を出力しない．
1 – 中間結果を出力する．

JO： 因子評点を求めるか否かを指示する (4 桁目).

0 – 因子評点を求めない．
1 – 因子評点を求める．

INP1： テスト項目数，共通因子数，因子負荷行列の入力方法を指示する (6 桁目).

0 – データで入力する．
1 – データセットから入力する (機番 1 のデータセット).

INP1=0 の場合次行より，テスト項目数，共通因子数，因子負荷行列を記述する．INP1=1 の場合，データは FACEST で作られたものとして入力している．また，このとき

```
//GO.FT01F001 DD DSN=&WORK1,UNIT=WORK,DISP=(OLD,DELETE)
```

または

```
//GO.FT01F001 DD DSN= データセット名,DISP=SHR
```

等の DD 文をデータの後に挿入しなければならない．

INP2： 因子評点を決める場合 (JO=1)，原データの入力方法を指示する (8 桁目).

0 – データで入力する．
1 – データセットから入力する (機番 2 のデータセット).

また，原データは一時的に機番 2 のデータセットに書き出し，後の計算で使っているので

```
//GO.FT02F001 DD DSN=&WORK1,UNIT=WORK,
//    SPACE=(TRK,(10,10),RLSE)
```

等の DD 文をデータの後に挿入しなければならない．INP2=1 の指定した場合，原データは FACEST で作られたものとして入力している．また，このとき

```
//GO.FT02F001 DD DSN=&WORK1,UNIT=WORK,DISP=(OLD,DELETE)
```

または

```
//GO.FT02F001 DD DSN= データセット名,DISP=SHR
```

等の DD 文をデータの後に挿入しなければならない．なお， JO=0 のとき， INP2 は意味をもたない．

(b) 二行目：FORMAT(2I4)

順に NP, NK を入力する．

NP – テスト項目数 p を与える (1 ~ 4 桁).

$2 \leq NP \leq 50$

NK – 共通因子数 K を与える (5 ~ 8 桁)．

$$1 \le NK < NP, (NP - NK)^2 - (NP + NK) \ge 0.$$

この行は INP1=1 の場合は不要である．

(c) <u>三行目以降：`FORMAT(6E12.5)`</u>
因子負荷行列 $A = (a_{ij})$ $(p \times k)$ を次のように与える．i 行目の要素 $a_{i1}, a_{i2}, \cdots, a_{ik}$ を一組として $i = 1, 2, \cdots, p$ に対して与える．1 行には 6 個のデータを与え，$k > 6$ の場合は数行にわたって与える． i が変わる時はそれ以前のデータが行の途中で終わっていても新しい行から与える．INP1=1 の場合，これらのデータは不要である．

(d) <u>因子評点を求める場合</u>
`FORMAT(I6)` で原データの組数 ND を与える．JO=1 であれば，INP2 の値にかかわらず，このデータは用意しなければならない．JO=0 の時は不要である．

(e) <u>JO=1 かつ INP2=0 の場合</u>
`FORMAT(8F10.0)` で原データを与える．原データの与え方は FACEST と同じ．JO=1 かつ INP2=0 の場合に用意しなければならない．

3. 出力データ形式

(a) 次のタイトルを印刷後，直交回転の結果が印刷される．

```
*** ORTHOGONAL ROTAION BY KEISERS VARIMAX METHOD ***
```

- IO=1 を指定した場合，巡回の中間結果が出力される．
  `INTERMEDIATE LOADING MATRIX(CYCLE 巡回の回数 )`
- 次のメッセージを印刷した後回転の最終結果が出力される．
  `*** INTERATION IS CONVERGED 巡回回数 CYCLES ***`
  - 因子負荷行列 $\hat{G}$ が出力される．
  - 共有性 $C_i = \sum_{j=1}^{k} g_{ij}$ $(i = 1, 2, \cdots, p)$ が出力される．
  - 変換行列 T が出力される．

(b) 次のタイトルを印刷して斜交回転の結果を出力する．

```
***** OBLIQUE ROTATION BY KEISER-DICKMAN METHOD *****
```

- IO=1 の場合，巡回の中間結果を出力される．
  `KEISER-DICKMAN SOLUTION(CYCLE 巡回の回数 )`
  - `REFERENCE STRUCTURE` と出力し巡回途中の参考構造解行列 $W = (w_{ij})$ $(p \times k)$ が出力される．
  - `TRANSFORMATION MATRIX` と出力し巡回途中の変換行列 $T = (t_{ij})$ $(k \times k)$ を出力する．

- `*** INTERATION IS CONVERGED` 巡回回数 `CYCLES ***` と印刷した後，結果を出力する．
  - 斜交回転後の参考因子構造行列とそれへの変換行列を出力する．

    `KEISER-DICKMAN SOLUTION REFERENCE STRUCTURE`
    （参考因子構造行列）
    `TRANSFORMATION MATRIX`
    （変換行列）

  - `FACTOR CORRELATION MATRIX` を出力した後，因子間相関行列 $(k \times k)$ を出力する．
  - `MATRIX OF ANGLE FOR EACH FACTOR` を出力した後，因子軸間角度行列 $(k \times k)$ を出力する．
  - `FACTOR STRUCTURE(PRIMARY)` を出力した後，根元因子構造行列 $(p \times k)$ を出力する．
  - `TRANSFORMATION MATRIX OF FACTOR STRUCTURE` を出力した後，変換行列 $(k \times k)$ を出力する．
  - `FACTOR PATTERN(PRIMARY)` を出力した後，根元因子パターン行列 $(p \times k)$ を出力する．
  - `TRANSFORMATION MATRIX OF FACTOR PATTERN` を出力した後，変換行列 $(k \times k)$ を出力する．
- JO=1 の場合，それぞれ直交回転，斜交回転の出力結果の後に各々の場合について因子評点係数行列 $C = (c_{ij})$ $(p \times k)$ と各個体の因子評点ベクトル $S = (s_i)$ $(i = 1, \cdots, k)$ が出力される．
  - `A COEFFICIENT MATRIX OF FACTOR SCORES` と印刷した後に行列 C が出力される．
  - `FACTOR SCORES FOR INDIVIDUALS` と印字した後に ND 個の各個体についてベクトル S が出力される．

4. 制限事項

このプログラムは，FACEST の計算結果として得られた因子負荷行列をもとに計算を進めているが，この因子負荷行列は，FACEST を繰り返し実行させて十分吟味されたものであることを前提としている．また，すでに述べたように次の制限がある．
テスト項目数 NP は $2 \leq \mathrm{NP} \leq 50$
共通因子数 NK は $1 \leq \mathrm{NK} \leq \mathrm{NP}, (\mathrm{NP}-\mathrm{NK})^2 - (\mathrm{NP}+\mathrm{NK}) \geq 0$
原データ数 ND は $\mathrm{NP} < \mathrm{ND} \leq 999999$

5. エラー処理

- 制限条件をはずれた場合すなわち
  $\mathrm{NP} \leq 1$ または $\mathrm{NP} > 50$
  $\mathrm{NK} \leq 0$ または $\mathrm{NK} \geq \mathrm{NP}$ または $(\mathrm{NP}-\mathrm{NK})^2 - (\mathrm{NP}+\mathrm{NK}) < 0$
  のいずれかを与えた時は，次のエラーメッセージを出力して何も計算を行わない．

```
STOP THIS CALCULATON BECAUSE INPUT DATA ERROR
**  IO=   JO=   INP1=   INP2=    NP=    NK=
```

- ND $\leq$ NP を与えた場合，次のエラーメッセージを出力して計算を打ち切る．

```
STOP THIS CALCULATION BECAUSE ND.LE.NP
```

【備考】

1. このプログラムでは，センターのライブラリとして登録されている次のルーチンを使用している．

   FACEST – JACOBS, MINVS
   FACROT– MINVS

2. FACEST の計算結果をデータセットに出力，その結果を FACROT で入力する場合，次の二通りがある．

   (a) 別ジョブで，それぞれに計算を行う場合
   (b) 同一ジョブ内で，一度に計算を行う場合

   それぞれの場合で使い方が異なるので次に説明する．

   別ジョブで，それぞれに計算を行う場合，データセットは保存されていなければならない．従って，FACEST で結果をデータセットに出力する前にデータセットを ALLOCATE しておかなければならない．

```
//A79999A1 JOB CLASS=A
// EXEC GO,PROG=FACEST
//GO.SYSGO DD *
 0 2 1
  ....
/*
//GO.FT01F001 DD DSN=A79999A.TMP.DATA,DISP=SHR
//
```

   ここでは IP=2 で機番 1 の既存データセットに出力している．

```
//A79999A1 JOB CLASS=A
// EXEC GO,PROG=FACROT
//GO.SYSGO DD *
 0 0 1
  ....
/*
//GO.FT01F001 DD DSN=A79999A.TMP.DATA,DISP=SHR
//
```

ここではINP=1で機番1のデータセットから入力している．原データをデータセットに出力し，FACROTで入力する場合も同様にできるが，この場合データセットは機番1のデータセットとは別のものを確保して使う必要がある．次に，同一ジョブ内で，一度に計算を行う場合，同一ジョブ内で計算を終えて，その結果を後で利用することがなければ一時データセットを使っても良い．

```
//A79999A1 JOB CLASS=A
// EXEC GO,PROG=FACEST
//GO.SYSGO DD *
 1 2 1
  ....
/*
//GO.FT01F001 DD DSN=&WORK1,DISP=(,PASS),
//   SPACE=(TRK,(10,10),RLSE)
//GO.FT02F001 DD DSN=&WORK2,DISP=(,PASS),
//   SPACE=(TRK,(10,10),RLSE)
//*
// EXEC GO,PROG=FACROT
//GO.SYSGO DD *
 0 1 1 1
  ....
/*
//GO.FT01F001 DD DSN=&WORK1,DISP=(OLD,DELETE)
//GO.FT02F001 DD DSN=&WORK2,DISP=(OLD,DELETE)
//
```

IN=1, IOUT=1で原データを入力し，機番2のデータセットに出力している．IP=2で機番1のデータセットに出力することを指示．JO=1, INP2=1で原データを機番2のデータセットから入力することを指示．INP1=1で機番1のデータセットから入力することを指示. `DISP=(,PASS)`で後続のジョブステップにデータを引き継ぐ．

【注意事項】

- 本ライブラリを用いて得られた結果を公表する場合は，プログラム名と開発者名を明記すること．

【参考文献】

[1] 「因子分析法通論」 浅野 長一郎, 共立出版社.
[2] 「Factor Analysis as a Stutistical Method」 Lawley-Maxwell.
（邦訳：「因子分析法」丘本正監 訳, 日科技連.）
[3] 「特集 - 因子分析」 数理科学, 1971, ダイヤモンド社.
[4] 「因子分析は有用か」 津村 善郎, 科学, 1971, Vol.41, No.8, pp.437–441, 岩波書店.
[5] 「Some contributions to Maximum Likelihood Factor Analysis」 Joreskog,K,G, Psychometrika 1967, Vol.32, No.4, pp.443–482.

6.5 PCAJCB, PCAQR

- 主成分分析 – JACOBI 法 (Principal Component Analysis – Jacobi Method)
- 主成分分析 – QR 法 (Principal Component Analysis – QR Method

プログラム形式	コンプリートプログラム
作成者 作成年月日	柳川 尭，工藤 奈津子 (九州大学理学部) 1973 年 2 月
改訂者 改訂年月日	柳川 尭，田中 靖子 (九州大学理学部) 1979 年 1 月
ソースの公表	`'QS.LIBSC(G6PCAJCB)'` `'QS.LIBSC(G6PCAQR)'`(MSP)
使用言語	Fortran
使用 OS	MSP(M1800)

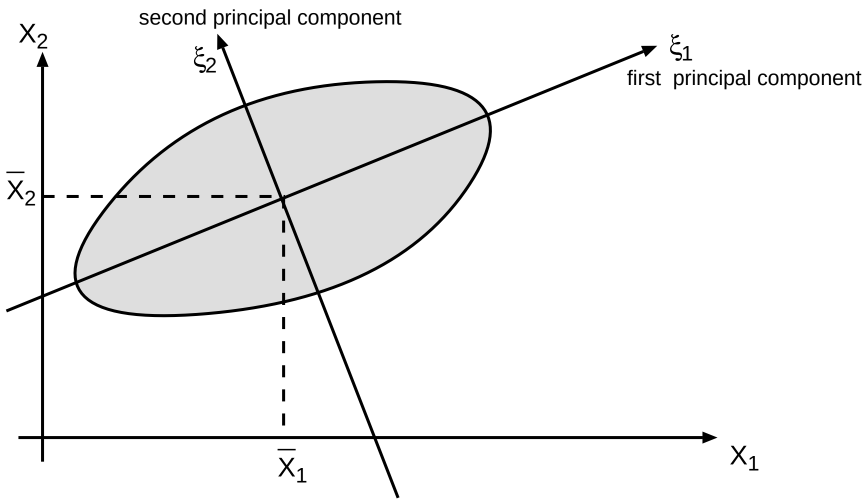
【 概要 】

(1) 目的

n 個のサンプルについての k 個の特性値 (k 変量) のデータは，k 次元空間に散布する n 個の点とみなすことが出来る．しかしながら, 4 次元以上の空間の点の分布状態を直観的につかまえることはきわめて難しい．そこで，k 個の特性値間の相関をうまく利用してなるべく情報量の損失が少なくなるように k 次元空間を m ($m < K$) 次元空間に射影する方法が工夫されており, 主成分分析法とよばれている．

主成分分析法の考え方を簡単な例で示そう．下図は n 個のサンプルについて X_1, X_2 という二つの特性値がはかられた結果を示している．この時，図の様に座標変換を行い新しい座標軸を ξ_1, ξ_2 とすると

1. この n 個のデータについて ξ_1 と ξ_2 との間の相関はほとんど 0 になる．
2. 2 次元平面 (X_1, X_2) 上で点のバラツキの大部分は ξ_1 軸上での変動に帰着し，ξ_2 軸上での変動はほとんど無視出来るほどに小さくなることが期待される．



このような ξ_1, ξ_2 のことを第 1 主成分, 第 2 主成分とよぶ．図で楕円が十分偏平なら（即ち相関係数の絶対値が大のとき）我々は ξ_2 軸上の全変動をまったく無視して ξ_1 軸上に射影された n 個の点の変動のみに注目しても大きな誤りはおかさないであろう．この時, 2 次元が 1 次元に減らされたことになる．

このように主成分分析は基本的には座標軸の変換 (平行移動, 回転) であるといえる．なお, ここで述べる様な主成分分析法を成分分析, または成因分析 (Component analysis) とよび, 第 1 主成分のみに注目して統計解析を行う場合を主成分分析とよぶ場合があるので注意されたい．

(2) 計算法

n 個のサンプルについての k 個の特性値のデータを

$$\mathbf{X_1} = \begin{bmatrix} X_{11} \\ X_{21} \\ \vdots \\ X_{k1} \end{bmatrix}, \quad \mathbf{X_2} = \begin{bmatrix} X_{12} \\ X_{22} \\ \vdots \\ X_{k2} \end{bmatrix}, \cdots, \quad \mathbf{X_n} = \begin{bmatrix} X_{1n} \\ X_{2n} \\ \vdots \\ X_{kn} \end{bmatrix}$$

とする．このとき, 標本平均, 標本分散共分散行列, 標本相関行列は次式で与えられる．

- 標本平均：

$$\bar{\mathbf{X}} = \frac{1}{n}\sum_{i=1}^{n}\mathbf{X}_i = \begin{bmatrix} \bar{X}_1 \\ \bar{X}_2 \\ \vdots \\ \bar{X}_k \end{bmatrix}, \qquad \bar{X}_i = \frac{1}{n}\sum_{\alpha=1}^{n} X_{i\alpha}, \quad i = 1, 2, \cdots, k.$$

- 標本分散共分散行列：

$$S^2 = \frac{1}{n}\sum_{\alpha=1}^{n}(\mathbf{X}_\alpha - \bar{\mathbf{X}})(\mathbf{X}_\alpha - \bar{\mathbf{X}})^T = \begin{bmatrix} S_{11} & S_{12} & S_{13} & \cdots & S_{1k} \\ & S_{22} & S_{23} & \cdots & S_{2k} \\ & & S_{33} & \cdots & S_{3k} \\ & & & \ddots & \vdots \\ & & & & S_{kk} \end{bmatrix},$$

$$S_{ij} = \frac{1}{n}\sum_{\alpha=1}^{n}(X_{i\alpha} - \bar{X}_i)(X_{j\alpha} - \bar{X}_j).$$

- 標本相関行列：

$$R = \begin{bmatrix} 1 & \gamma_{12} & \gamma_{13} & \cdots & \gamma_{1k} \\ & 1 & \gamma_{23} & \cdots & \gamma_{2k} \\ & & 1 & \cdots & \gamma_{3k} \\ & & & \ddots & \vdots \\ & & & & 1 \end{bmatrix}, \qquad \gamma_{ij} = \frac{S_{ij}}{\sqrt{S_{ii}S_{jj}}},$$

(但し, S^2, R は対称行列)

標本分散共分散行列 S^2 の固有値を大きい方から $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_k$ とする．また, λ_i に対応する固有ベクトルを $\gamma_i = (\gamma_{1i}, \gamma_{2i}, \cdots, \gamma_{ki})^T$ $(i = 1, 2, \cdots, k)$ とする．この時, 第 1 主成分, 第 2 主成分，$\cdots$，第 k 主成分をそれぞれ $\xi_1, \xi_2, \cdots, \xi_k$ とすると, $\xi_1, \xi_2, \cdots, \xi_k$ は, 次式で与えられる．

$$
\begin{array}{rclcrcccl}
\xi_1 &=& \gamma_{11}(X_1 - \bar{X}_1) &+& \gamma_{21}(X_2 - \bar{X}_2) &+& \cdots &+& \gamma_{k1}(X_k - \bar{X}_k) \\
\xi_2 &=& \gamma_{12}(X_1 - \bar{X}_1) &+& \gamma_{22}(X_2 - \bar{X}_2) &+& \cdots &+& \gamma_{k2}(X_k - \bar{X}_k) \\
 & & & & \vdots & & & & \\
\xi_k &=& \gamma_{1k}(X_1 - \bar{X}_1) &+& \gamma_{2k}(X_2 - \bar{X}_2) &+& \cdots &+& \gamma_{kk}(X_k - \bar{X}_k).
\end{array}
$$

このとき, ξ_i の平均は 0, 分散は λ_i, ξ_i と ξ_j $(i \neq j)$ はお互いに無相関になることが示される．また, 関係式 $tr(S^2) = \lambda_1 + \lambda_2 + \cdots + \lambda_k$ が示され ($tr(S^2)$ とは S^2 の対角要素の和) X の全分散と ξ の全分散は等しくなる．これにより, ξ_i は k 次元空間のバラツキ全体の

$$\alpha_i = \frac{\lambda_i}{\lambda_1 + \lambda_2 + \cdots + \lambda_k} \times 100\%$$

を説明しているものとみなすことが出来る． α_i は ξ_i の寄与率とよばれている． $\xi_i, \xi_2, \cdots, \xi_m$ の寄与率は

$$\beta_m = \frac{\lambda_1 + \lambda_2 + \cdots + \lambda_m}{\lambda_1 + \lambda_2 + \cdots + \lambda_k} \times 100\%$$

で与えられる．普通は寄与率 β_m が 70 ~ 80% 程度になるように m を定め, m 個の主成分 $\xi_i, \xi_2, \cdots, \xi_m$ に注目する．この時上式より

$$
\begin{array}{rclcrcccl}
X_1 - \bar{X}_1 &=& \gamma_{11}\xi_1 &+& \gamma_{12}\xi_2 &+& \cdots &+& \gamma_{1m}\xi_m \\
X_2 - \bar{X}_2 &=& \gamma_{21}\xi_1 &+& \gamma_{22}\xi_2 &+& \cdots &+& \gamma_{2m}\xi_m \\
 & & & & \vdots & & & & \\
X_k - \bar{X}_k &=& \gamma_{k1}\xi_1 &+& \gamma_{k2}\xi_2 &+& \cdots &+& \gamma_{km}\xi_m.
\end{array}
$$

が得られ, $X_1, X_2, \cdots, X_k$ が新しい変量 $\xi_1, \xi_2, \cdots, \xi_k$ で説明されることになる．

今, $\xi_i^* = \xi_i/\sqrt{\lambda_i}$ により ξ_i を基準化しておくと上式から, $X_1, X_2, \cdots, X_k$ は平均 0, 分散 1 を持つ変量 $\xi_1^*, \xi_2^*, \cdots, \xi_k^*$ により

$$
\begin{array}{rclcrcccl}
X_1 - \bar{X}_1 &=& \gamma_{11}^*\xi_1^* &+& \gamma_{12}^*\xi_2^* &+& \cdots &+& \gamma_{1m}^*\xi_m^* \\
X_2 - \bar{X}_2 &=& \gamma_{21}^*\xi_1^* &+& \gamma_{22}^*\xi_2^* &+& \cdots &+& \gamma_{2m}^*\xi_m^* \\
 & & & & \vdots & & & & \\
X_k - \bar{X}_k &=& \gamma_{k1}^*\xi_1^* &+& \gamma_{k2}^*\xi_2^* &+& \cdots &+& \gamma_{km}^*\xi_m^*.
\end{array}
$$

と書ける． $\gamma_i^* = (\gamma_{1i}^*, \gamma_{2i}^*, \cdots, \gamma_{ki}^*)$ を基準化された固有ベクトル (Standardized eigenvector) とよぶ．このパターンは普通主成分の意味づけに使われる．各サンプル (個人) について k 個の特性値に対する $(\xi_1, \xi_2, \cdots, \xi_k)$ の値を個人得点 (individual scores) という．

[主成分分析に関する注意]

上で示されたように, 主成分分析の計算は分散共分散行列の固有値, 固有ベクトルをもとめることにほかならない．この場合, X の特性値の測定単位はすべて同一であることを仮定している．もしそれが相異なっている場合には主成分分析の正当性を主張することはむつかしい．し

かしながら, 人体測定学, 計量心理学, 計量経済学等では各特性値は次のように基準化されているのが普通である．

$$Y_{ij} = \frac{X_{ij} - \bar{X}_i}{\sqrt{\sum_{j=1}^{n}(X_{ij} - \bar{X}_i)^2}}, \quad i = 1, 2, \cdots, k, \quad j = 1, 2, \cdots, n.$$

このとき, 分散共分散行列 S^2 は相関行列 R でおきかえられ, R の固有値, 固有ベクトルを求めることになる．

上と同様にして,

$$\begin{array}{rclcclcccl}
Y_1 & = & \gamma_{11}^{\sharp}\xi_1^{\sharp} & + & \gamma_{12}^{\sharp}\xi_2^{\sharp} & + & \cdots & + & \gamma_{1m}^{\sharp}\xi_m^{\sharp} \\
Y_2 & = & \gamma_{21}^{\sharp}\xi_1^{\sharp} & + & \gamma_{22}^{\sharp}\xi_2^{\sharp} & + & \cdots & + & \gamma_{2m}^{\sharp}\xi_m^{\sharp} \\
 & & & \vdots & & & & & \\
Y_k & = & \gamma_{k1}^{\sharp}\xi_1^{\sharp} & + & \gamma_{k2}^{\sharp}\xi_2^{\sharp} & + & \cdots & + & \gamma_{km}^{\sharp}\xi_m^{\sharp},
\end{array}$$

ここに, $(\gamma_{1i}^{\sharp}, \gamma_{2i}^{\sharp}, \cdots, \gamma_{mi}^{\sharp})$: R の基準化された固有ベクトル，$\xi_i^{\sharp}$: R の第 i 主成分．

本プログラムは user の指定に応じて, S^2 または R から主成分を求めることが出来る．一般に S^2 から求めた主成分と, R から求めた主成分は一致しないのみならず, 簡単な変換によって一方から他方を求めることも出来ないことに注意されたい．

【 使用法 】

本プログラムは, 固有値・固有ベクトルを求めるのに, JACOBI 法を用いる場合と, QR 法を用いる場合の 2 通りがある．

JACOBI 法は, QR 法に比べると若干精度はよいが, 変量 (特性値) の個数 k が大きくなると時間がかかりすぎる欠点を持っている．QR 法では, 固有値の大きさが著しく異なる場合には小さい固有値の精度が悪くなるが, JACOBI 法に比べると, 計算時間はかなり少なくてすむ．(k が小さい場合はあまり変わらないが, 大きくなるにつれて, かなり差がでる) 従って，k の大きさによってこれら 2 つを使い分けるとよいと思われる．

また, 固有値 $\leq EPS$ (後述の収束判定値) となる時は, 丸め誤差を考慮して固有値を機械的に $0.647713E-77$ と与えているので, 注意されたい．なお, 実行形式プログラム名は,

- JACOBI 法を用いる場合 — PCAJCB
- QR 法を用いる場合 — PCAQR

である．

1. 入力形式

```
//A79999A1 JOB CLASS=A
//  EXEC GO,PROG=PCAJCB     <--- または PCAQR を指定
//GO.SYSGO DD *
    データ 1 行目
    データ 2 行目
    データ 3 行目
        :
/*
//
```

2. 入力データ形式

(a) 一行目：FORMAT(3(1X,I1))
順番に IIN, ICAL, IOUT を入力する．IIN, ICAL, IOUT はそれぞれ対応する桁の値によって以下のような意味を持つ.

IIN：入力データの区別を表す (2 桁目)．

0 – 原データを入力する.
1 – 標本分散共分散行列を入力する.
2 – 標本相関行列を入力する.

ICAL：主成分を何から求めるかを指定する (4 桁目)．ただし IIN=2 の時はこの桁は指定しなくてよい．

0 – 標本相関行列から求める.
1 – 標本分散共分散行列から求める.

IOUT：IIN=0 の時，中間結果を出力するかどうかを指定 (6 桁目)．

0 – 出力しない．
1 – 出力する．

出力されるのは，標本平均，標本標準偏差， 標本分散共分散行列または標本相関行列.

(b) 二行目：FORMAT(I3,I6,F8.0)
順番に K, ND, EPS を入力する．

K：変量（特性値）の個数を示す（1 ~ 3 桁）．$2 \leq K \leq 100$.

ND：データの組み数を与える（4 ~ 9 桁）．$1 \leq ND \leq 999999$.

EPS：固有値，固有ベクトルを求める際の収束判定値を示す．（10 ~ 17 桁）．
$0 < EPS < 1.0$. 通常は 10^{-5} 程度を与えるが，K が大きい場合は多少 EPS を大きくするとよい．また，これらのデータは右よせで入力すること．

(c) 三行目以降：FORMAT(8F10.0)
一行目の IIN の値に応じて次のようにデータを与える．

- IIN=0 の場合
原データを与える．k 個の観測特性に対する原データを 1 組みずつ与える．i

組み目のデータを X_{ij} $(j = 1, 2, \cdots, k)$ とする．1 行には 10 桁ずつ 8 個のデータを与え，$k > 8$ の時は同じ形式で数行にわたって入力する．新しいデータの組みを与える時は，前のデータが行の途中で終っていても必ず改行し，新しい行からデータを与える．

- IIN=1 または IIN=2 の場合
  標本分散共分散行列または標本相関行列 A の下三角行列を次のように与える．

$$\begin{pmatrix} a_{11} & & & & \\ a_{21} & a_{22} & & & \\ a_{31} & a_{32} & a_{33} & & \\ \vdots & \vdots & \vdots & \ddots & \\ a_{k1} & a_{k2} & a_{k3} & \cdots & a_{kk} \end{pmatrix}$$

  $(a_{ij},\ j = 1, 2, \cdots, i)$ を $i = 1, 2, \cdots, k$ に対して与える．一行には 10 桁ずつ 8 個まで与える．$i > 8$ の場合は数行にわたって同じ形式で与える．i が変わる時は前のデータが途中で終わっていても新しいデータから与える．

3. 出力データ

   (a) まず, タイトルが出力される．

   (b) 固有値, 固有ベクトルの計算を標本分散共分散行列 (Sample Covariance Matrix) から行ったか, 標本相関行列 (Sample Correlation Matrix) から行ったかの区別が出力される．

   (c) 変量 (特性値) の個数 K (`NO.OF VARIABLES= `), 収束判定値 EPS (`EPS= `)
   データの組数 ND (`SAMPLE SIZES= `) が出力される．ただし, IIN> 0 の時は, データの組数は出力されない．

   (d) i. IIN=0 の時
      - IOUT=1 の時は
        標本平均 (Sample Mean), 標本標準偏差 (Sample Standard Deviation)
        ICAL=0 の時は Sample Covariance Matrix，
        ICAL=1 の時は Sample Correlation Matrix が出力される．
      - 次に演算の対象となる
        ICAL=0 の時, Sample Correlation Matrix
        ICAL=1 の時, Sample Covariance Matrix が出力される．

      ii. IIN $\neq 0$ の時
      - 入力行列が出力される．(`INPUT DATA`)
      - 次に演算の対照となる行列
        ICAL=0 の時, Sample Correlation Matrix
        ICAL=1 の時, Sample Covariance Matrix が出力される．

   (e) 以後が, 計算の結果の出力である旨が出力される．

```
***** RESULTS OF THE CALCULATION *****
```

(f) 番号 (NO．) n
固有値 (EIGENVALUE) λ_n
累積寄与率 (PERCENT) P_n
固有ベクトル (EIGENVECTOR) V_{ni} $(i = 1, \cdots, k)$
規準化された固有ベクトル (STANDARDIZED EIGENVECTOR)
SV_{ni} $(i = 1, \cdots, k)$ $(n = 1, 2, \cdots, k)$
が固有値の大きい順に, 次のような表形式で出力される．出力形式は, λ_n, P_n, V_{ni}, SV_{ni} は共に, 有効桁数 6 桁の E 行列である．

NO.	1	2	3	4	5	6	7	8
EIGENVECTOR	λ_1	λ_2	λ_3	λ_4	λ_5	λ_6	λ_7	λ_8
PERCENT	P_1	P_2	P_3	P_4	P_5	P_6	P_7	P_8
EIGENVECTOR	V_{11}	V_{21}	V_{31}	V_{41}	V_{51}	V_{61}	V_{71}	V_{81}
	V_{12}	V_{22}	$\vdots$					$\vdots$
	$\vdots$	$\vdots$	$\vdots$					$\vdots$
	V_{1k}	V_{2k}	V_{3k}	$\cdots$	$\cdots$	$\cdots$	$\cdots$	V_{8k}
STANDARDIZED	SV_{11}	SV_{21}	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	SV_{8k}
EIGENVECTOR	SV_{12}	SV_{22}	$\cdots$					$\vdots$
	$\vdots$	$\vdots$						$\vdots$
	SV_{1k}	SV_{2k}	$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$	SV_{8k}

4. 制限事項
変量 (特性値) の個数 k は $2 \leq k \leq 100$
データの組数 ND は $1 \leq ND \leq 999999$
収束判定値 EPS は $1 < EPS < 1.0$
ただし, もともとの有効桁数が 8 桁弱しかないので, 通常 10^{-5} 程度 (10^{-4} ~ 10^{-6} 位) の値を与えればよい．

5. エラー処理

- 制限条件をはずれた場合．即ち
  $k \leq 1$ または $K > 100$
  $ND \leq 0$ (ただし $\mathrm{IIN} \neq 0$ の時は無関係)
  $EPS \leq 0.0$ または $EPS \geq 1.0$
  のいずれかを与えた時は, 次のエラーメッセージを出力して, 何も計算を行わない

```
STOP THIS CAKCULATION BECAUSE INPUT DATA ERROR
** K=            ND=           EPS=
```

- 固有値, 固有ベクトルの計算 (サブルーチン JACOBS または SHQS) の中で収束せず, 解が求まらなかった時は, 次のエラーメッセージを印刷して計算を打ち切る．

```
**** ILL CONDITION IN XXXXXX ****
```

この時は, 1 つには Sample Covariance Matrix または Sample Correlation Matrix に問題がある事が考えられるので, これらの行列を検討してみる必要がある．また, 1 つには, 収束判定値 EPS の値が小さすぎた (条件がきびしすぎた) ことも考えられる．

6. 使用サブルーチン
このプログラムでは, センターのライブラリとして登録されている次のルーチンを使用している．

PCAJCB – JACOBS
PCAQR　– SHQS

【参考文献】

- 「因子分析法通論」浅野 長一郎, pp.19–48, 共立出版社.
- 「多変量解析の理論」 伊藤 孝一, pp.156–167, 培風館.
- 「多変量解析法・第３章」 奥野 忠一, 二科技連.
- 「因子分析」数理科学 (特集), 1971, ダイヤモンド社.
- 「Factor Analyasis as a Statisical Methods, Capter 4」Lawley Maxwell.
（邦訳：「因子分析法」丘本 正監 訳, 野本科学技術連盟.）

6.6　S

【概略】

S は AT&T のベル研で開発されたデータ解析とグラフィックスのための対話環境．user interface までも射程に入れた統計システム．グラフィックスとして Tektronix 端末と X Window 端末，および PostScript プリンターをサポートしている．これら 3 つは S のセッションの中から切替可能である

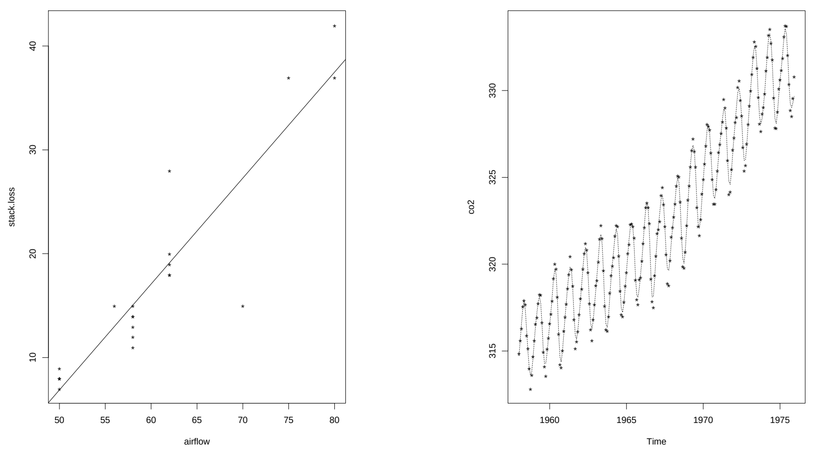
【処理環境】

- UXP/M1800
  command : `S(/usr/local/bin/S)`

【使用例】

- `X Window` を使用して `S` を起動する．ホスト名が `XXX` であるとする．

```
kyu-cc% setenv DISPLAY XXX:0.0
kyu-cc% S
> x11()
> .....
```



S graphics

【提供】

- AT&T

【参考文献】

- 「New S Language」R.A. Becker, J.M. Chambers and A.R. Wilks, Wadsworth & Brooks/Cole Advanced Books & Software, 1988
- 「UXP/M for the Working Scientists」佐藤 周行, 1993.

6.7 SALS

【概略】
SALS は自然科学，工学をはじめ農学，薬学，医学などさまざまな分野において，測定データを最小二乗法およびその改良法を用いて解析するための汎用プログラム・パッケージ．

プログラムは，サブルーチン SALS を頂点とした約 140 個のサブルーチンから構成されている．利用者は SALS を CALL する主プログラムとモデルを計算するためのサブルーチンを用意する必要がある．また，サンプルプログラム，ソースプログラムを以下のデータセット（区分編成）に公開している．

ソースプログラム – 'LIB.SALS.FORT'
サンプルジョブ – 'LIB.SALSTEST.CNTL'

【処理環境】

- MSP/M1800, MSP/VP2600
  cataloged procedure : FORT

【使用例】

- SALS を CALL した主プログラム'A79999A.TEST.FORT' を実行する．SALS データはジョブ制御文に与える．

```
//A79999A1 JOB CLASS=A
//  EXEC FORT,STEP=CLG,OPT=E
//FORT.SYSIN DD DSN=A79999A.TEST.FORT,DISP=SHR
//GO.SYSIN  DD *
PROBLEM       2.          3.          4.      1.          7.          3.
METHOD
GAUNEW
PARAMETER  4.
    :
ENDSALS
//
```

【SALS システムの処理方法】

- 線形最小二乗法
  - 修正 Gram-Schmit
  - Householder
  - 特異値分解
  - Cholesky
  - 固有値分解

- 非線形最小二乗法
  - Gauss-Newton
  - 修正 Marquardt
- 統計出力
  - 残差二乗和，標準偏差，情報量基準 AIC
  - パラメータ推定値の誤差，誤差行列
  - 残差グラフ，正規確率プロット

【注意事項】

- SALS は学術研究の信頼性確保と学術の普及，発展のために，ソースプログラムを公開しているが，無断でコピーして他の計算機に乗せることは禁じられている．
- SALS を利用した研究の論文発表には SALS の名前とともに，下記参考文献のいずれかを引用すること．
- MSP の `MANUAL` コマンドで SALS user's manual(英文) が NLP に出力可能．

【提供】

- SALS グループ

【参考文献】

- 「SALS 入門 – 実験データの解析」 小柳 義夫，中川 徹，東京大学出版会.
- 「最小二乗法による実験データ解析 プログラム SALS」小柳 義夫，中川 徹，東京大学出版会.
- 「Program System SALS for Nonlinear Least Squares Fitting in Experimental Sciences」 Totu Nakagawa and Yoshio Oyanagi, Recent Developments in Statistical Inference and Data Analysis, K.Matusita, editor, pp.221–225, North Holland Publishing Company, Amsterdam, 1980.

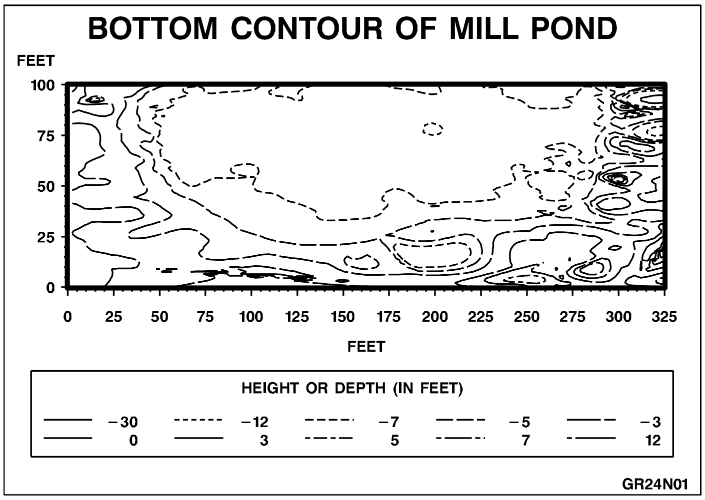
6.8 SAS

【概略】

SAS は Statistical Analysis System の略で，1966 年アメリカ・ノースカロライナ州立大学で，IBM 大型汎用コンピュータ用に開発された汎用統計解析システム．現在は大型汎用コンピュータ以外のコンピュータシステムへの移植が広く行なわれている．SAS の用途は，レポート作成，グラフ表示，データ検索，予測・シミュレーション，回帰分析などの高度な統計解析，時系列分析，オペレーションズ・リサーチ，品質管理，行列演算言語など，多岐にわたる．

【処理環境】

- MSP/M1800
  command : `SAS`, `RSAS`
  cataloged procedure : `SAS`



SAS graphics 1

【 センターで利用できる SAS プロダクト 】

- Base SAS
  SAS システムの基本プロダクト．DMS (Display Manager System) と呼ばれる統合化操作環境を提供する．DATA ステップ，レポート作成等の一般的なデータ管理機能をサポートする．

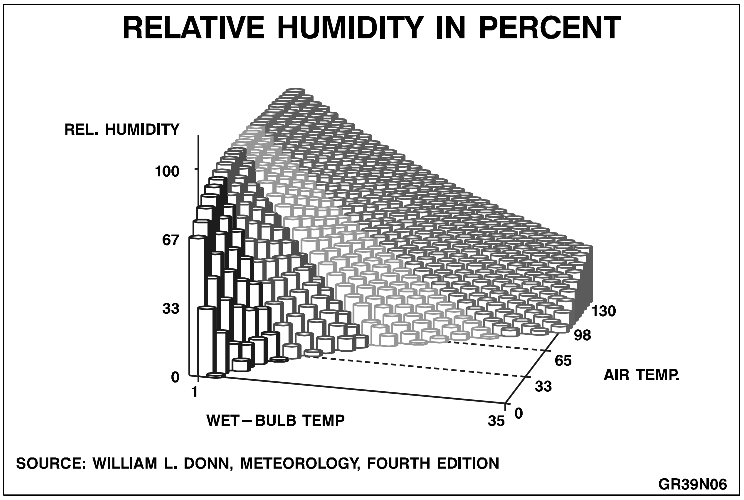
- SAS/STAT
  統計解析機能を提供する．回帰分析，分散分析，多変量分析，判別分析，クラスター分析，カテゴリカル解析，生存時間分析などを行なう．解析結果は SAS/GRAPH でグラフ化可能．

- SAS/IML
  行列を操作単位として線形代数に近い形で演算を行なう，インタプリター形式の対話型

行列演算プロダクト．一般逆行列，固有値分解，FFT 等の行列演算が関数・サブルーチン形式で組み込まれている．QP や時系列モデリング手法に最適である．

- SAS/ETS
  時系列・システムモデリングを提供するプロダクト．時系列予測，自己回帰モデル，ARIMA モデル，X11 季節調整，スペクトル分析等の時系列分析と，非線形方程式モデルをシミュレートする機能がある．

- SAS/GRAPH
  解析結果のグラフ化，資料の作成等を行なうグラフィックプロダクト．折れ線，棒線，円，3 次元グラフ，等高線，地図グラフ等をサポート．PostScript 形式の出力も可能．



SAS graphics 2

【注意事項】

- TSS で SAS を実行する場合は，LOGON 時に拡張領域を確保しておいた方がよい．

```
LOGON TSS A79999A S(50) ↵
```

【提供】

- SAS インスティチュート ジャパン

【参考文献】

- SAS Language : Reference, Version 6, First edition.
- SAS Procedures Guide, Version 6, Third edition.
- 「SAS system release 6.07J について」九州大学大型計算機センター広報, Vol.26, No.3, pp.195–220, 1993.

6.9 SPSSX

【概略】

SPSSX(Statistical Package for the Social Sciences/X) はアメリカ SPSS 社で開発された汎用統計解析パッケージ．データ変換機能と，各種統計プログラムのもつ解析機能を一つの大きな有機的システムとしてまとめ，各種のデータ処理や計算を一つのジョブの中で連続的に実行できることから，世界的に高い評価を受けているソフトウェア．

【処理環境】

- MSP/M1800
  command : `SPSSX`
  cataloged procedure : `SPSSX`

【主な SPSSX プログラム】

PEARSON CORR	: ピアソン積率相関係数	PARTIAL CORR	: 偏相関係数
ANOVA	: 多次元配置分散共分散分析	ALSCAL	: 多次元尺度法
ONEWAY	: 一元配置分散分析	PROXIMITIES	: 距離行列作成
REGRESSION	: 回帰分析	NONPAR CORR	: 順位相関係数
PROBIT	: プロビット分析	RELIABLITY	: 信頼性解析
FACTOR	: 因子分析	REGRESSION	: 重回帰分析
CLUSTER	: クラスター分析	DISCRIMINANT	: 判別分析
QUICK CLUSTER	: クイック・クラスター分析	MANOVA	: 多変量分散分析
HAYASI	: 林の数量化理論	BOX-JENKINS	: 時系列データ解析

【提供】

- SPSS 株式会社

【参考文献】

- 「SPSS/X user's guide 2nd ed」, SPSS Inc.
- 「新版 SPSS/X　I 基礎編」 三宅 一郎，山本 嘉一郎, 東洋経済新報社.
- 「新版 SPSS/X　II 解析編 1, 2」 三宅 一郎，山本 嘉一郎，垂水 共之，西脇 二一，石田 千代子，小野寺 孝義, 東洋経済新報社.
- 「利用の手引・ MSP コマンド編」九州大学大型計算機センター・ライブラリ室, 1993.
- 「利用の手引・バッチジョブ編」九州大学大型計算機センター・ライブラリ室, 1992.
- 「SPSSX 第 3.0 版について」 三宅 一郎，山本 嘉一郎, 小野寺 孝義，九州大学大型計算機センター広報, Vol.24, No.2, pp.127–158, 1991.

7

画像処理

ライブラリ名	機能	MSP	UXP	CM5
AVS	汎用可視化ソフトウェア			○
FIVIS	画像情報システム，富士通提供	○		
GDP/EX	図形編集出力システム	○		
GKS85	ISO 規格サブルーチン	○	○	
GRAPHMAN	対話式図形処理システム	○		
gnuplot	X Window, PostScript, Tektronix をサポート		○	
gs	日本語 PostScript の X Window 上エミュレータ		○	
PHIGS89	ISO 規格図形処理ソフトウェア	○		
PKS	図形印刷出力サブルーチン	○		
PBMPLUS	format 変換ツール		○	
Mathematica	X Window, PostScript をサポート		▲	
S	X Window, PostScript をサポート		○	
SAS/GRAPH	PostScript をサポート	○		
tgif+	X Window 上の作図ツール. 日本語版		○	
VOGLE	3 次元グラフィックスライブラリ		○	
XV	X Window 上の image file 表示ユーティリティ		○	

▲：バックエンドのワークステーション上の動作

Mathematica, SAS/GRAPH, S は強力なグラフィックス機能があり，下手な専門ソフトより使い勝手もいい．Mathematica, SAS/GRAPH, S に関してはそれぞれ「数式処理」，「統計処理」の項を参照．

7.1 AVS

【概略】
AVS(Application Visualization System) は科学技術分野向けの 3 次元グラフィックシステム．格子点のスカラー，ベクトルデータや非構造格子データ，ジオメトリ，イメージ，分子データ等を入力とし，各種ソフトウェア部品を組み合わせてネットワークに組み込むことにより，独自の可視化アプリケーションが作成できる．センター二階のワークステーション qvisa でのみの動作．また， CM5 を用い，バックグラウンドで処理も可能．

【処理環境】

- qvisa
  command : `avs`

【備考】

- `AVS` は計算化学，医療，構造力学，流体力学，気象学等の計算で得られたデータを変換し，高度なグラフィックス表示を実現する．操作は対話的にマウス操作によって行なう．
- 画像処理ソフト，写真等のイメージスキャナー出力， `paint` ソフト出力などのイメージデータの処理も可能．
- アニメーション，シミュレーション機能をサポート．

【提供】

- KUBOTA computer

【参考文献】

- AVS User's Guide

7.2 FIVIS

【概略】
FIVIS(Fujitsu Integrated Visual Infomation System) は富士通株式会社が開発した画像情報システム．次のソフトウェアがある．

IPEX
Image Processing EXpert system
画像データの入出力，画像処理，画像解析システム．

PROSID II
PROcessing System for Image Data II
リモートセンシング画像データ処理システム．

VIPL
Visual Information Processing Library
画像処理用サブルーチンライブラリ．

CGMS
Computer Graphics and Movie System
数値データの図形化，イメージ化するシステム．

【処理環境】

- MSP/M1800

【提供】

- 富士通株式会社

【参考文献】

- 「FACOM OS IV/F4 MSP CGMS 説明書」(78AR-3060), 富士通株式会社．
- 「FACOM OS IV PROSID II 説明書」(70AR-5630), 富士通株式会社.
- 「FACOM IPEX 使用手引書」(99AR-5030), 富士通株式会社.

7.3 GDP/EX

【概略】

GDP/EX(Graphic image Drawing Package/Extended) は各種の図形処理プログラムで作成した図形またはグラフを表示，編集，印刷，保存するユーティリティ．

PKS で作成されたグラフィックファイル (`GOUT(*)` 及び `KEEP` を指定したときに作成される) や EGRET/DRAW で作成された保存図形を端末に表示，あるいは NLP に出力する．

【処理環境】

- MSP/M1800
  command : `GDP`

【備考】

- `GDP/EX` は `member`，`page`，`graph` 単位で表示，回転，縮小，拡大，印刷等の操作ができる．
- プリンターへの出力は，`NLP`，`CLP` の `A4`，`B4` 版（縦横）が可能．

【提供】

- 富士通株式会社

【参考文献】

- 「OS IV GDP/EX 説明書 V10L20 用」(70SP-3651)，富士通株式会社．
- 「利用の手引・MSP コマンド編」九州大学大型計算機センター・ライブラリ室, 1993.
- 「PKS の利用法」垣内 良子，多久島 智，山寄 信広，九州大学大型計算機センター広報, Vol.26, No.6, pp.637–682, 1993.

7.4 GKS85

【概略】
GKS85 は国際標準化機構 ISO が 1985 年に制定した GKS(Graphical Kernel System) に準拠した図形処理基本ソフトウェア．Fortran から呼び出されるサブルーチンライブラリ．

【処理環境】

- MSP/M1800
  command：`GKS85`

【備考】

- 作成された図形は，メタファイルで保存される．形式は `CGM(IS 8632 Computer Graphics Metafile)` に準拠．
- 富士通提供の拡張機能として，日本語文字関連サブルーチンが追加されている．

【提供】

- 富士通株式会社

【参考文献】

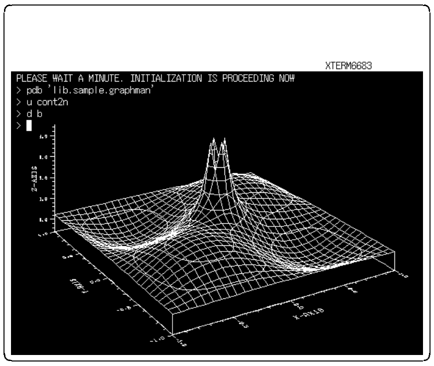
- 「富士通 GKS85 文法書 (FORTRAN77 編)」(99SP-4040)，富士通株式会社．
- 「FACOM GKS85 使用手引書 V10 用」(99SP-4080)，富士通株式会社．
- 「GKS 入門」九州大学大型計算機センター広報, Vol.22, No.5, pp.446–468, 1989.
- 「利用の手引・ MSP コマンド編」九州大学大型計算機センター・ライブラリ室, 1993.

7.5 GRAPHMAN

【概略】
GRAPHMAN(Graphic Man-machine System For Scientific Calculation Results) は図形処理を対話的に行うことで，研究者，技術者にとっての図形作画のための複雑な作業から解放することを目的として開発されたシステム．グラフィックディスプレイを使用して，あたかもソースプログラムのテキスト編集をするかのように，迅速かつ容易に図形編集および図形表示ができる．BCOPY データセットのプリンターへの出力は GOUTコマンドで行なう．

図形パターンは以下の通り．

- 関数 $y = f(x)$ の表示
- 2次元メッシュ図，流れ図
- 関数 $z = f(x, y)$ の等高線図，鳥瞰図
- 3次元物体の透視図
- 折れ線データ等の作成，編集



Graphman Graphic

【処理環境】

- MSP/M1800
  command : GRAPHMAN
  cataloged procedure : GRAPHMAN

【注意事項】

- 処理データが大きい場合は，GRAPHMAN で使用する変数領域 (Named Storage) の大きさを大きめに指定する．省略値は 80KB．

```
 READY
GRAPHMAN NS(2000) ↵     <--- 変数領域を 2,000KB 使用することを指定
```

【提供】

- 富士通株式会社

【参考文献】

- 「FACOM OS IV/F4 MSP GRAPHMAN 説明書」(78AR-5170), 富士通株式会社.
- 「GRAPHMAN を使ってみませんか !?」橋倉 聡，寺田 亮子，九州大学大型計算機センター広報, Vol.23, No.2, pp.67–101, 1990.
- 「利用の手引・ MSP コマンド編」九州大学大型計算機センター・ライブラリ室, 1993.

7.6 gnuplot

【概略】

GNUPLOT は command-driven interactive function plotting program. X Window，PostScript，Tektronix をサポート．使用方法は `/usr/local/doc/gnuplot.dvi` に参考文献が TEX の dvi ファイルとしてあるので，参照されたい．

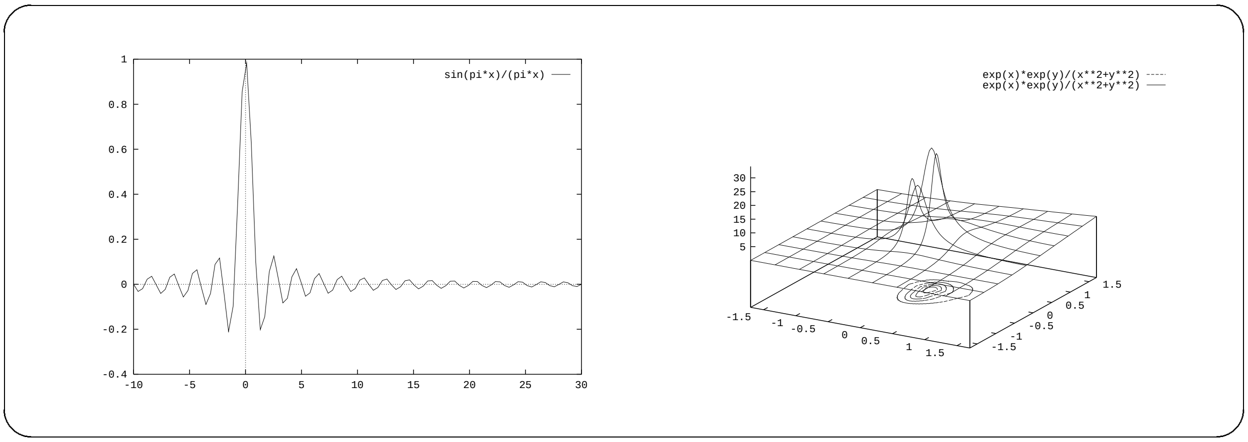
【処理環境】

- UXP/M1800
  command : `gnuplot(/usr/local/bin/gnuplot)`

【使用例】

- `GNUPLOT` で作図したデータを `PostScript` ファイルに保存する．

```
kyu-cc% gnuplot
gnuplot> set terminal postscript portrait
gnuplot> set outout ``file''
        :
     作図のコマンド
        :
gnuplot> q
```



gnuplot graphics

【提供】

- Gershon Kelley, etc.

【参考文献】

- 「GNUPLOT : An Interactive Plotting Program」Thomas Williams & Colin Kelley.

7.7 gs

【概略】

日本語 PostScript の X Window 上でのエミュレータ Ghostscript を起動する．

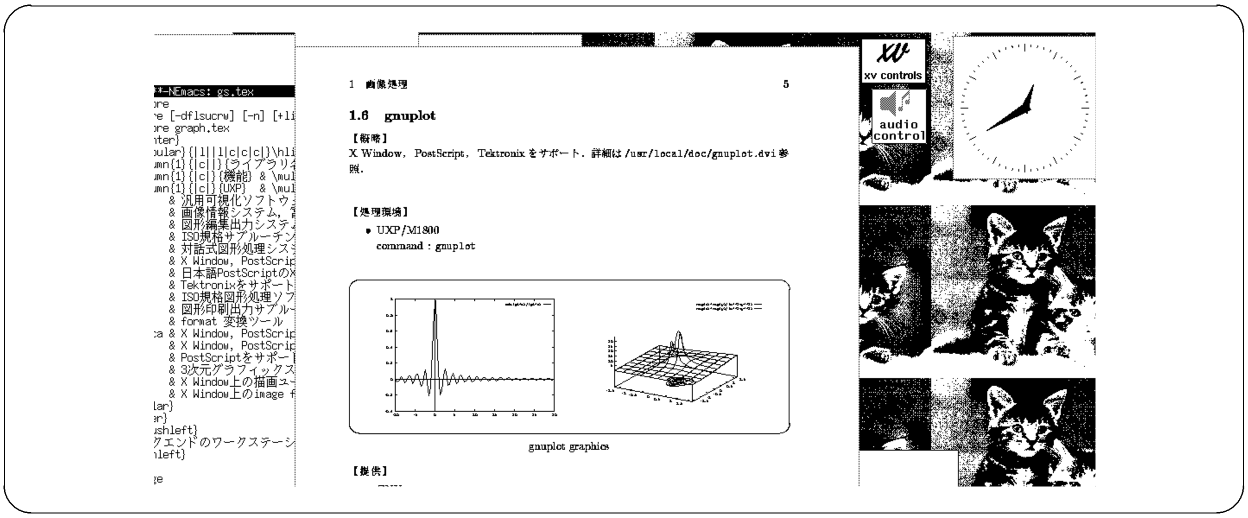
【処理環境】

- UXP/M1800
  command : `gs(/usr/local/bin/gs)`

【使用例】

- `gs` を起動し PostScript ファイル `tmp.ps` をエミュレートする．

```
kyu-cc% gs tmp.ps ↵
```



ghostscript sample（中央）

【提供】

- Aladdin Enterprise

【参考文献】

- 「UXP/M for the Working Scientists」佐藤 周行, 1993.

7.8 PHIGS89

【概略】
PHIGS(Programmer's Hierarchical Interactive Graphics System) は国際標準化機構 ISO の制定した図形処理の国際標準規格．

PHIGS は 3 次元図形処理応用プログラムの開発に利用されるソフトウェア．図形データの作成・編集，出力，図形データの保存・復元，対話入力などの豊富な機能がある．

【処理環境】

- MSP/M1800
  command : `PHIGS89`

【提供】

- 富士通株式会社

【参考文献】

- 「OS IV PHIGS89 使用手引書 V10L11 用」(70SP-4430), 富士通株式会社.
- 「PHIGS 概説 (I)」 佐藤 周行，九州大学大型計算機センター広報, VoL.25, No.5, pp.393–404, 1992.
- 「PHIGS 概説 (II)」佐藤 周行，九州大学大型計算機センター広報, VoL.25, No.6, pp.507–529, 1992.
- 「PHIGS 概説 (III)」佐藤 周行，九州大学大型計算機センター広報, VoL.26, No.3, pp.221–231, 1993.
- 「PHIGS 概説 (IV)」佐藤 周行，九州大学大型計算機センター広報, VoL.26, No.5, pp.555–562, 1993.

7.9 PKS

【概略】
PKS(Plotter Kernel System) は CALCOMP 製 HCBS(Host Computer Basic Software) と互換性のあるサブルーチンライブラリ．

PSP(Plotter Subroutine Library) 互換モードがあり，PSP を使った応用プログラムをそのまま PKS 下で実行することができる．

【処理環境】

- MSP/M1800
  command : `PKS`, `PKSCL`, `PKSGO`
  catalogued procedure : `PKS`

【注意事項】

- PKS はデバイスドライバーを持つので，直接出力装置に出力可能．ただし，画面に表示する場合はグラフィック・データセットに一旦データを出力し，GDP/EX を起動することによって画面に図形を表示する．

【提供】

- 富士通株式会社

【参考文献】

- 「富士通 PKS 文法書 V10 用」(99SP-7200)，富士通株式会社.
- 「FACOM PKS 使用手引書 V10 用」(99SP-7210)，富士通株式会社.
- 「PKS の利用法」垣内 良子，多久島 智，山嵜 信広，九州大学大型計算機センター広報, Vol.26, No.6, pp.637–682, 1993.

7.10 PBMPLUS

【概略】
グラフィックスシステムの出力フォーマット変換ツール．/usr/local/doc/pbmplus 下に document があるので参照されたい．

ツールは /usr/local/bin/pbmplus 下にある．変換は例えば

```
kyu-cc% giftoppm file1.gif > file2.ppm ↵
```

として行なう．

主な形式は次の通り．逆の変換も出来る場合がある．

- PBM(ビットマップ形式) への変換
  - Sun raster
  - X11 bitmap
  - MacPaint
  - X11 window dump
- PGM(モノクロ・グレイスケール形式) への変換
  - TIFF
  - PostScript image data
  - Encapsulated PostScript
  - X11 window dump
- PPM(24 ビットフルカラー形式) への変換
  - color Sun raster
  - GIF
  - Color Encapsulated PostScript
  - Color X11 window dump

【処理環境】
- UXP/M1800

【参考文献】

- 「UXP/M for the Working Scientists」佐藤 周行, 1993.

7.11 tgif+

【概略】

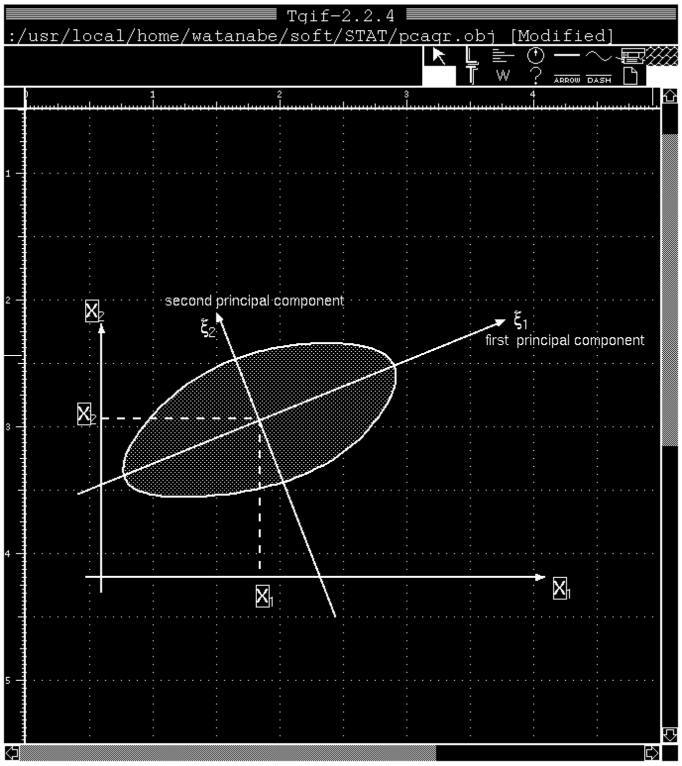
tgif+ はX window を用いて会話的に図形の作成・操作を行なう作図ツールである tgif を日本語化＋αしたもの．LaTeX で書かれた文書内に取り込むことが可能な形式の PostScript ファイル (encapsulated PostScript) を生成したりすることも可能 (カラーもサポート) ．

描画プリミティブとしては，テキスト，四角形，楕円，ポリライン，ポリゴン，楕円弧，角丸四角形，ビットマップがある X window dump(xwd) の結果は，ビットマップに変換することで図中に取り込める．

使用方法は `/usr/local/doc/tgifman` の下にマニュアルのソース (jLaTeX)，dvi があるので，参照されたい．

【処理環境】

- UXP/M1800
  command : `tgif+ (/usr/local/bin/X11/tgif+)`



tgif+ を用いての作図

【提供】

- William Chia-Wei Cheng & 松田 晃一

【参考文献】

- 「作図ツール tgif+ の使い方 – 日本語化＋αされた tgif – 」松田 晃一.
- 「UXP/M for the Working Scientists」佐藤 周行, 1993.

7.12 VOGLE

【概略】

VOGLE(A Very Ordinary Graphics Learning Environment) は GL ライクの 3 次元グラフィックスライブラリおよびそのユーティリティコマンド群. 2 次元，3 次元図形, アニメーションの生成, 表示が可能. X Window, PostScript, Tektronix をサポート.

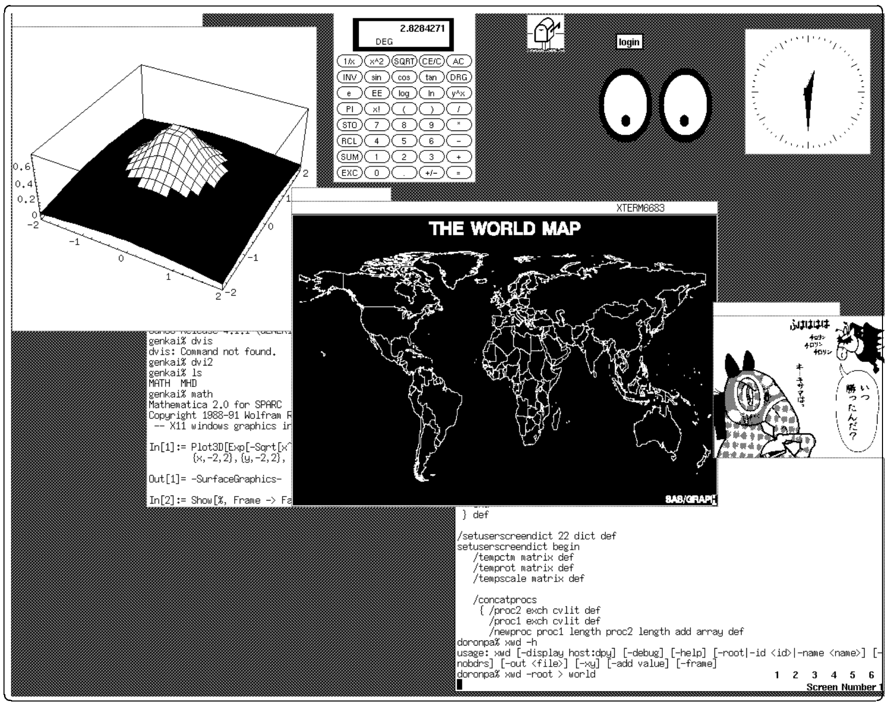
`/usr/local/doc/VOGLE` 下のドキュメントを参照．

【処理環境】

- UXP/M1800

【参考文献】

- 「UXP/M for the Working Scientists」佐藤 周行, 1993.



X Window 上での Graphics の表示

7.13 XV

【概略】
X Window 上で動作するイメージファイル (GIF,TIFF など) の表示ユーティリティ．Save 時の format は以下のスタイルが選択できる．

GIF	PBM(raw)
X11 Bitmap	PBM(ascii)
Sun Rasterfile	TIFF
PostScript	JPEG
PM	

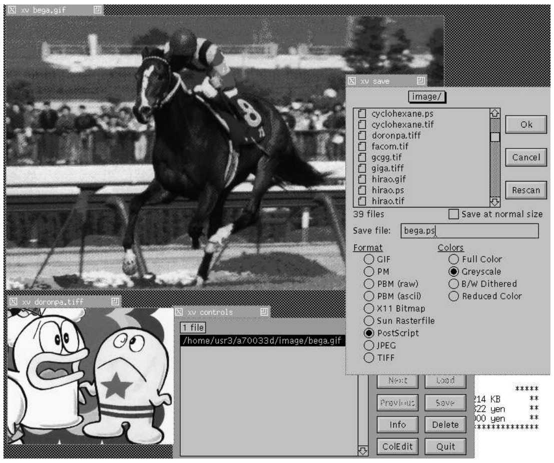
【処理環境】

- UXP/M1800
  command : `xv(/usr/local/bin/X11/xv)`

【使用例】

- `xv` を起動し `tiff` ファイル `tmp.tif` を表示する．

```
kyu-cc% xv tmp.tif ↵
```



xv での image data の表示例

【提供】

- John Bradley

【参考文献】

- 「UXP/M for the Working Scientists」佐藤 周行, 1993.

- 完全にメニュー形式なので，とりあえず起動してみれば分かる．

8

電子メール

1993 年 1 月現在，電子メールが到達可能な国は約 120 カ国である．紙の文書は，封筒に入れて宛先を書き，郵便局に渡すと，宛先をもとに郵便局が配送を行ない，相手に届けることができる．これと同じように，コンピュータのファイルとして作成した文書も，宛先を指定し，電子メール配送システムに渡せば，宛先をもとに電子メール配送システムが配送を行ない，相手に届けることができる．

ライブラリ名	機能	MSP	UXP
mailx	tty 系メールインターフェイス		○
MH	Emacs から起動可能		○
MHS	CCITT 勧告に基づく国際標準	○	
RMAIL	Emacs から起動する		○

電子メールには，手紙にはない特徴がある．

座ったままでよい　手紙の場合，机の前に座り，文字を書き，手紙を封筒に入れ，糊付けし，切手を貼り，ポストのあるところまで歩き，投函する．外が台風の場合や見知らぬ土地でポストの在処がわからない場合があることなどを考慮すると，相当な努力が必要になる場合がある．これに対し，電子メールは，これと同じことをするのに，コンピュータの前に座って指を動かすだけでよい．最初はどう指を動かすかわからずに苦労するかもしれないが，慣れてしまうと非常に楽である．

宛先が複数書ける　手紙に複数の宛先を書くとはできないが，電子メールの場合，宛先を複数書くことができ，同時に複数の人にメールを送信することができる．

コピーが容易　機械可読な文書は，紙の文書と比べて，コピーが非常に容易であるという特徴を持つ．したがって，他人の文章を引用することも容易である．しばしば文脈を無視し，一部の文章だけを引用した挙げ足取りが横行するのも電子メールの特徴である．ご用心．

文書以外を送受信できる　電子メールでバイナリーデータを送受信すれば，プログラム，画像，音声など，テキストに拘らない「データ」のやりとりが可能である．

その他　セキュリティ，配送の速さ，配送の確実さなども手紙とは異なる．

8.1 mailx

【概略】
対話型メッセージ処理システム．mailx は電子メールの送信，受信のための快適で柔軟な環境を提供する．メールを読み取るときには，`mailx` はメッセージの保存，削除，およびメッセージへの応答を容易にするためのコマンドを提供する．`mailx` によってメールを送信するときは，メッセージを入力しながら編集，見直し，およびその他の変更をすることが可能．

【処理環境】

- UXP/M1800
  command : `mailx(/usr/bin/mailx)`

【使用例】

- `a79999a` に予め作成したファイル
  `letter.txt` を送信する．

```
kyu-cc% mailx a79999a < letter.txt ↵
```

【提供】

- 富士通株式会社

【参考文献】

- `man mailx`
- 「UXP における電子メール入門（電子メール講習会資料）」九州大学大型計算機センター・ネットワーク室.
- 「UXP/M for the Working Scientists」佐藤 周行, 1993.

8.2 MH

【概略】
MH はメールハンドラーと呼ばれ，メールを整理する機能が充実しているため，mailx と並んで代表的なインターフェイスの一つとして広く使用されている．Emacs 上で MH を使うためのフロントエンドプロセッサ mh-e もある．

【処理環境】

- UXP/M1800
  command：`/usr/local/bin/mh` 下のコマンド群

【使用例】

- `Emacs` を通したメールの読み書きを行なう．

```
kyu-cc% emacs -f mh-rmail ↵
```

【注意事項】

- MH を利用する時はパスの先頭に
  `/usr/local/bin/mh` を追加すること．
- Emacs 内から MH を使う場合は，ホームディレクトリの下に`.mh_profile` があること．

【参考文献】

- 「UXP/M for the Working Scientists」佐藤 周行, 1993.
- 「UXP における電子メール入門（電子メール講習会資料）」九州大学大型計算機センター・ネットワーク室.

8.3 MHS

【概略】
国際規格 (CCITT 勧告 X.400) に基づく MHS(Message Handling System) 方式での電子メールシステム．MHS メールシステムは国際標準の電子メールで，現在 7 つの大型計算機センターと学術情報センターを結んで運用が行われている電子ニュース，電子メールのユーザー・インターフェイスとして，富士通株式会社提供の FINESシステムを利用している．

【処理環境】

- MSP/M1800
  comamnd : MHS

【注意事項】

- MHS システムより，BITNET，JUNET や海外とのメールの送受信が可能．
- 電子ニュースの検索，投稿もできる．

【提供】

- 富士通株式会社

【参考文献】

- 「MHS メールシステムの利用法」 原田 盛一，川嵜 正子，富山 実，竹生 政資，松延 栄治, 九州大学大型計算機センター広報, Vol.24, No.6, pp.580–625, 1991.
- 「富士通 FINES 使用手引書（電子メール，電子ニュースシステム）」(78AR-1670), 富士通株式会社．
- 「利用の手引・MSP コマンド編」九州大学大型計算機センター・ライブラリ室, 1993.

8.4 RMAIL

【概略】
Emacs の持つメール送受信機能．Rmail は起動されると，マシンのスプールから自分宛のメールを取り込み，ホームディレクトリにある `RMAIL` ファイルに格納する．

【処理環境】

- UXP/M1800

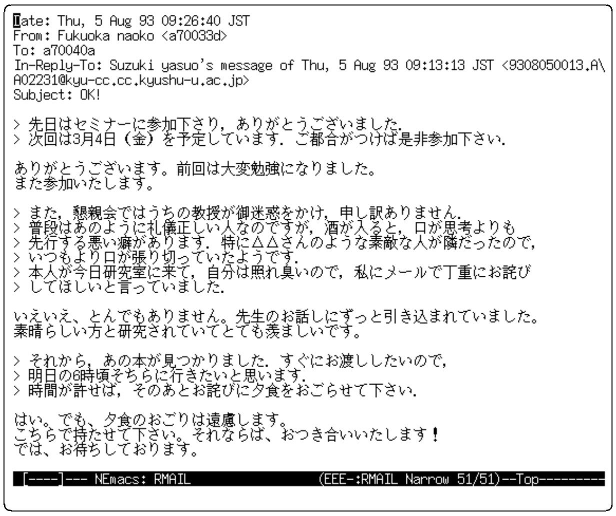
【使用例】

- `Emacs` を通したメールの読み書きを行なう．

```
kyu-cc% emacs -f rmail ⏎
```

【提供】

- Satoru Tomura



```
Date: Thu, 5 Aug 93 09:26:40 JST
From: Fukuoka naoko <a70033d>
To: a70040a
In-Reply-To: Suzuki yasuo's message of Thu, 5 Aug 93 09:13:13 JST <9308050013.A\
A02231@kyu-cc.cc.kyushu-u.ac.jp>
Subject: OK!

> 先日はセミナーに参加下さり，ありがとうございました．
> 次回は3月4日（金）を予定しています．ご都合がつけば是非参加下さい．

ありがとうございます。前回は大変勉強になりました。
また参加いたします。

> また，懇親会ではうちの教授が御迷惑をかけ，申し訳ありません．
> 普段はあのように礼儀正しい人なのですが，酒が入ると，口が思考よりも
> 先行する悪い癖があります．特に△△さんのような素敵な人が隣だったので，
> いつもより口が張り切っていたようです．
> 本人が今日研究室に来て，自分は照れ臭いので，私にメールで丁重にお詫び
> してほしいと言っていました．

いえいえ、とんでもありません。先生のお話しにずっと引き込まれていました。
素晴らしい方と研究されていてとても羨ましいです。

> それから，あの本が見つかりました．すぐにお渡ししたいので，
> 明日の6時頃そちらに行きたいと思います．
> 時間が許せば，そのあとお詫びに夕食をおごらせて下さい．

はい。でも、夕食のおごりは遠慮します。
こちらで持たせて下さい。それならば、おつき合いいたします！
では、お待ちしております。

[----]--- NEmacs: RMAIL          (EEE-:RMAIL Narrow 51/51)--Top--------
```

RMAIL を用いた電子メール（内容と ID は無関係）

【参考文献】

- 「UXP/M for the Working Scientists」佐藤 周行, 1993.
- 「UXP における電子メール入門（電子メール講習会資料）」九州大学大型計算機センター・ネットワーク室.

9

電子ニュース

MSP, UXP で電子ニュースシステムのサービスを行なっている．電子ニューストピックス毎に News Group に分けられる．現在 comp, sci, soc, rec などの国際的なニュースグループと日本国内で主に流れている fj ニュースグループが主なもの．九州大内部で流れている kyushu-u 下のニュースグループもある．

ニュースの読み書きは，以下のインターフェイスで行なう．

ライブラリ名	機能	MSP	UXP
MHS	ニュース読み / 投稿機能	○	
Rn	コマンド rn, Pnews		○
Nn	記事が選択的に読める		○
gnus	emacs から起動する		○

主なニュースグループはつぎの通り．

(1) kyushu-u（九州大学の話題; Kyushu University）

```
kyushu-u.general    ; 九州大学全体に関係する一般的な話題
kyushu-u.q-and-a    ; Ｑ＆Ａ，質問をすると誰かが回答してくれる（かもしれない）
kyushu-u.archives   ; 九州大学の anonymous ftp サーバで入手可能なソフト，データ情報
kyushu-u.cc         ; 大型計算機センターに関係する話題（質問，センターニュースなど）
kyushu-u.ec         ; 情報処理教育センターに関係する話題
kyushu-u.csce       ; 情報工学科に関係する話題
kyushu-u.cslabs     ; 情報工学科計算機研究グループに関係する話題
kyushu-u.is         ; 総合理工学研究科の情報システム学研究グループに関係する話題
kyushu-u.theorynet  ; ‘‘Theory Net’’ メーリングリストの情報を記事として掲載している
kyushu-u.test       ; 記事投稿のテストやシステム動作のテスト
kyushu-u.misc       ; 九州大学に関係するが上記のどれにも当てはまらない話題
```

(2) kudpc(京都大学大型計算機センターの話題; Kyoto University, Data Processing Center)

```
kudpc.consult       ; センターに対する相談など
kudpc.news          ; センターニュース
kudpc.announce      ; センターからのアナウンス
kudpc.general       ; センター全体に関係する一般的な話題
kudpc.bugs          ; システムやライブラリのバグに関する情報
```

(3) tains (東北大学の学内 LAN の話題; Tohoku university Academic/All-round/Advanced

```
tains.net          ; 学内 LAN に関する議論
tains.q-a          ; Q&A
tains.misc         ; tains に関係するが上記のどれにも当てはまらない話題
```

(4) fj (日本全国規模のニュースグループ; From Japan)

```
fj.announce          ; すべての人に対する一般的なアナウンス (Moderated)
fj.binaries.mac      ; テキスト化されたバイナリの Macintosh 用プログラム (Moderated)
fj.binaries.msdos    ; テキスト化されたバイナリの MS-DOS 用プログラム (Moderated)
fj.binaries.msdos.d; fj.binary.msdos に投稿された記事に関する話題
fj.books             ; 各種書籍についての話題.
fj.comp.image        ; CG,アート,アニメーション,画像処理
fj.comp.printers     ; プリンタ,ハード,ソフト
fj.editor.emacs      ; Unix のエディタとして有名な EMACS に関する話題
fj.followup          ; fj.general の記事に対するフォローアップ
fj.forsale           ; さっと読めておいしい「売ります買います」
fj.general           ; 重要かつタイムリーなすべての人に対するアナウンス
fj.guide.general     ; Junet の手引,規則,監修など (Moderated)
fj.guide.newusers    ; 新規ユーザに対する説明 (Moderated)
fj.lan               ; ローカルエリアネットワークのハードとソフトに関する話題
fj.lang.c            ; C言語に関する議論
fj.lectures          ; ニュース購読者が呼んでためになる話
fj.living            ; 日常生活に関する色々なことに関する話題
fj.meetings          ; 会議やワークショップ等のアナウンス
fj.misc              ; 日本国内の話題だが,他のどのグループにも当てはまらない話題
fj.news.group        ; ニュースグループのリストとニュースグループに関する議論
fj.news.lists        ; ニュースに関係した統計情報やリスト
fj.os.msdos          ; MSDOS に関する話題
fj.sci.bio           ; 生物学に関する話題
fj.soc.men-women     ; 男女平等,男女同権などに関する話題
fj.sources           ; ソースプログラム
fj.sys.sun           ; SUN ワークステーションに関する議論
fj.test              ; JUNET 全体でニュースの動作を確認するために使われる.
........
........
```

(注意) (2)kudpc (3)tains には投稿できない.

9.1 Rn

【概略】
Rn はコマンド rn でニュースを読み，Pnews で記事を投稿する．

【処理環境】

- UXP/M1800
  command : rn(/usr/local/bin/rn), Pnews(/usr/local/bin/Pnews)

【使用例】

- 大型計算機センターに関するニュースグループ kyushu-u.cc のニュースを読む．

  ```
  kyu-cc% rn kyushu-u.cc ↵
  ```

【注意事項】

- follow 記事は rn コマンドの記事選択でサブコマンド F または f を用いて投稿するほうが便利．

【提供】

- Larry Wall

【参考文献】

- 「UXP における電子メール入門（電子メール講習会資料）」九州大学大型計算機センター・ネットワーク室.
- 「UXP/M for the Working Scientists」佐藤 周行, 1993.

9.2 Nn

【概略】
Nn(No News is good news) は記事を選択的に読むことに適している．Nn を起動するとメニュー形式となり，必要な記事を選択する．nngrab という高速なキーワード検索プログラムが用意されている．

【処理環境】

- UXP/M1800
  command : /usr/local/bin 下のコマンド群

Nn のコマンドは次の通り．

nn	:	読む
nnpost	:	投稿する
nngrab	:	記事検索
nngrep	:	news group 検索
nngoback	:	再度未読にする
nnusage	:	ニュース状況検索
nntidy	:	.newsrc 管理
nncheck	:	未読記事，グループ検索

【使用例】

- 電子ニュースを読む

```
kyu-cc% nn ↵
```

【提供】

- Kim Fabricius Storm

【参考文献】

- 「UXP における電子メール入門（電子メール講習会資料）」九州大学大型計算機センター・ネットワーク室.
- 「UXP/M for the Working Scientists」佐藤 周行, 1993.

9.3 gnus

【概略】

Emacs 上でニュースを読み，投稿するツール．`emacs -f gnus` または Emacs の中で `ESC gnus` を実行すること．

【処理環境】

- UXP/M1800

【使用例】

- `gnus` を用いたニュースの読み書きを行なう．

```
kyu-cc% emacs -f gnus ↵
```

主なサブコマンドは次の通り．

空白	:	グループの選択
a	:	記事の投稿
n	:	次のグループへ
f	:	フォローアップ
p	:	前のグループへ
r	:	記事への返事（mail で）
j グループ名	:	グループの指定
q	:	gnus を抜ける

【提供】

- Masanobu Umeda

【参考文献】

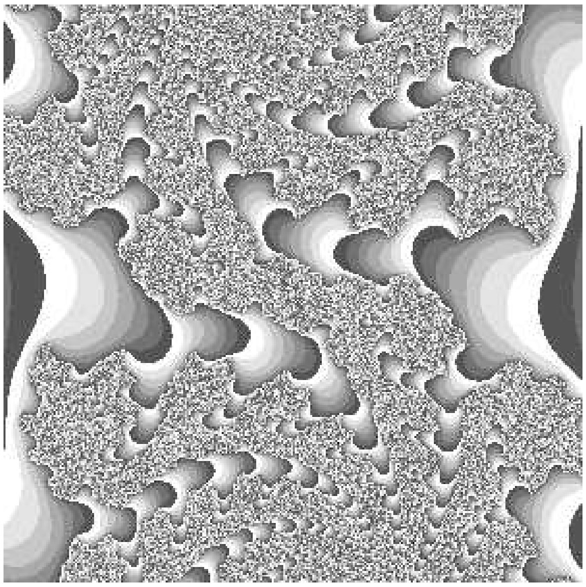
- 「UXP/M for the Working Scientists」佐藤 周行, 1993.
- 「UXP における電子メール入門（電子メール講習会資料）」九州大学大型計算機センター・ネットワーク室.

10

Editor

editor はコンピュータで作業をするために必要不可欠のライブラリである．どれか憶える必要がある．

editor 名	特徴	MSP	UXP
EDIT	`EDIT` コマンドで起動	○	
PFD/EDIT	プログラム開発支援システム	○	
ed, ex	line editor		○
emacs	vi と並び UNIX の標準 editor		○
ned	富士通提供 full screen editor		○
sed	stream editor		○
vi	emacs と並び UNIX の標準 editor		○



自己同型フラクタル 2

10.1 EDIT

【概略】
ソースプログラムやデータなどからなる順編成，または区分編成データセットを作成および編集する．EDIT コマンドは言語プロセッサの入力ソースプログラムやアプリケーションプログラムで使用するデータセットの作成，変更，保存，検索を行うための豊富なサブコマンドが用意されている．EDIT コマンドで起動される行エディタは，フルスクリーン機能を持たない端末でも利用可能である．フルスクリーン端末でデータセットの編集を行なう場合は，PFD の EDIT 機能を用いた方が便利である．

【処理環境】

- MSP/M1800
  command : EDIT

【使用例】

- Fortran ソースプログラム用データセットを作成し，編集する．

```
 READY
EDIT TEST.FORT NEW  ↵
```

【提供】

- 富士通株式会社

【参考文献】

- HELP EDIT
- 「利用の手引・ TSS 編」九州大学大型計算機センター・ネットワーク室, 1993.
- 「利用の手引・ MSP コマンド編」九州大学大型計算機センター・ライブラリ室, 1993.
- 「OS IV/MSP TSS/E コマンド文法書」(79SP-4091)，富士通株式会社．

10.2 PFD/EDIT

【概略】

PFD(Programming Facility for Display users) はフルスクリーン型端末を対象に設計されたプログラム支援ツール．PFD コマンドによって画面表示されるメニューから必要事項を選択する方式により，TSS の効率的な利用が可能となる．PFD はデータセットを指定して作業を行なう機能（PFD コマンドで起動）と，データセット一覧を表示した上でコマンドを指定して作業を行なう機能（PFD/E と呼ばれ PFDE コマンドで起動）がサポートされている．一般には PFD のオプションで起動される editor を用いて MSP 上の作業を行なう．関連コマンドとして PFDE，ED がある．

【処理環境】

- MSP/M1800
  command : PFD，PFDE，ED

【使用例】

- 日本語を含むデータセット TEST1.TEXT を編集する．

```
 READY
ED TEST1.TEXT NI  ⏎
```

【提供】

- 富士通株式会社

【参考文献】

- 「利用の手引・MSP コマンド編」九州大学大型計算機センター・ライブラリ室, 1993.
- 「PFD と PFDE の使用法」 平野 広幸，九州大学大型計算機センター広報, Vol.25, No.2, pp.119–165, 1992.

10.3 ed, ex

【概略】

ed は編集するファイルをバッファという一時ファイルに複写し，操作を実行する．一時ファイルに加えられた変更は書き込みコマンドを入力するまで，編集するファイルに対しては何の影響も及ぼさない．バッファは 1 つである．

ex は ed の機能をすべて含み，画面編集機能を持つ拡張版． ASCII 文字と同様に補助コード・セットの文字を扱うことができる．

【処理環境】

- UXP/M1800
  comamnd : `ed(/usr/bin/ed)`, `ex(/usr/bin/ex)`

【提供】

- 富士通株式会社

【参考文献】

- `man ed`
- `man ex`
- 「UXP/M for the Working Scientists」佐藤 周行, 1993.

10.4 emacs

【概略】

単なる editor ではなく，多くの拡張性をもつ機能が追加でき．総合的 user interface となっている．GNU Emacs に日本語機能を追加している．

【処理環境】

- UXP/M1800
  comamnd : `emacs(/usr/local/bin/emacs)`

emacs 画面

【提供】

- Richard Stallman

【参考文献】

- 「UXP/M for the Working Scientists」佐藤 周行, 1993.
- 「初めて使う GNU Emacs」矢吹 道郎，宮城 史朗，啓学出版, 1992.

10.5 ned

【概略】

F6262 型端末向けに設計された full screen text editor．ned の強力な機能の 1 つはその正規表現のマッチング処理にある．正規表現とはファイル中のデータ行の部分とマッチングをとるための文字列パターン．専用端末 (F6262 型端末) が必要．

ed や vi のような他の editor とは異なり，ned は標準入力を編集ファイルとして受け取ることができる．従って ned は pg の代わりに，パイプライン内での最終コマンドとして使用できる．

【処理環境】

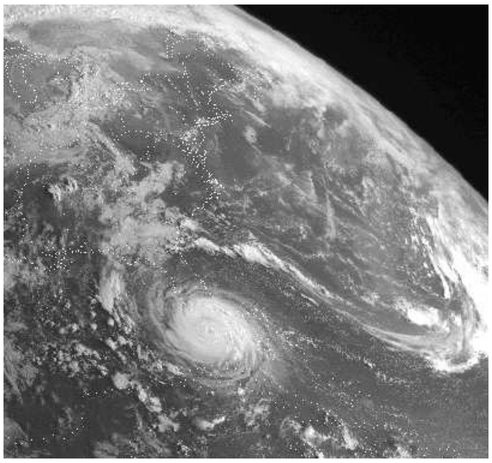
- UXP/M1800
  comamnd : `ned(/usr/uxp/ned)`

【提供】

- 富士通株式会社

【参考文献】

- `man ned`
- 「UXP/M for the Working Scientists」佐藤 周行, 1993.



10.6 sed

【概略】
sed は stream editor．指定した file (省略時は標準入力) をコマンド・スクリプトに従って編集し，標準出力に複写する．-f オプションを指定するとスクリプトはファイル sfile から取られる．このオプションを複数指定するとそれらがまとめられて 1 つのスクリプトとなる．-e オプションが 1 つだけで-f オプションを指定しない場合にはフラグ-e は省略しても構わない．-n オプションを指定すると省略時出力が抑制される．

【処理環境】

- UXP/M1800
  comamnd : `sed(/usr/bin/sed)`

【提供】

- AT&T

【参考文献】

- `man sed`
- 「UXP/M for the Working Scientists」佐藤 周行, 1993.

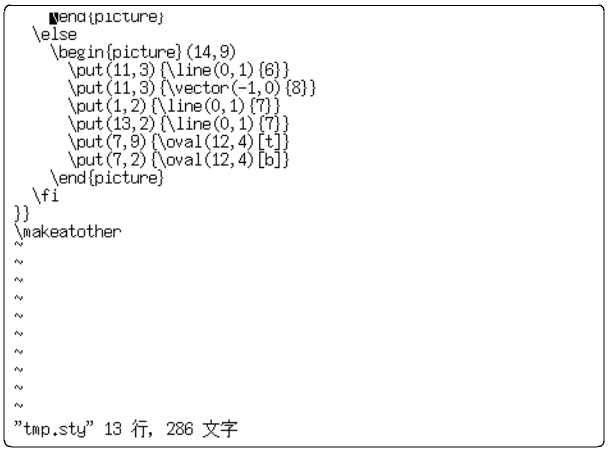
10.7 vi

【概略】

vi(visual) は基礎となる Line editor ex に基づくディスプレイ指向の text editor． vi 内から ex のコマンド・モードを使用することも，その逆も可能． vi 使用中ファイルに対する user による変更は端末画面上の表示に反映される．画面上のカーソルの位置はファイル内での位置を示す。 vi は一貫した user interface を使用して補助コード・セットの文字を処理し，表示することができる．

【処理環境】

- UXP/M1800
  comamnd : `vi(/usr/bin/vi)`



vi 画面

【提供】

- UCB

【参考文献】

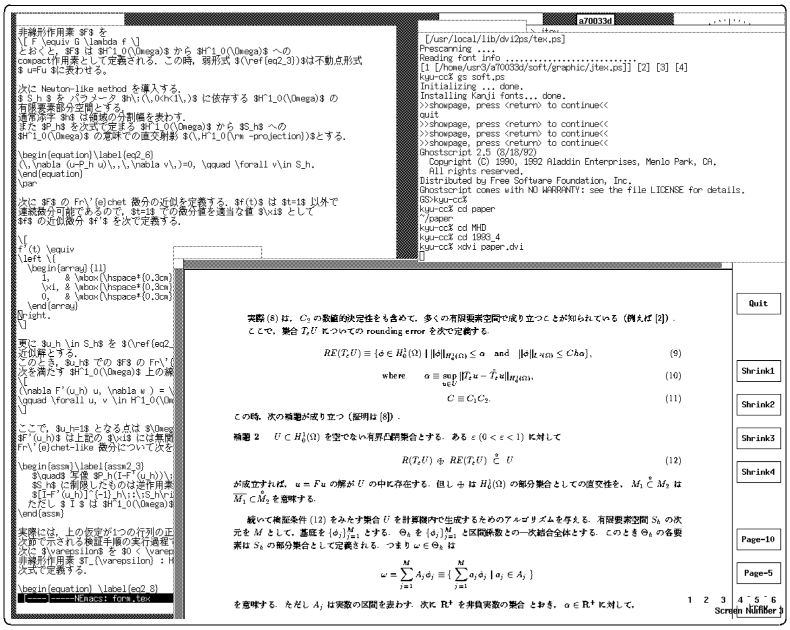
- `man vi`
- 「UXP/M for the Working Scientists」佐藤 周行, 1993.

11

文書処理

最近はノート型パーソナル・コンピュータでそこそこの memory, harddisk を搭載したものが出回り，文書処理環境が，電車の中や出張先に持ち込まれる時代である．しかし，UXP で行なう TeX のコンパイルの早さも捨て難い．

ライブラリ名	特徴	MSP	UXP
Roff	AT&T 開発の文書処理システム		○
TeX	NTT JTeX をサポート	○	○



JLaTeX を用いての論文作成風景

11.1 Roff

【概略】
Roff(Run off) は文章を指定した形式に整えて出力する文書作成システム．1960 年代に AT&T で開発された．Roff の入力ソースは自由形式の英文テキストと出力の書式を制御するコマンドが混在した形になっている．Unix のオンラインマニュアルのソースは Roff 形式で書かれている．なお，UXP の Roff は日本語の使用が出来ない．

【処理環境】

- UXP/M1800

作業内容	コマンド
Format	`troff(/usr/ucb/troff)`
Preview	`nroff(/usr/ucb/nroff)`
Print out	`lp(/usr/bin/lp)`, `psroff(/usr/local/bin/psroff)`
Utility	`eqn(/usr/ucb/eqn)` `tbl(/usr/ucb/tbl)` `refer(/usr/ucb/refer)`

Roff Family

詳しい使用方法は下記参考文献または `man` コマンドを参照．

【形式の変換】
Roff 形式への（からの）変換ツールとして，次のコマンドがある．

`plain2(/usr/local/bin/plain2)`
テキスト形式を Roff 形式に変換

`tr2tex(/usr/local/bin/tr2tex)`
Roff 形式を TEX 形式に変換

【提供】

- AT&T, 富士通株式会社

【参考文献】

- 「UXP/M for the Working Scientists」佐藤 周行, 1993.
- `man` コマンド

11.2 TEX

【概略】
TEX は，スタンフォード大学の Donald E. Knuth が開発した高品質組版プログラム．もともと Knuth 氏の自著 *The Art of Computer Programming* を，活版印刷によらずにコンピュータで組版する目的で開発されたため，数式の綺麗さは他の清書ソフトにないものである．

TEX は American Mathematical Society の商標であり，理科系の研究者では，実質論文原稿の標準となってしまっている．論文雑誌の掲載先から，「TEX のソースを送ってくれ」と言われたり，最近は電子メールで簡単に海外と情報が交換できるので，いきなり「これを読んで感想を聞かせろ」という文面と共に一面識もない研究者から TEX のソースを送りつけられて困った人もいるだろう．

LATEX は，DEC 社の Leslie Lamport が作成した TEX のマクロ（拡張命令）集．LATEX は文書整形のためのコマンドよりもテキスト構造に神経を集中させることで，user が組版にかける手間を省けるように追加されたコマンド群である．特に数式や図表を書かない場合でも，面倒なことは勝手に LATEX が清書処理をしてくれる簡便さと，文書の形式を規定する style file が，論文用だけでなく，手紙用，議事録用，テープラベル用，FAX 用と出回っていることから，数式を使用しない場合でも，ワープロソフトとして LATEX を利用する人も多い．センターの TEX は，NTT 版 jTEXをサポートしている．アスキー版の日本語 TEX とは非互換があるので注意．

【処理環境】

- MSP/M1800
  command : `TEX`, `LATEX`, `RTEX`, `RLATEX`
- UXP/M1800
  command : `jlatex`, `jtex`, etc

MSP の TEX 環境は未整備である．ここでは UXP を中心に解説を行なう．

作業内容	コマンド	機能
Format	`jlatex`	jLATEX の起動
	`jtex`	jTEX の起動
	`amstex`	数学論文用 TEX マクロ
Preview	`dvi2tty`	tty 端末
	`dvi2vdu`	Tektronix 端末
	`xdvi`	X Window
Print out	`dvi2ps`	PostScript 出力
Utility	`makeindex`	索引作成
	`texdraw`	簡単な図作成
	`jbibtex`	文献データベース

TEX family

上記 TEX ライブラリは，全て `/usr/local/bin` 配下のファイル．

【出力例】
本書をパラパラめくる．表紙，裏表紙を除いて全て UXP 上の JLATEX で作成している．挿入されている図は PostScript 形式でファイルに保存したものをJLATEX に組み込んだものである．紙の切り張りは一切ない．

【形式の変換】
TEX 形式への（からの）変換ツールとして，次のコマンドがある．

`plain2 -tex(/usr/local/bin/plain2)`
: テキスト形式を TEX 形式に変換

`tr2tex(/usr/local/bin/tr2tex)`
: Roff 形式を TEX 形式に変換

`dvi2tty(/usr/local/bin/dvi2tty)`
: TEX の dvi ファイルをテキスト形式に変換．ソースファイルが何らかの原因で消えてしまった場合の復活法として利用できる．

【備考】

- TEX の `Style file` は，`/usr/local/lib/tex/inputs` 下にある．
- JLATEX の入門書は市販に大変優れたものがある．参考文献の最後の 3 冊を全て揃えれば，ソフトの入手，パソコンへのインストール方法から，自分の好みにあった形式に `Style file` を修正する方法やフォントの作成方法まで全て分かる．但し，お金が 8,000 円程かかる．

【参考文献】

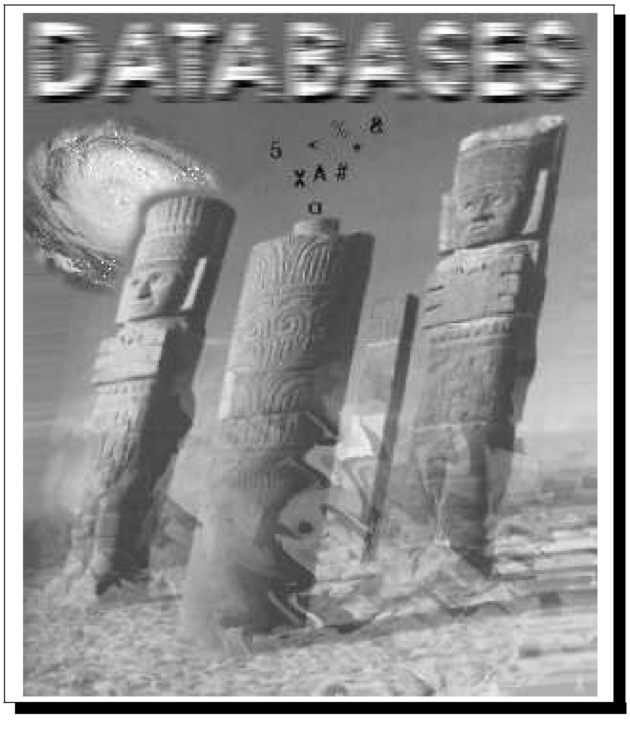
- 「UXP/M for the Working Scientists」佐藤 周行, 1993.
- 「利用の手引・ MSP コマンド編」九州大学大型計算機センター・ライブラリ室, 1993.
- 「The TEXbook」 Donald E. Knuth, Addison Wesley Publishing Company, 1984.
- 「文書処理システム LATEX」 Leslie Lamport 著, Edgar Cooke, 倉沢 良一 監訳，大野 俊治，小暮 博道，藤浦 はる美 訳，アスキー出版局, 1990.
- 「LATEX 美文書作成入門 – パソコンによる文書レイアウト」 奥村 晴彦，技術評論社，1991.
- 「LATEX 自由自在」 磯崎 秀樹，サイエンス社, 1992.

12

データベース

データベース名	特徴	MSP	UXP
CXDB	有機及び有機金属化合物の結晶構造データ		○
GENEDB	ヌレオクチド，蛋白質配列データ	○	○
ゲーテ・ファイル	ゲーテ全集の機械可読化	○	
トーマス・マン・ファイル	トーマス・マン全集の機械可読化	○	
INSPEC	IEE 提供の情報サービス	○	
JICST-J	日本科学情報センター提供の科学技術文献ファイル	○	
KONTHU	昆虫学雑誌からの書誌情報	○	
RAMBIOS	分子生物科学のレビュー文献	○	
日本語単語辞書	拡張 B 木による検索	○	

計算機の利用は，数値計算的な側面と，記憶された情報を整理し，必要なものを検索する側面とがある．有機的に組織された大規模なデータの集合（データベース）の中から，指定された条件を満たす情報を的確かつ迅速に選び出す検索方法や収納方式を考える分野を“データ処理 (data processing)”と呼ぶ．所詮有限個のデータだと侮ってはいけない．分類を行なう操作の手間は，要素の個数に単純に比例するのではなく，下手をすれば要素の個数に指数的に比例して増大するのである．出した質問が“たらい回し”にされて，無限ループに陥る危険もある．データ処理の研究者は，計算機ハードウェアに関する知識はもちろん，グラフ理論，組み合わせ理論，更には代数学全般に深い造詣のある人たちである．尊敬して下さい．



12.1 CXDB

【概略】
CXDB は，英国の Cambridge Crystallographic Data Centre, University Chemical Laboratory(CCSC) が全世界的規模で集積・配布を行なっている結晶構造データ．X 線又は中性子の回折強度測定による結晶構造解析を報告した論文からとられており，対象となる化合物は蛋白質と高分子化合物を除く有機化合物，有機金属化合物及び金属間化合物である．

【処理環境】

- UXP/M1800
  command : `xdt(/usr/local/bin/xdt)`

【データベース作成】

- ケンブリッジ結晶データセンター

【データベース提供】

- 河野 重昭（九州女子学院短期大学）

【参考文献】

- 「結晶構造データベースシステム XDT の使用法（暫定版第一報）」 河野 重昭，高木 利久，二村 祥一，松尾 文碩，鬼塚 千代子, 九州大学大型計算機センター広報, Vol.16, No.1, pp.32–53, 1983.
- 「結晶構造データベースシステム XDT について（綜合版）」河野 重昭，高木 利久，二村 祥一，松尾 文碩，鬼塚 千代子, 九州大学大型計算機センター広報, Vol.16, No.6, pp.556–604, 1983.
- 「XDT 用統計プログラムについて」河野 重昭, 九州大学大型計算機センター広報, Vol.17, No.2, pp.67–74, 1984.
- 「結晶構造データベースシステム XDT のデータ追加について」九州大学大型計算機センター広報, Vol.20, No.5, pp.482, 1987.
- 「結晶構造データベースシステム XDT のデータ追加について」九州大学大型計算機センター広報, Vol.21, No.6, pp.584, 1988.
- 「ケンブリッジ結晶構造データベース用プログラム QUEST88 の使用法」河野 重昭, 九州大学大型計算機センター広報, Vol.23, No.3, pp.186–202, 1990.
- 「ケンブリッジ結晶構造データベースの更新と機能追加について」九州大学大型計算機センター広報, Vol.24, No.6, pp.634–636, 1991.

12.2 GENEDB

【概略】

データベース GENEDB は次のデータから構成されている．

- ヨーロッパ分子生物学研究所 EMBL(European Molecular Biology Laboratory) の Nucleotide Sequence Data Library の塩基配列データ．
- Georgetown 大学の NRBF(National Biomedical Research Foundation) の Protein Sequence Database のデータ
- アメリカ Los Alamos 国立研究所が監修して Bolt Beranek and Newman(BBN) 社が作製，配布している遺伝子データーベース GenBank(R) の塩基データ

データベースに対する検索及び，応用プログラムによる処理は，上記データベースと配列データ解析プログラムとをデータベース管理システム Adbisによって統合したシステム GENAS によって行なう．

【処理環境】

- MSP/M1800
command : `GENAS, TSUNO`
- UXP/M1800
command : `genas(/usr/local/bin/genas)`

【データベース作成】

- EMBL, NBRF, Los Alamos 研究所

【データベース提供】

- 榊 佳之（九州大学遺伝情報実験施設）

【参考文献】

- 「核酸塩基配列データベース GENAS の使用法 (1)」 久原 哲，榊 住之，高木 利久，松尾 文碩，二村 祥一，鬼塚 千代子，九州大学大型計算機センター広報, Vol.16, No.5, pp.497–521, 1983.
- 「核酸塩基配列データベース GENAS の使用法 (2)」 久原 哲，榊 住之，高木 利久，松尾 文碩，二村 祥一, 九州大学大型計算機センター広報, Vol.18, No.6, pp.533–551, 1985.
- 「核酸塩基配列データベースシステム GENAS の使用法 (3)」古市 恵美子，松尾 文碩，高木 利久，二村 祥一，榊 住之，久原 哲, 九州大学大型計算機センター広報, Vol.24, No.1, pp.1–18, 1991.
- 「新 GENAS の公開」九州大学大型計算機センター広報, Vol.26, No.3, pp.321–326, 1993.

12.3 ゲーテ・ファイル, トーマス・マン・ファイル

【概略】

テキスト・データベース「ゲーテ・ファイル」はJ.W.v.Goethe(1749-1832) のハンブルク版ゲーテ全集（全 14 巻）のテキスト（約 6,800 頁）及び，ワイマル版ゲーテ全集の第４部書簡集１～４０巻（約 12,000 頁）より構成されている．「トーマス・マン・ファイル」はS. Fischer 書店より出版された Thomas Mann 全集の全 13 巻を機械可読化したものであり， 11,410 頁，約 40 万行のデーター量である．

利用はテキスト・データベース管理システム SIGMA で行なう．SIGMA は多数のテキストファイルを管理・検索・編集する機能を持っており，文献データベースの作成や検索，自然言語の解析などに活用されている．

【処理環境】

- MSP/M1800
  command : `SIGMA`

【データベース作成，提供】

- 樋口 忠治（九州大学言語文化部）

【注意事項】

- 「ゲーテ・ファイル」「トーマス・マン・ファイル」のソースデータ自体のコピーを作ることは如何なる目的のためであっても，これを禁止する．

【参考文献】

- 「テキストデータベース管理システム SIGMA 第 2 版について」 有川 節夫，篠原 武，武谷 峻一，大島 一彦，原口 誠，白石 修二，酒井 浩，宮原 哲浩，山本 章博，井上 仁，川崎 洋浩，湯浅 寛子, 九州大学大型計算機センター広報, Vol.20, No.6, pp.517–581, 1987.
- 「テキスト・データベース『ゲーテ・ファイル』の公開」 樋口 忠治, 九州大学大型計算機センター広報, Vol.21, No.3, pp.167–176, 1988.
- 「ゲーテ・ファイルの追加について」 樋口 忠治, 九州大学大型計算機センター広報, Vol.24, No.6, pp.637–639, 1991.
- 「公用データベース トーマス・マン・ファイル／ SIGMA の公開」樋口 忠治，篠原 武, 九州大学大型計算機センター広報, Vol.16, No.4, pp.379–393, 1983.
- 「公用データベース 『トーマス・マン・ファイル』のファイル追加について」樋口 忠治，篠原 武, 九州大学大型計算機センター広報, Vol.18, No.2, pp.73–79, 1985.
- 「テキストデータベース『トーマス・マン・ファイル』の完成と再編成について」 樋口 忠治，篠原 武, 九州大学大型計算機センター広報, Vol.20, No.6, pp.582–596, 1987.

12.4 INSPEC

【概略】
INSPEC(Information Service in Physics, Electrotechnology and Control) は，イギリス IEE (Institution of Electrical Engineers) の提供する情報サービス．二次文献（文献内容の要約）情報は INSPEC テープと呼ばれる磁気テープの形で送られる．内容は抄録誌に対応して，分類コード A, B, C, D を持つ．センターでのサポートは次の通り．

データベース名	データベースの内容	
INSPECA	物理学の文献情報	1969 年以降
INSPECB	電気・電子工学の文献情報	1969 年以降
INSPECC	制御工学，計算機科学，情報工学の文献情報	1969 年以降
INSPECJ	INSPEC 雑誌データの詳細	1992 年版

【処理環境】

- MSP/M1800
  command : `AIR`

【データベース作成】

- イギリス電気学会 (IEE)

【データベース提供】

- 九州大学大型計算機センター

【参考文献】

- 「情報検索システム AIR の改訂について」篠原 武，二村 祥一，松尾 文碩, 九州大学大型計算機センター広報, Vol.20, No.3, pp.221–238, 1987.
- 「利用の手引き・情報検索システム AIR —AIR を用いた文献情報の検索—」九州大学大型計算機センター・データベース室, 1993.

12.5 JICST-J

【概略】
JICST-J は日本科学技術情報センター (JICST) 提供の科学技術文献ファイル．情報工学関係の和文二次文献情報を含むデータベース．

JICST-J を検索できるのは JICST との契約により，九州大学の利用者に限られている．ただし，九州大学以外の利用者でも九州大学大型計算機センターの端末から利用するのは差し支えない．

【処理環境】

- MSP/M1800
  command : `AIR`

【データベース作成】

- 日本科学技術情報センター (JICST)

【データベース提供】

- 九州大学大型計算機センター

【参考文献】

- 「情報検索システム AIR の改訂について」篠原 武，二村 祥一，松尾 文碩, 九州大学大型計算機センター広報, Vol.20, No.3, pp.221–238, 1987.
- 「利用の手引き・情報検索システム AIR —AIR を用いた文献情報の検索 —」九州大学大型計算機センター・データベース室, 1993.

12.6 昆虫学データベース

【概略】

「昆虫学データベース (KONCHU)」は昆虫学に関する 1900 年以後の国内主要昆虫学雑誌をもとにしたデータベース．単なる文献データベースではなく，昆虫の様々な属性をデータとして含む新しいファクト・データベースとしての性格をもつ．

利用はテキスト・データベース管理システム SIGMA で行なう．SIGMA は多数のテキストファイルを管理・検索・編集する機能を持っており，文献データベースの作成や検索，自然言語の解析などに活用されている．

【処理環境】

- MSP/M1800
  command : `SIGMA`

【データベース作成，提供】

- 多田内 修（九州大学農学部）

【参考文献】

- 「テキストデータベース管理システム SIGMA 第 2 版について」 有川 節夫，篠原 武，武谷 峻一，大島 一彦，原口 誠，白石 修二，酒井 浩，宮原 哲浩，山本 章博，井上 仁，川崎 洋浩，湯浅 寛子, 九州大学大型計算機センター広報, Vol.20, No.6, pp.517–581, 1987.
- 「SIGMA による公用データベース昆虫学データベース (KONCHU) の公開とその利用法」多田内 修, 九州大学大型計算機センター広報, Vol.20, No.6, pp.597–614, 1987.

12.7 RAMBIOS

【概略】
RAMBIOS(Review Articles in Molecular Biosciences) は，生化学，生物物理学，分子生物学，分子生理学を中心とした分子生物科学のレビュー文献を収録したデータベース．キーワード，キーフレイズに配慮を加え，性格語（文献の性格を表すための語）を付与することにより，レビュー文献の検索が容易である．検索は情報検索システム AIR により行なう．

【処理環境】

- MSP/M1800
  command : `AIR`

【データベース作成】

- RAMBIOS 刊行会

【データベース提供】

- 九州大学大型計算機センター

【参考文献】

- 「情報検索システム AIR による RAMBIOS の検索」木原 裕，二村 祥一，河津 秀利，林 健児，和田 正美，郷 信宏, 九州大学大型計算機センター広報, Vol.18, No.6, pp.497–506, 1985.

12.8 日本語単語辞書

【概略】
日本語機械辞書の基本語部分（自立語 約 8 万 3 千語）に付属語辞書（形式名詞，補助用言，自立語の活用語尾を含む）と文節内における二単語間の接続検定用テーブルより構成される公用データベース．サブルーチンとして応用プログラム内から呼び出すことで起動する．

【処理環境】

- MSP/M1800

【データベース作成】

- 稲永 紘之（九州芸術工科大学）

【データベース提供】

- 日高 達（九州大学工学部）

【参考文献】

- 「公用データベース日本語単語辞書の使用について」吉田 将，日高 達，稲永 紘之，田中 武美，吉村 賢治，九州大学大型計算機センター広報, Vol.16, No.4, pp.335–361, 1983.

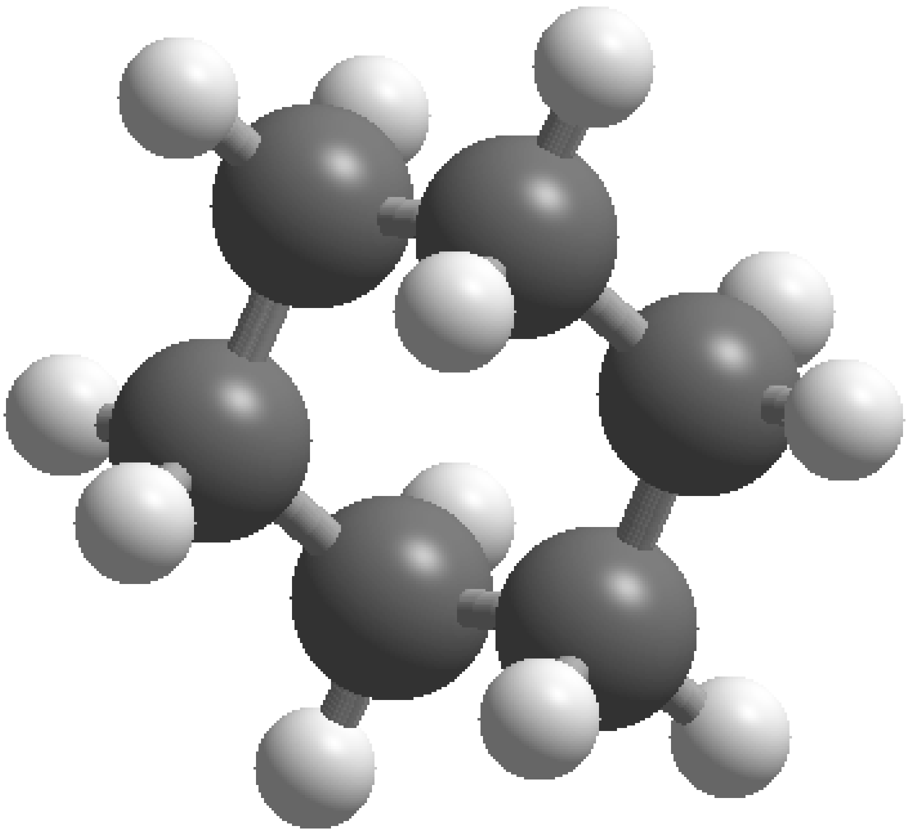
Graphics in this section: *Konomi, Shin'ichi*

13

分子軌道計算

以下のライブラリが利用可能．

ライブラリ名	特徴	MSP	UXP
Gaussian 88	Gaussian Inc. 提供	○	
GSCF3	Gauss 型基底 ab initio 分子軌道法	○	
JAMOL3	ab initio LCAO MO SCF Calculation	○	



cyclohexane

13.1 Gaussian 88

【概略】
専門用語が乱発して誤訳しそうなので，マニュアルから機能を英文で抜粋する．

Gaussian 88 is a connected system of programs for performing semiempirical and *ab initio* molecular orbital (MO) calculations.

The capabilities od *the Gaussian 88* system include:

- Calculation of one- and two-electron integrals over s, p, d, and f contracted gaussian functions.
- Self-consistent field calculations for restricted closed-shell (RHF), unrestricted open-shell (UHF), and open-shell restricted (ROHF) Hartree-Fock wavefunctions as well as those types of multiconfigurational wavefunctions that fall within the Generalized Valence Bond-Perfect Pairing (GVB-PP) formalism.
- Evaluation of various one-electron properties of the Hartree-Fock, MP2, and CI energies and wavefunctions, including Mulliken population analysis, multipole moments, and electrostatic fields.
- Semi-empirical calculations using the CNDO, INDO, MINDO/3, MNDO, and AM1 model Hamiltonians.
- Automated geometry optimization to either minima or saddle points, numerical differentiation to produce force constants, polarizabilities, and dipole derivatives, and reaction path following.
- Correlation energy calculations using Moller-Plesset perturbation theory carried to second, third, or fourth order.
- Correlation energy calculations using configulation interaction (CI) using either all double excitations (CID) or all single and double excitations (CISD).
- Correlation energy calculations using coupled cluster theory with double substitutions (CCD), and Quadratic Configuration using single and double substitutions (QCISD). Cite (QCISD)
- Analytic computation of the nuclear coordinate gradient of the RHF, UHF, ROHF, GVB-PP, MP2, RCID and RCISD energies.
- Computation of force constants(nuclear coordinate second derivatives), polarizabilities hyperpolarizabilities, dipole derivatives, and polarizability derivatives analytically for RHF and UHF energies, and numerically for RHF, UHF, MP2, CID and CISD energies.
- Harmonic vibrational analysis.
- Determination of intensities for vibrational transitions at the HF, MP2, and CI levels.

- Testing the SCF wave functions for stability under release of constrains.

【処理環境】

- MSP/M1800
  cataloged procedure : `GAUSSIAN`

【使用例】

- ジョブ制御文は次のように記述する．

```
//A79999AA JOB CLASS=F
//  EXEC GAUSSIAN
//SYSIN DD *
   # HE/STO-3G
   (-)-EPICATECHIN   DXCX1=180

    0  1
    X
    C  1  1.387
    C  1  1.388  2   60.0
    C  1  1.386  3   60.0  2  180.0
    C  1  1.386  4   60.0  3  180.0
    C  1  1.406  5   60.0  4  180.0
    C  1  1.397  6   60.0  3  180.0
    X  2  1.0    1   90.0  7   90.0
    X  3  1.0    1   90.0  2   90.0
    X  4  1.0    1   90.0  3   90.0
    X  5  1.0    1   90.0  4   90.0
           :

/*
//
```

【提供】

- Gaussian, Inc.

【参考文献】

- User's Guide and Programmer's Reference, Australian National University Facility.
- Gaussian88 User's Guide and Programmer's Reference (Revision C Version), Multiflow Computer Inc.
- 「利用の手引・バッチジョブ編」九州大学大型計算機センター・ライブラリ室, 1992.

13.2 GSCF3

【概略】
GSCF3 は各種分子の電子状態をガウス型基底 ab initio 分子軌道法により計算するシステム．d 軌道まで (Cartesian タイプで 6 関数) 扱え，制限型 SCF(RHF) 法と非制限型 SCF(UHF) 法の両方が実行可能．早い収束性を実現し，励起状態に対する SCF 計算も可能．

現在東京大学大型計算機センターで公開中の GSCF3 の SCF 部分を九州大学大型計算機センターに移植し, 電子間反発積分計算部分についてはベクトル計算機用に最適化されたプログラムを追加している．

プログラム形式	コンプリートプログラム
作成者 作成年月日	小杉 信博 (東京大学工学部) 1985 年
移植者 移植年月日	八尋 秀一，権藤 恭彦（九州大学理学部） 1990 年 11 月
使用言語	Fortran
使用 OS	MSP(M1800, VP2600)

【処理環境】

- MSP/M1800, MSP/VP2600
  command : `GSCF3`
  catalogued procedure : `GSCF3`

【備考】

- 使用に際しては東京大学大型計算機センター発行の利用の手引と，九州大学版を利用するための解説記事が必要（下記参考文献）．

【注意事項】

- GSCF3 を用いて得られた結果を公表する場合は，開発者名およびプログラム名を明記すること．

【参考文献】

- 「GSCF3 利用の手引き（SCF 部分）」 東京大学大型計算機センター, 1986.
- 「分子軌道計算システム GSCF3 （SCF 部分）の利用法とベクトルマシンアルゴリズムの導入について」 八尋 秀一，権藤 恭彦, 九州大学大型計算機センター広報, Vol.23, No.6, pp.597–610, 1990.
- 「利用の手引・バッチジョブ編」九州大学大型計算機センター・ライブラリ室, 1992.
- 「利用の手引・ MSP コマンド編」九州大学大型計算機センター・ライブラリ室, 1993.

13.3 JAMOL3

【概略】
JAMOL3 は ab initio LCAO MO SCF Calculation のための分子軌道計算プログラムシステム．多原分子の波動関数を Roothaan の方法，即ち LCAO SCF MO (Linear Combination of Atomic Orbitals, Self Consitent Field, Molecular Orbital) 法により求める．閉殻および開殻電子状態の分子軌道計算は非経験的に行ない，population analysis や 1 電子演算子の期待値の計算も可能．n 電子系の波動関数を n 個のスピン軌道の積の形をした関数の線形結合であらわす．九州大学版として，価電子のみを取り扱い，全電子を考慮した計算を十分な精度で再現するモデルポテンシャル法におけるポテンシャル・パラメータ及び basis sets が JAMOL プログラムにデータベースとして組み込まれている．

プログラム形式	コンプリートプログラム
作成者	柏木 浩，高田 俊和 三好 永作，小原 繁 (北海道大学理学部)
作成年月日	1977 年 8 月
改訂者	酒井 嘉子（九州大学教養部） 三好 永作（福岡歯科大学）
改訂年月日	1986 年 7 月
使用言語	Fortran
使用 OS	MSP(M1800)

【処理環境】

- MSP/M1800, batch 処理

【備考】

- 使用に際しては MSP の MANUAL コマンドでマニュアル（英文）を NLP に出力して参照されたい．

```
 READY
MANUAL JAMOL3 ↵
```

- 出力されるマニュアルにオリジナル版と異なる九州大学版の note，テスト用ジョブ制御文が含まれている．

【注意事項】

- JAMOL3 を用いて得られた結果を公表する場合は，開発者名およびプログラム名を明記すること．

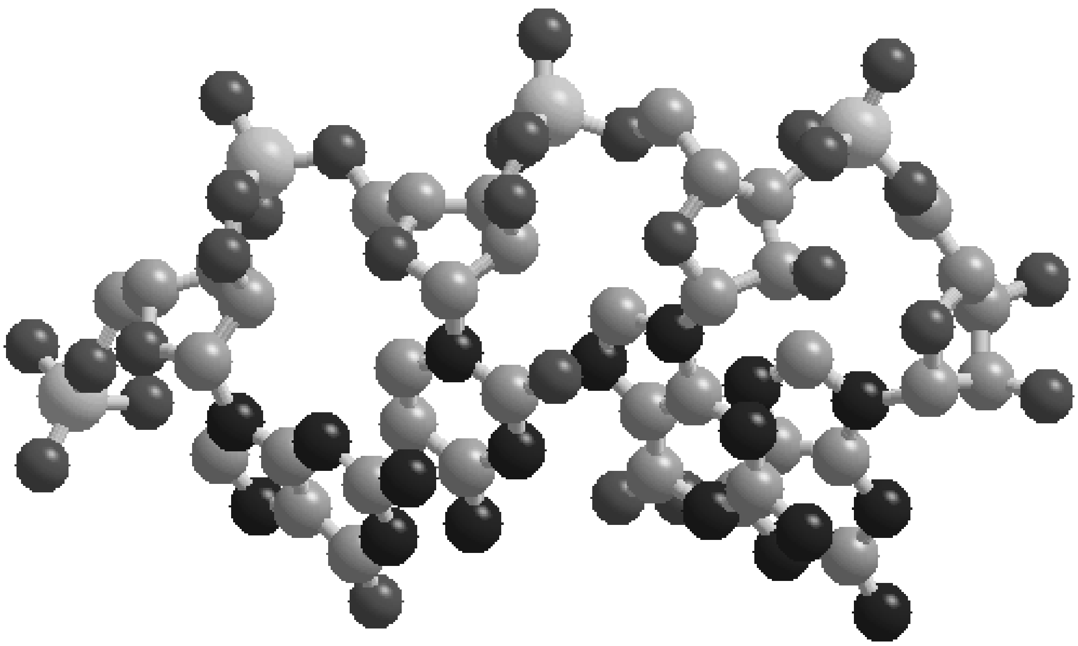
【参考文献】

- 「JAMOL3 MANUAL」(MANUAL コマンドで NLP に出力可).

14

原子核理論

ライブラリ名	特徴	MSP	UXP
CA01	計量修正による関数の極小点発見	○	
DA02	BCS 方程式	○	
DB01	Clebsch-Gordan 係数	○	
DB02	Racah 係数	○	
DB03	9-J 係数	○	
KNL1	直交条件模型の積分核	○	
VAR1	微積分方程式の散乱境界条件解	○	



GCGG

14.1 CA01

計算修正による関数の極小点発見 (Variable Metric Method for Minimization)

プログラム形式	サブルーチンプログラム
作成者 作成年月日	岡井 末二 (東京大学原子核研究所) 1969 年 6 月
改訂者 改訂年月日	上村 正康 (九州大学理学部) 1971 年 9 月
使用言語	Fortran
ソースの公表 使用 OS	'QS.LIBSC(Y3CA01)' MSP(M1800, VP2600)

【概要】

非線形多変数関数の極小点を Variable metric method を用いて求めるプログラムである．このプログラムでは変数の変域を指定することができる．また簡単な方法により，ある若干個の変数を固定して関数の極小点を求めることが出来る．

このプログラムは phase shift analysis や optical model analysis 等における best-fit parameters の search program として使用できることを念頭において作られた．既にそれぞれの分析の完成コードがある場合，これ等のコードをあまり変更しないで，サブルーチン形式で与えられているこのプログラムを，たとえばメインプログラムに書き直して使用する方が良いかもしれない．

【使用法】

1. 呼出方法

```
CALL HUNTER(N,X,FX,IPRINT,NCUT,DELTA)
```

- 引数による入力

N	: 整数，パラメータ (独立変数) の数
X	: 一次元配列実数，各パラメータの初期値 (推定値)
IPRINT	: 整数， iteration 毎のパラメータと関数値の出力の有無を指定 1 – 出力する 2 – 出力しない
NCUT	整数， iteration の上限
DELTA	: 実数，極小点から幾らの半径内に入れば良いかを指定

- 引数による出力

X	: 一次元配列実数，極小点におけるパラメータの値
FX	: 実数，極小点に於ける関数の値

- COMMON による入力

```
COMMON/XDATA/XMAX(20),XMIN(20),JOKEN
```

XMAX　: 一次元配列実数，各パラメータについての変域の上限
XMIN　: 一次元配列実数，各パラメータについての変域の下限
JOKEN　: 整数，正の整数でないと，変域の上下限指定は無効になる

- COMMON による出力
`COMMON/HDATA/H(20,20),KOUNT,IFN,INDEX,ISPILL`

H　: 二次元配列実数，極小点における error matrix
KOUNT　: 整数，iteration の current number
IFN　: subroutine FUNDER への入り口が 3 カ所ある．
各入口は INF の値 $(-1,-2,-3)$ で区別される
INDEX　: 整数，KOUNT が NCUT を越えると負の整数になる
ISPILL　: 整数，OVERFL や DVCHK が on のとき正の整数．

2. 出力データの形式

(a) IPRINT=1 のとき，FX と (X(I), $\mathrm{I}=1,\cdots,N$) の current values が次の形式で印刷される．

`CYCLE`	`F`	$(X(I),\quad \mathrm{I}=1,\cdots,N)\ \cdots$
`iteration KOUNT`	関数の値	パラメータの値
`I4`	`E16.5`	`(5E16.5)`

(b) 各パラメータについて変域の上下限の指定をしたとき，極小点が指定の領域外にあることが明らかな時，次の形式で指示される．

```
MINIMUM OUTSIDE OF X(I)=E12.5
```

このことから，どの変数の上限域は下限が問題なのかが判る．

3. エラー・メッセージ

(a) iteration KOUNT がその上限 NCUT を越えたとき

`OVER NCUT=` 入力で指定された数値

を出力して calling program に返る．この原因として NCUT が小さすぎることが考えられるが，大抵 FX, GX の計算かリンクの間違いによる．

(b) OVERRFL あるいは DVCHK が起こったとき，次のメッセージを出力し，calling program に返る．このとき subroutine FUNDER の内容と，そのリンクの仕方について再検討が期待される．

```
OVERFL TROUGH FNDR
DVCHK TROUGH FNDR
```

4. 所要時間
極小点を求むべき関数の複雑さの度合いに応じて決まる．大体の目安としてサブルーチン FUNDER の計算に要する時間を t_0 (sec) とすると，所要時間 t は $t \approx 4t_0 \times \mathrm{NCUT}$ (sec) 程度である．但し NCUT は大きな N に対しては 2N 程度で充分である．

5. 精度
   精度は，用いられる computer の有効桁数と，error matrix から決定される標準偏差の両方から決まる．一般にパラメータの数値は，次元 (ある成分は長さ，ある成分はエネルギーの次元等) が異なるために，order が異なっている．各パラメータの数値を，すべて order 1 に規格化しておくことはきわめて望ましい．通常 DELTA=0.01 あるいは 0.001 程度，つまり 1% あるいは 0.1% の精度に指定しておく方が安全である．

6. 制限事項
   パラメータ N の上限は，このプログラムでは 20 と指定してあるが，20 以上に変更しても差し支えない．ただし，関数が簡単な場合を除き，余り感心出来ない．関数の簡単さの規準は t_0 と t の値で判断する．この 20 という制限は，主として computer の計算速度と有効桁数によって決まる．

7. 使用ルーチン

   (a) 極小点を求めようとする関数は，user が準備しなくてはならない．この HUNTER ルーチンは，パラメータの値が与えられたとき，関数の値と，一次偏微分係数の値を計算するサブルーチン・サブプログラム

   ```
   SUBROUTINE FUNDER(N,X,FX,GX)
   ```

   を要求する．引数はすべて単精度．

   - 引数による変数の入力
     N　: HUNTER の N と同じ
     X　: HUNTER の X と同じであるが，ここでは関数の current values
   - 引数による変数の出力
     FX : HUNTER の FX と同じであるが，ここでは関数の current values
     GX : 一次元配列実数，一次元偏微分係数の current values
   - `COMMON/HDATA/` の中の IFN の利用方法．
     FUNDER ルーチンの中で，さらに細かい中間結果を出力したいとき，IFN は出力された一組の中間結果の identification として役に立つ．

   (b) この HUNTER ルーチンにはさらに 2 つのサブルーチンが組み込まれている．

      i. `SUBROUTIN PRINT(LINE)`
         - 引数による入力 (整数)
           LINE : WRITE 命令の直前にあって，その命令で印字される行数．
         - `COMMON/DATE/NUMRUN(5),NPGS,KLINE` による入力
           メイン・プログラムで準備されるべきものである．
           NUMRUN(1) : 整数で，年を示す．例えば 1994 とする
           NUMRUN(2) : 整数で，月を示す．例えば，8 とする
           NUMRUN(3) : 整数で，日を示す．例えば，25 とする
           NUMRUN(4) : 整数 (<1000) で各種計算の番号付けに用いる．
           NUMRUN(5) : 整数 (<1000) で各種計算の番号付けに用いる．
           NPGS　　 : 整数で，頁数を示す．
           　　　　　 メイン・プログラムに NPGS=0 があれば充分

KLINE : 整数で，一頁に印刷される全桁数 (ここでは 60 と指定した)
メイン・プログラムに KLINE=60 があれば充分

ii. `SUBROUTINE SPILL`
OVERFL と DVCHK の check

8. 備考

(a) 極小点を求めようとする関数は至る所 non-negative でなければならい．

(b) FUNDER ルーチンで一次偏微分 GX を求めるが，これ等の値は極小点近傍では非常に小さくなるので，プログラムを作るとき特に極小点での精度に留意して欲しい．

(c) パラメータのある特定の成分，例えば I 番目の成分を固定したいときは，その成分に対する偏微分係数 GX(I) を zero に reset するとよい．reset はもちろん FUNDER ルーチンで行なう．既に計算された GX(I) を御破算にする形を取れば良い．

(d) FUNDER ルーチンのテスト方法について
一般に，取り扱う関数 $f(x)$ のプログラムを作るとき，不注意に入る間違いは極めて稀であるが，$f(x)$ の gradient $g(x)$ の間違いは意外に多い，従って FUDER ルーチンのチェックの主な目的は $g(x)$ に関係する命令が正しく作動しているかどうかという点にかかっている．今，x_0 を任意点とし，σ をある方向を与える任意ベクトルとする．$f(x_0+\lambda\sigma)$ は，λ を変えてゆけば，どこか (λ_{min}) で極小になる．即ち $f(x_0+\lambda_{min}\sigma)$ では λ について微係数は零になる．この事実を用いるならば $f(x)$ の gradient $g(x)$ のプログラムが正しく作動している事を確認出来る．
具体的には

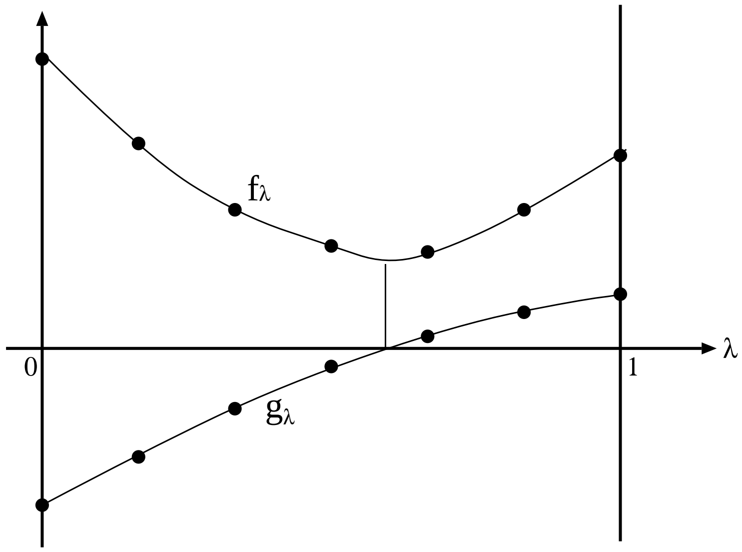
- 任意の点 x_0 で関数 $f(x_0)$ とその gradient $g(x_0+\lambda\sigma)$ を求める．
- $\sigma = -g(x_0)$ とおき，$f(x_0+\lambda\sigma)$ ($\equiv f_\lambda$) 及び $g(x_0+\lambda\sigma)$ を求める．但し $0<\lambda\leq 1.$
- 任意の値 λ に対する $f(x_0+\lambda\sigma)$ の微係数はスカラー積 $g_\lambda = \sigma g(x_0+\lambda\sigma)$ で与えられる．
- いくつかのの λ に対する f_λ , g_λ を求め，グラフとして plot する．
- グラフから，f_λ の極小点と $g_\lambda = 0$ なる点が，目で見て充分一致しているときは $g(x)$ は正しく作動していると考えてよい．

【注意事項】

- 本ライブラリを利用して得られた結果を公表する場合，プログラム名と開発者名を明記すること．

- 原子核ライブラリで以下のものは，1993 年 3 月末日で運用を停止し，ソースプログラムを改訂者に返却している．利用を望まれる方は改訂者に連絡されたい．

  – AA01, AA02
  – DA03
  – DB05
  – SHL1 ~ SHL5



$f(x)$ の gradient $g(x)$ の作動確認

14.2 DA02

BCS 方程式 (BCS equation)

プログラム形式	サブルーチンプログラム
作成者 作成年月日	大西 直毅 (東京大学) 1969 年 9 月
改訂者 改訂年月日	上村 正康 (九州大学理学部) 1971 年 9 月
使用言語	Fortran
ソースの公表 使用 OS	'QS.LIBSC(Y3DA02)' MSP(M1800, VP2600)

【概要】

Pairing force が一定の場合の Gap 方程式を解く．即ち coupling constant G と粒子数 N(但し偶数に限る) が与えられているとき，

$$\frac{2}{G} = \sum_{\nu=1}^{\Omega} \frac{1}{\sqrt{(\varepsilon_\nu - \lambda)^2 + \Delta^2}}$$

$$N = \sum_{\nu=1}^{\Omega} \left(1 - \frac{\varepsilon_\nu - \lambda}{\sqrt{(\varepsilon_\nu - \lambda)^2 + \Delta^2}} \right)$$

なる連立方程式を満足する λ (Chemical Potential) と Δ (Gap Energy) を求める．但し，ε_ν は 2 つずつ対に縮退した Single particle Energy で，その level の数は Ω 個である．

【使用法】

1. 呼出方法

   `CALL BEQ(ESP,NOTLT,NPT,GT,EPS3,DELTA,RAMDA,ET,U,V)`

   - 引数による入力

     ESP : ε_v, Single particle Energy (`DIMENSION EPS(100)`).
     NOTLT : Ω, level 数
     NPT : N, 粒子数
     GT : G, coupling constant
     EPS3 : 収束条件，概要の二つの式がどの程度の精度で満足するかを指定

   - 引数による出力

     DELTA : Δ, Gap Energy
     RAMDA : λ, Chemical potential
     ET : E_ν, Quasi-Particle Energy (`DIMENSION ET(100)`).
     U : $U_\nu \equiv \sqrt{\frac{1}{2}\left(1 - \frac{\varepsilon_\nu - \lambda}{E_\nu}\right)}$, u-factor (`DIMENSION U(100)`).
     V : $V_\nu \equiv \sqrt{\frac{1}{2}\left(1 - \frac{\varepsilon_\nu - \lambda}{E_\nu}\right)}$, v-factor (`DIMENSION V(100)`).

- 注意) ε_v の値がレベルの角運動量 j の z 成分 m について縮退している場合，ε_ν の入力は $(2j+1)/2$ ずつ同じ値をくりかえして input することになる.

2. 備考

   N が偶数で，G があまり小さすぎると，trivial な特別解 $\Delta = 0$ 以外に望む解が存在しない場合がある．それは $\Delta = 0$ で $\varepsilon_{\nu-1}$ と ε_ν の間に $x(\Delta, \lambda)$ においての saddle point があり，その点 (Δ_s, λ_s) での x の値が $2/\mathrm{G}$ より小さければ解は存在しない．この解の存在するかどうかはプログラムの最初でチェックしてある．もし，解が存在しない時には

   $$\lambda = \lambda_s \quad , \quad \Delta = 2^{-10}$$

   (但し λ_s は概要の 2 番目の式を満足する $x(\Delta, \lambda)$ の saddle point の λ)

   $$E_\nu = |\varepsilon_\nu - \lambda_s|$$

   $$U_{\nu>\lambda_s} = 1 - 2^{-21} \quad , \quad V_{\nu>\lambda} = 2^{-10}$$
   $$U_{\nu<\lambda_s} = 2^{-10} \quad , \quad V_{\nu>\lambda} = 1 - 2^{-21}$$

   をいれて RETURN する.

【注意事項】

- 本ライブラリを利用して得られた結果を公表する場合，プログラム名と開発者名を明記すること．

14.3 DB01, DB02, DB03

- クレブシュ・ゴルダン係数 (Clebsch-Gordan Coefficient)
- ラカー係数 (Racah Coefficient)
- 9-J 係数 (Nine - J Symbol)

プログラム形式	関数プログラム
作成者	田村 太郎 (東京大学)
改訂者 1 改訂年月日	板倉 守 (東京大学) 1969 年 4 月 23 日
改訂者 2 改訂年月日	上村 正康 (九州大学理学部) 1971 年 5 月 1 日
使用言語	Fortran
ソースの公表 使用 OS	'QS.LIBSC(Y3DB01)' 'QS.LIBSC(Y3DB02)' 'QS.LIBSC(Y3DB03)' MSP(M1800, VP2600)

【概要】
Clebsh-Gordan 係数，Racah 係数及び 9-J Symbol を計算する関数型サブプログラムである．これらのコードは出来るだけ手短かに，また理解し易いように組んだつもりである．これらのコードは，Argument が非常に大きい場合 (大体 200 程度まで) に使えるようにしてある．従って，重い粒子による散乱，反応とか，High-energy projectiles による反応，散乱の場合にも使用することができる．

【使用法】

1. 田村 太郎 氏作成の CLEB，RAC7，NINEJ は，サブルーチン型のサブプログラムであったが，使用頻度，及び使用上の便利のために関数型のサブプログラムに書き直した．また，Argument が適当でない場合には，エラーメッセージを書き出して，その値を 0.0 と出力するようにした．

   これらのプログラムは夫々 Function Type であるから，計算式の途中に Array name と同じように使用でき，単精度実数型の正しい値をそのまま使用することが出来る．次に Argument の並べ方を説明する．

   (a) `FUNCTION CLEB(FJ1,FM1,FJ2,FM2,FJ3,FM3)`
   この Program は，Clebsh-Gordan 係数 (FJ1,FM1,FJ2,FM2 | FJ3,FM3) 即ち $(j_1, m_1, j_2, m_2|j_3, m_3)$ の値を計算するものである．Type は全て単精度実数．夫々の Argument の順序は

Arguments	Physical meaning
FJ1	j_1
FM1	m_1
FJ2	j_2
FM2	m_2
FJ3	j_3
FM3	m_3

(b) `FUNCTION RAC7(FJ1,FJ2,FJ,FJ3,FJ12,FJ23)`
このプログラムは Racah 係数 W(FJ1,FJ2,FJ,FJ3 ; FJ12,FJ23)，即ち $W(j_1, j_2, J, j_3; j_{12}, j_{23})$ の値を計算するものである． Type は全て単精度実数．夫々の Arguments の順序は

Arguments	Physical meaning
FJ1	j_1
FJ2	j_2
FJ	J
FJ3	j_3
FJ12	j_{12}
FJ23	j_{23}

(c) `FUNCTION U9J(FJ1,FJ2,FJ3,FJ4,FJ5,FJ6,FJ7,FJ8,FJ9)`
このプログラムは 9j 係数

$$\left\{ \begin{array}{ccc} FJ1 & FJ2 & FJ5 \\ FJ3 & FJ4 & FJ6 \\ FJ7 & FJ8 & FJ9 \end{array} \right\}$$

即ち

$$\left\{ \begin{array}{ccc} j_1 & j_2 & j_5 \\ j_3 & j_4 & j_6 \\ j_7 & j_8 & j_9 \end{array} \right\} = \left\{ \begin{array}{ccc} j_1 & j_2 & j_{12} \\ j_3 & j_4 & j_{34} \\ j_{13} & j_{24} & J \end{array} \right\}$$

の値を計算するものである． Type は全て単精度実数．それぞれ Argument の順序は

Arguments	Physical meaning
FJ1	j_1
FJ2	j_2
FJ3	j_3
FJ4	j_4
FJ5	j_{12}
FJ6	j_{34}
FJ7	j_{13}
FJ8	j_{24}
FJ9	J

2. エラーメッセージ
   これらのプログラムは Argument の入れ方が適当でないときに次のエラーメッセージを書き出し，出力を 0.0 として Return する．

   - `FUNCTION CLEB(FJ1,FM1,FJ2,FM2,FJ3,FM3)`
     - j_1, j_2, j_3 のどれかの値に負の実数が入っている場合
       `ERROR*THERE ARE SOME NEGATIVE A.M.IN CLEB*J1= ....`
     - $m_1 + m_2 - m_3$ の値が 0 にならない場合
       `ERROR*SUM OF MAGNETIC QUANTUM NUMBER IS NOT ZERO IN CLEB*M1= ....`
     - $j_1 + j_2 + j_3$ が整数にならない場合
       `ERROR*SUM OF ANGULAR MOMENTUM IS NOT INTEGER IN CLEB*J1= ....`
     - j_1, j_2, j_3 が三角条件を満たさない場合
       `ERROR*TRIANGULAR CONDITION IS BIOLATED IN CLEB*J1= ....`
     - $m_j > j_i$ なる Arguments が入っていた場合
       `ERROR*ABS(M.Q.N).GT.A.M IN CLEB*J1=....`
     - $(j_i + m_i)$ が整数にならないものがある場合
       `ERROR*INVARID USE OF (J.M) PAIRS IN CLEB*J1= .....`
   - `FUNCTION RAC7(FJ1,FJ2,FJ,FJ3,FJ12,FJ23)`
     - Arguments のどれかに負の実数が入っている場合
       `ERROR*SOME ARGUMENTS ARE NEGATIVE IN RAC7*J1= ....`
     - Arguments の組に三角条件を満たさないものがある
       `ERROR*INVARID TRIANGULAR CONDITION IN RAC7*J1= .....`
   - `FUNCTION U9J(FJ1,FJ2,FJ3,FJ4,FJ5,FJ6,FJ7,FJ8,FJ9)`
     - Arguments のどれかに負の数が入っている場合
       `ERROR*SOME ARGUMENTS ARE NEGATIVE IN UNINEJ***** J1=....`
     - Arguments の組に三角条件を満たさないものがある
       `ERROR*TRIANGULAR CONDITION IS INVALID IN UNINEJ***** J1= ....`

3. 精度
   有効数字 6 桁以上

4. 制限事項
   これらのプログラムは核反応の計算に使用されることを念頭においてあるので Argument の値が大きい場合多少精度は落ちるようである．テストの項参照．

5. 使用ルーチン
   U9J を使用する場合には RAC7 を同時に使用すること．RAC7，CLEB は夫々単一，独立のルーチンである．

6. 備考
   「CLEB，RAC7 の計算には，$\log_e(n!)$ の和の計算が分子，分母のそれぞれに入るので，Arguments が大きすぎる場合には，オーバーフローする危険がある」と田村氏は注意しているが，普通の使い方をする場合にはほとんどその危険性はないものと思われる．

【使用例】

- $(j_1, m_1, j_2, m_2|j_3, m_3) = (2, 1, 3, -2|4, -1)$ の値が必要な場合には，この値の必要な箇所において

```
CLEBSH=CLEB(2.0,1.0,3.0,-2.0,4.0,-1.0)
```

とおいて CLEB を呼べば CLEBSH には 0.59160817 の値が返ってくる．

- $W(a, b, c, d; e, f) = W(2, 3, 3/2, 5/2; 3, 1/2)$ の値が必要な場合には，この値の必要な箇所において

```
RACAH=RAC7(2.0,3.0,1.5,2.5,3.0,0.5)
```

とおいて RAC7 を呼べば RACAH=-0.4638501 の値が返ってくる．

- $P = (2, 1, 3, -2|4, -1) \cdot W(2, 3, 3/2, 5/2; 3, 1/2)$ の値が必要な場合，この値の必要な箇所において

```
P=CLEB(2.0,1.0,3.0,-2.0,4.0,-1.0)*RAC7(2.0,3.0,1.5,2.5,3.0,0.5)
```

とおいて CLEB と RAC7 を呼べば P には -0.08660257 の値が返ってくる．

- 実数 Q の値として

$$Q = \left\{ \begin{array}{ccc} 1/2 & 2 & 3/2 \\ 3/2 & 3 & 5/2 \\ 2 & 3 & 4 \end{array} \right\} \cdot \left\{ \begin{array}{ccc} 1/2 & 2 & 3/2 \\ 3/2 & 3 & 5/2 \\ 2 & 3 & 3 \end{array} \right\}$$

の値が必要な場合には，この値の必要な箇所において

```
Q=U9J(0.5,2.0,1.5,3.0,1.5,2.5,2.0,3.0,4.0)
 *U9J(0.5,2.0,1.5,3.0,1.5,2.5,2.0,3.0,3.0)
```

とおいて U9J を呼べば Q には -0.00001283 の値が返ってくる.

- 実数 R の値として

$$R = W(2, 3, 3/2, 5/2; 3, 1/2) \cdot \left\{ \begin{array}{ccc} 1/2 & 2 & 3/2 \\ 3/2 & 3 & 5/2 \\ 2 & 3 & 4 \end{array} \right\}$$

の値が必要な場合には，この値の必要な箇所において

```
R=RAC7(2.0,3.0,1.5,2.5,3.0,0.5)
 *U9J(0.5,2.0,1.5,3.0,1.5,2.5,2.0,3.0,4.0)
```

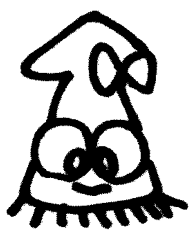
とおいて RAC7 と U9J を呼べば R は 0.00058451 の値が返ってくる.

【注意事項】

- 本ライブラリを利用して得られた結果を公表する場合，プログラム名と開発者名を明記すること．

【参考文献】

- M.E.Rose，Elementary Theory of Angular Momentum.
- A.R.Edmond，Angular Momentum in Quantum Mechanics.
- A.De-Shalit and I.Talmi，Nuclear Shell Theory.
- T.Inoue，Table of the Clebsh-Gordan Coefficients.
- T.Ishidzu et al，Table of the Racah Coefficients.
- K.Smith and J.W.Stevenson，A Table of Wigner 9j Coefficients for Integral and Half Integral Values of the Parameters, ANL-5776 & 5886.



14.4 KNL1

生成座標方式による直交条件模型の積分核 (Kernels of the Orthogonality Condition Model)

プログラム形式	サブルーチンプログラム
作成者 作成年月日	上村 正康（九州大学理学部） 1976 年 7 月
使用言語	Fortran
ソースの公表 使用 OS	'QS.LIBSC(Y3KNL1)' MSP(M1800, VP2600)

【概要】

複合粒子間散乱における直交条件模型の微積分方程式

$$\sqrt{1-K_L}(T_L+V_D^{(N)}(R)+V_D^{(C)}(R)-E_{cm})\sqrt{1-K_L}u_L=0$$

$$T_L=-\frac{h^2}{2\mu}\frac{d^2}{dR^2}+\frac{hL(L+1)}{2\mu R^2}$$

を生成座標方式の試行関数を用いて Kohn-Hulthen 変分原理によって解く (VAR1 を用いる) とき必要な積分核 (kernel)

$$H_L(S_i,S_j)=\int_0^\infty\int_0^\infty\Gamma_L(R,S_i)\sqrt{1-K_L}(T_L+V_D^{(N)}+V_D^{(C)})\sqrt{1-K_L}\Gamma(R',S_j)dRdR'$$

$$N_L(S_i,S_j)=\int_0^\infty\int_0^\infty\Gamma_L(R,S_i)(1-K_L)\Gamma_L(R',S_j)dRdR'$$

$$\Gamma_L(R,S)=Re^{-\lambda(R^2+S^2)}I_L(2\lambda SR):\ I_L(z)\text{ は modified Bessel}$$

を計算するサブルーチンプログラム．

$\sqrt{1-K_L}$ を Λ_L でおきかえたものも求めることができる．周知のように結果の phase shift は同じである．$1-K_L$ $(=\delta_L(R,R')-K_L(R,R'))$ は norm kernel，Λ_L $(=\Lambda_L(R,R'))$ はパウリ原理による相対運動の禁止状態を排除する operator．$\alpha-\alpha$, $\alpha-^{16}O$, $^{16}O-^{16}O$, $\alpha-^{40}Ca$ 系については $1-K_L$ を内蔵している．その他の場合については，適切に (適当に) 禁止状態の数を入力して Λ_L を用いて解くことになる．$V_D^{(N)}(R)$, $V_D^{(C)}(R)$ は nuclear direct potential および coulomb direct potential. これには微視的に求めた結果を用いることも，また現象論的に適当な関数形を与えることもできる．上記 4 系については微視的に求めたもの (H.O. 型 L-S 閉殻を仮定) を内蔵しているので有効 = 体力を指定すれば計算する．ただし有効 = 体力としては，Volkov No.1, No.2 型，Brink-Boecker 型，Hasegawa-Nagata-Yamamoto 型を内蔵している．

【使用法】

1. 呼出方法

```
CALL OCM1(FORCE1,PARAME,VDNFNC,VDCFNC,AM1,AM2,KAKU,L,ALAM,
          NDX,SS,IMAX,HH,ANN,NDIM)
```

2. パラメータ

以下において，I, L, N で始める変数はすべて整数型変数であり，その他はすべて倍精度実数型変数である．

AM1, AM2 : 散乱する 2 つの核の質量数．^{16}O の質量を 16 とする単位
1 質量 $\times C^2 = 931.478 MeV$. 1 と 2 の順不問

KAKU : 内蔵している $1 - K_L$ 計算用サブルーチンを働かせるための指示
KAKU=11 : $\alpha - \alpha$
KAKU=12 : $\alpha -^{16} O$
KAKU=22 : $^{16}O -^{16} O$
KAKU=13 : $\alpha -^{40} Ca$
これ以外の場合は KAKU=0 と指定し，下の NDX を与えることにより Λ_L を作る．

L : 角運動量 L

ALAM : ガウス変換にあらわれるサイズパラメータ λ．単位 fm^{-2}
1 体調和振動子のサイズパラメータを $\beta = mw/h$ とするとき通常
$\lambda = \beta/2 * AM1 * AM2/(AM1 + AM2)$ の値をとっておく．
この λ をサイズパラメータとする調和振動子波動関数 ($\propto e^{-\frac{\lambda}{2}R^2}$)
φ_{NL} を用いて Λ_L 作ったり $1 - K_L$ を展開したりする．

NDX : KAKU=0 指定の時
Λ_L を作る際に要する禁止状態の数 (L に依る) を表す．
ないときは 0 とする．$\Gamma_L = \sum_{N=0}^{NDX-1} |\varphi_{NL} >< \varphi_{NL}|$
KAKU≠0 指定の時
$\sqrt{1 - K_L}$ を調和振動子波動関数で展開するときの個数を表す．
すなわち，$\sqrt{1 - K_L} = \sum_{N=0}^{NDX-1} \gamma_N |\varphi_{NL} >< \varphi_{NL}|$
KAKU≠0 指定のときでも KAKU=0 指定のときと同様に NDX として
禁止状態の数を入力すると Λ_L を用いる kernel が求まる．
NDX $\leq$ 30. NDX を禁止状態の数と一致させるならば $\sqrt{1 - K_L}$ の
替わりに Λ_L を用いることになる．

HH : kernel $H_L(S_i, S_j)$ を表す `HH(I,J)` なる配列．

ANN : kernel $N_L(S_i, S_j)$ を表す `ANN(I,J)` なる配列．

NDIM : 正方配列 `HH`, `ANN` のメインプログラムで宣言した大きさ．NDIM $\leq$ 20

FORCE1 : $V_D^{(N)}(R)$ を微視的に求める際に用いられる二体核力の名前を表す function 副プログラム名．内蔵している 4 種の二体核力には次の名前がつけてある．
Volkov No.1 — VOLKV1
Volkov No.2 — VOLKV2
Brink Boecker — BRINK
Hasegawa-Nagata-Yamamoto — HNY
`EXTERNAL VOLKV1` 等の宣言が必要である．
微視的に $V_D^{(N)}(R)$ を計算しない場合 (woods-saxon を用いるなど) は，
これに備えて何もしない `FUNCTION FREE(PARAME)` なるものを
内蔵しているので，`EXTERNAL FREE` 宣言をして，FORCE1 に FREE を
当てはめると良い．上記以外の二体核力を用いる場合は，
`FUNCTION FORCE1(PARAME)` の形式で定義 (倍精度) 作成すること．

		`EXTERNAL FORCE1` 宣言がいる
PARAME	:	上の `FUNCTION FORCE1(PARAME)` の引数． FORCE1 が VOLKV1, VOLKV2 の場合は，majorana mixture parameter の m を表す．HNY の場合は第 2 range potential の ^{31}P 成分 (singlet-even) の強さを増減する parameter．単位は MeV，引力を追加するならば `PARAME < 0`．BRINK の場合は free parameter はないので任意の数をいれておけばよい．FREE の場合も任意の数を入れておけばよい． 一般に FORCE1 を自作する場合は，1 つの parameter はこの `PARAME` を用いても良いが，他の parameter については，メインプログラムと FORCE1 及び次の $V_D^{(N)}(R)$ を表す `VDNFNC` の 3 つのプログラムの間を COMMON でつなぐ必要がある．
VDNFNC VDCFNC	:	$V_D^{(N)}(R)$，$V_D^{(C)}(R)$ を表す FUNCTION 副プログラム名． `FUNCTION VDNFNC(R), VDCFNC(R)` の形式で定義 (倍精度) 作成すること メインプログラムで `EXTERNAL VDNFNC, VDCFNC` の宣言が必要である． このプログラムには，$\alpha-\alpha$, $\alpha-^{16}O$, $^{16}O-^{16}O$, $\alpha-^{40}Ca$ 系については H.O. 閉殻核波動関数を用いて微視的に求めるものが内蔵されている． FUNCTION 名は，$\alpha-\alpha$ では，VDNAA，VDCAA， $\alpha-^{16}O$ では VDNAO，VDCAO，$^{16}O-^{16}O$ では VDNOO, VDCOU, $\alpha-^{40}Ca$ では VDNACA, VDCACA である．

3. 途中出力
   出力リストのはじめのページに次が出力される．
   (a) $R = 0.25fm$ ~ $10.0fm$, $\Delta R = 0.25fm$ で $V_D^{(N)}(R)$，$V_D^{(C)}(R)$ の値．
   (b) norm kernel $1 - K_L = \sum_{N=0}^{NDX-1} \lambda_N |\varphi_{NL} >< \varphi_{NL}|$ とするときの λ_N ($N = 0, \cdots, NDX-1$) の値．
   (c) 途中の数値積分の誤差の目安のため同一 mesh で
   $$\int_0^\infty \varphi_{NDX-1\ L}(R)\varphi_N(R)R^2dR, \quad (N = 1, \cdots, NDX-1)$$
   を数値積分した値が出る．($N = NDX-1$ のとき 1 を引いたもの). 0 からのずれが誤差の目安を与える．

4. 精度
   有効数字 10 桁程度 (HH, ANN について).

14.5 VAR1

微積分方程式または積分方程式の変分原理による散乱境界条件解 (Scattering Solution of Integro-Differential or Integral Equation by Using Variational Principle)

プログラム形式	サブルーチンプログラム
作成者 作成年月日	上村 正康，見戸 義英 (九州大学理学部) 1976 年 7 月
使用言語	Fortran
ソースの公表 使用 OS	'QS.LIBSC(Y3VAR1)' MSP(M1800, VP2600)

【概要】
Kohn-Hulthén 変分原理により，次の形の微積分方程式または積分方程式を散乱境界条件のもとに解き，散乱行列 S-matrix を求めるプログラムである．解法は作成者による参考文献で述べてある．ここで扱う方程式は微積分方程式

$$\mathcal{L}_L \cdot u_L \equiv \int_0^\infty \mathcal{L}_L(R,R')u_L(R')dR'$$

$$\begin{aligned}\mathcal{L}_L(R,R') &= \left[-\frac{h^2}{2\mu}\frac{d^2}{dR^2} + \frac{h^2L(L+1)}{2\mu R^2} + V_D^{(N)}(R) + V_D^{(C)}(R) - E_{cm}\right]\delta(R,R') \\ &+ G_L(R,R') - E_{cm}N_L^{(ex)}(R,R')\end{aligned}$$

またはこれと

$$\int \Gamma_L(R,S)\mathcal{L}_L(R,R')\Gamma_L(R',S')dR\,dR' = H_L(S,S') - E_{cm}N_L(S,S')$$

$$u_L(R) = \int_0^\infty \Gamma_L(R,S)f_L(S)dS$$

$$\Gamma_L(R,S) = Re^{-\lambda(R^2+S^2)}I_L(2\lambda SR)$$

の関係にある (等価) Hill-Wheeler 型積分方程式

$$\int_0^\infty [H_L(S,S') - E_{cm}N_L(S,S')]\,f_L(S')\,dS' = 0$$

である．$I_L(z)$ は modified Bessel function. 境界条件は

$$u_L(0) = 0, \quad \kappa R \gg 1 \text{ で} u_L(R) \sim [G_L(\kappa R) - iF_L(\kappa R)] - e^{2i\eta_L}[G_K(\kappa R) + iF_L(\kappa R)]$$

で与える．ここに，$F_L(\kappa R)$, $G_L(\kappa R)$ はクローン関数，$h^2\kappa^2/2\mu = E_{cm}$. $e^{2i\eta L}$ は散乱 S 行列．

試行関数は

$$\tilde{u}_L = \sum_{i=1}^{i_{max}} \tilde{c}_i\Gamma_L(R,S_i) + u_L^{(-)}(\kappa,R) - \tilde{S}_L u_L^{(+)}(\kappa,R)$$

の形で与える [$\kappa R \gg 1$ で $u_L^{(\pm)}(\kappa,R) \sim G_L(\kappa R) \pm iF_L(\kappa R)$，核内での形は文献参照]． mesh 点 S_i の適切な取り方は文字どおり "試行" 経験によるが，その目安は文献の例題によって読みとることができよう．

このプログラムでは $H_L(S_i,S_j),\ N_L(S_i,S_j)\ (i,j\ =\ 1,\cdots,i_{max})$ を入力する必要がある．この Kernel(生成座標形式) は元の多体 Hamiltonian から直接求められている場合もあろうし (Brink 流)，dynamical 座標で表された $\mathcal{L}_L(R,R')$ からガウス交換を通して求められている場合もあろう．

核力とクーロン力による direct potential $V_D^N(R),\ V_D^C(R)$ の関数形が分かっていると，計算が speed up される (mesh 点 S_i の外側がいくつか省ける) のでこれも入力すると良い．

このプログラムでは，散乱 S 行列の汎関数

$$J[\tilde{u}_L] = \tilde{S}_L + \frac{i\,\mu}{h\kappa}\int_0^\infty\int_0^\infty \tilde{u}_L(R)\mathcal{L}_L(R,R')\tilde{u}_L(R')dR\,dR'$$

を，変分パラメータ $\tilde{C}_i,\ \tilde{S}_L$ を動かすことによって停留させ，S 行列の停留点 $(\bar{S}_L)$ と停留値 (S_L) を求める．前者は，試行関数の誤差について 1 次，後者は 2 次の order である (後者の方が精度が高い)．

文献で詳述したように，$\bar{S}_L$ と S_L との差，および $|\bar{S}_L|$ と $|S_L|$ との差の 1(unitarity) からのずれが計算の精度つまり試行関数 (mesh 点 S_i) の適切さを判定するために非常に有効である．このように，計算の信頼性を自己判断できる形になっているのがこのプログラムの大きな利点の 1 つである．

このプログラムの出力は

1. $\bar{S}_L,\ S_L,\ |\bar{S}_L|,\ |S_L|$ と $\bar{S}_L = |\bar{S}_L|e^{2i\bar{\delta}_L},\ S_L = |S_L|e^{2i\delta_L}$ から求まる real phase shift $\bar{\delta}_L,\ \delta_L$
2. Bound-state 近似による方程式の固有エネルギー ($\sum_{j=1}^{i_{max}}[H_L(S_i,S_j)-E_{cm}N_L(S_i,S_j)]f_j = 0$ という固有方程式の対角化)

である．

次の拡張は別のプログラムとして登録される

- Pauli 原理による禁止状態を排除した形で波動関数を求めること
- $V_D^{(N)}(R)$ や $G_L(R,R')$ を複素ポテンシャルにすること
- $G_L(R,R')$ や $p'N_L^{(ex)}(R,R')$，$V_D^{(N)}(R)$，$V_D^{(C)}(R)$ を "関数形" または "数値" で与えれば自動的に $H_L(S_i,S_j),\ N_L(S_i,S_j)$ を求めて解を出すこと．

なお，このコード開発については，九州大学大型計算機センターライブラリ開発及び昭和 50 年度重イオン科研費 "重イオン科学" より計算費その他の援助を頂きました．

【使用法】

1. <u>呼出し方法</u>

```
CALL KOHN1(VDNFNC,VDCFNC,HH,ANN,NDIM,AM1,AM2,ZZ,L,ALAM,SS,IMAX,
           EMIN,ED,NEMAX,KBOUND,ICHECK,PHASE,PHASEJ,SABS,SABSJ)
```

2. <u>パラメータ</u>

   以下において I, K, L, N で始まる変数は全て整数型変数であり，その他は倍精度実数型変数もしくは同配列名である．

AM1, AM2

散乱する 2 つの核の質量数．^{16}O の質量数を 16 とする単位．
1 質量数 $\times C^2 = 931.478 MeV$．1 と 2 の順不問．

ZZ

2 つの核の電荷数の積．

L

角運動量 L．

ALAM

ガウス変換と試行関数に用いられているサイズパラメータ λ．単位 fm^{-2}．
1 体調和振動子のサイズパラメータを $\beta = mw/h$ とするとき，通常，$\lambda = \beta/2 * AM1 * AM2/(AM1 + AM2)$ 程度の値をとっておく．ただし，kernel $H_L(S_i, S_j)$, $N_L(S_i, S_j)$ が多体ハミルトニアンと反対称化によって微視的に導出されている時は，必ず上の関係がいる．

SS

試行関数の $\Gamma_L(R, S_i)$ と kernel $H_L(S_i, S_j)$, $N_L(S_i, S_j)$ におけるきざみ点 S_i を表す `SS(I)` なる配列．単位 fm．最外側の $R_C = (S_{i_{max}-1} + S_{i_{max}})/2$ の点で漸近形につなぐので $S_{i_{max}}$ の位置に注意を要する．(文献及び注意事項参照)

IMAX

S_i の上限を示す i_{max}．$S_1 < S_2 < \cdots < S_{i_{max}}$ と入力すること．

EMIN, ED, NEMAX

重心系入射エネルギー E_{cm} を変化させる際の，それぞれ，出発値，きざみ（負でも可），きざみの回数 (1 点なら 1)．単位 MeV．

KBOUND

Bound state 近似を用いて入力した kernel のエネルギー固有値を求めるかどうかの指定．

0：求めない．
1：求める．

ICHECK

$S_{i_{max}}$ つまり `SS(IMAX)` のとり方の適，不適をチェックするための計算をし，出力を行うかどうかの指定．

ICHECK=0：行わない．
ICHECK=1：行う．

計算されるのは energy surface $H_L(S_i, S_j)/N_L(S_i, S_j)$ とこれに相当する direct part の energy surface (3 の注意事項参照)

PHASE, PHASEJ

求まった phase shift．単位は degree で $0^\circ \sim 180^\circ$ の間の値で出す．PHASE= $\bar{\delta}_L$, PHASEJ $= \delta_L$．(後者の方が精度が良い，試行関数の誤差に対して 2 次の order, 前者は 1 次の order)．したがって $|\bar{\delta}_L - \delta_L|$ から誤差の 1 次の order が読みとれよう．`PHASE(I)`, `PHASEJ(I)`, $(I = 1, \cdots, \mathrm{NEMAX})$ なる配列．NEMAX 回のエネルギー変化による $\bar{\delta}_L$, δ_L を貯めたものが返ってくる．

SABS, SABSJ

求まった S 行列の絶対値．計算が 100% exact なら 1(unitarity) になるべき量である．変分法ではアプリオリには 1 にはならないが，逆にそれからのずれから計算の信頼性を読みとることができる．SABS=$|S_L|$ の 1 からのずれが誤差の 1 次の order，SABSJ=$|\bar{S}_L|$ の 1 からずれが誤差の 2 次の order，$\bar{S}_L = |\bar{S}_L|e^{2i\bar{\delta}_L}$, $S_L = |S_L|e^{2i\delta_L}$. `SABS(I)`, `SABSJ(I)`, $(I = 1, \cdots, \mathrm{NEMAX})$ なる配列．

VDNFNC, VDCFNC

nuclear direct potential $V_D^{(N)}(R)$ および Coulomb direct potential を表す FUNCTION 副プログラム名．`FUNCTION VDNFNC(R)` および `FUNCTION VDCFNC(R)` の形式で定義（倍精度），作成すること．メインプログラムで `EXTERNAL VDNFNC, VDCFNC` の宣言が必要である．この $V_D^{(N)}(R)$，$V_D^{(C)}(R)$, の関数形を入力しなくとも，$H_L(S_i, S_j)$, $N_L(S_i, S_j)$ が計算できていれば問題を解くことができる．この場合は VDNFNC, VDCFNC として必ず，それぞれ `VN000`, `VC000` という FUNCTION 名を用い，`EXTERNAL VN000, VC000` 宣言をすれば良い．

`FUNCTION VN000(R)` と `VC000(R)` はそれぞれ 0，$ZZ'e^2/R$ であり (計算に必要な領域は $R > R_C$)，このプログラムに内蔵されている (3 の注意事項と備考参照)．

HH

kernel $H_L(S_i, S_j)$ を表す `HH(I,J)` なる配列．intrinsic energy 部分 E_{in}・$\dot{N}_L(S_i, S_j)$ を繰り込んでいることに注意．

ANN

Norm Kernel $N_L(S_i, S_j)$ を表す `ANN(I,J)` なる配列．$N_L(S_i, S_j)$ にかかっている係数は全エネルギーでなく $E_{cm}(= E_{total} - E_{in})$．

NDIM

正方配列 HH, ANN のメインプログラムで宣言した大きさ．NDIM ≤ 20

3. 途中出力および注意事項

試行関数のとり方 (つまり SS(I), I=1, $\cdots$, IMAX のとり方）の適切さを判断するのに役立つ途中出力について述べる．上記の ICHECK=1 の場合．最初の出力は L, AM1, AM2, ZZ, ALAM, IMAX, (SS(I), I=1, $\cdots$, IMAX), RC, WIDTH, HALF-WIDTH である．RC は 上記 $R_C = (SS(IMAX-1) + SS(IMAX))/2$，WIDTH は $e^{-\lambda(R-S_i)^2}$ が e^{-16} に減少する $|R - S_i|$ の値，HALF-WIDTH は同じく 1/2 に減少する $|R - S_i|$ の値．単位 fm．S_i の間隔が後者よりも大きくなると試行関数はかなり悪くなる．($\Gamma_L(R, S_i)$ はほぼ R・$e^{-\lambda(R-S_i)^2}$ のような形をしている)．

次の出力は‘`ENERGY SURFACE CHECK`’である．ここでは，S, E, ED, N, C, T, HH, ANN, () という文字に続いてそれぞれ数字が出力される．IMAX 行にわたる．$S = S_i$, E は energy surface $E = H_L(S_i, S_j)/N_L(S_i, S_j)$ ED は direct part に関する energy surface, つまり

$$ED = \frac{\displaystyle\int_0^\infty \Gamma_L(R, S_i)\left[-\frac{h^2}{2\mu}\frac{d^2}{dR^2} + \frac{h^2 L(L+1)}{2\mu R^2} + V_D^{(N)}(R) + V_D^{(C)}(R)\right]\Gamma_L(R, S_i)\, dR}{\displaystyle\int_0^\infty \Gamma_L(R, S_i)\Gamma_L(R, S_i)\, dR}$$

$S_i = S_{i_{max}}$ で $|E - ED| \lesssim 1/4MeV$ となっておれば $S_{i_{max}}$ の場所は十分適切といえよう（$G_L(R, R'), N_L^{(ex)}(R, R')$ の寄与が無視できることを示す．）$|E - ED|$ がほぼ 0 であ

る必要がない理由（利点）は文献で述べてある（試行関数の $R \gtrsim R_C$ 近傍の作り方の巧妙さによる）．$V_D^{(N)}(R)$，$V_D^{(C)}(R)$ の関数形が入力してない場合（VN000，VC000 を使った場合）は ED の計算では $V_D^{(N)}(R) = 0$，$V_D^{(C)}(R) = ZZ'e^2/R$ とおく．

$|E - ED| \lesssim 1/4MeV$ を満たせば，$R > R_C$ での $G_L(R, R')$，$N_L^{(ex)}(R, R')$，$V_D^{(R)}(R)$，$V_D^{(C)}(R) - ZZ'e^2/R$ の寄与が無視できることを示す．

続いて N は ED の内 $V_D^{(N)}(R)$ からの寄与，C は $V_D^{(C)}(R)$ からの寄与，

T は $-(h^2/2\mu)\dfrac{d^2}{dR^2} + h^2L(L+1)/2\mu R^2$ からの寄与を表わす．E, ED は核外で T+C に近づき，$S_i \to \infty$ で $T = \dfrac{3}{4}\dfrac{h^2\lambda}{\mu}(= \dfrac{3}{4}hw)$ になるはずのものである．HH=$H_L(S_i, S_i)$, ANN=$N_L(S_i, S_i)$, 続く（ ）内は Norm Kernel の direct part (ED 式の分母) を表す．

KBOUND=1 の時の出力は，まず参考のため $N_L(S_i, S_j) = ANN(I, J)$ の行列の固有値が出力される．次に E_{cm} に相当するエネルギー固有値が IMAX 個低いほうから出力される．単位は MeV．

4. 精度
試行関数の取り方に依存する．具体例は文献にある．計算の信頼性チェックについては既に述べた．

【備考】

- $V_D^{(N)}(R)$，$V_D^{(C)}(R)$，を表す FUNCTION VDNFNC(R)，VDCFNC(R) として $\alpha - \alpha$，$\alpha -^{16} O$，$^{16}O -^{16} O$，$\alpha -^{40} Ca$ 系については H.O. 閉殻核波動関数を用いて微視的に求めたものがこのプログラムには内蔵されている．それぞれの場合について $V_D^{(N)}(R)$ に関しては，VDNAA，VDNAO，VDNOO，VDNACA という名前が付いており，$V_D^{(C)}(R)$ に関しては，VDCAA，VDCAO，VDCOO，VDCACA という名前が付いている．これらの $V_D^{(N)}(R)$ の計算には二体核力の情報が必要である．

- Volkov No.1，No.2，Brink-Bolcker，Hasegawa-Nagata-Yamamoto ポテンシャルについては，このプログラムに内蔵されている．それぞれ倍精度実数型関数（引数も倍精度として）FUNCTION VOLKV1(PARAME),VOLKV2(PARAME),BRINK(PARAME),HNY(PARAME) という FUNCTION 名がつけてある．引数 PARAME はこれらのポテンシャルに現れる．パラメータで，Volkov No.1，No.2 では Majorana mixture パラメータ m を表わし，Hasegawa-Nagata-Yamamoto では第 2 range ポテンシャルの ^{31}P 成分 (singlet-even) の強さを増減するパラメータ．単位は MeV．引力を追加するなら PARAME < 0.

  Brink-Bolcker では free parameter はないので任意の値でよい．CALL KOHN1 文の前で，1 回これらの二体核力用 FUNCTION を呼んでおく必要がある．これにより中で定義されている COMMON を通して必要な情報が VDNAA 等に送られる．なお，KNL1 の OCM1 で $H_L(S_i, S_j)$，$N_L(S_i, S_j)$ を作る場合その中で既に上記の COMMON パラメータ設定は済んでいる．

【注意事項】

- 本ライブラリを利用して得られた結果を公表する場合，プログラム名と開発者名を明記すること．

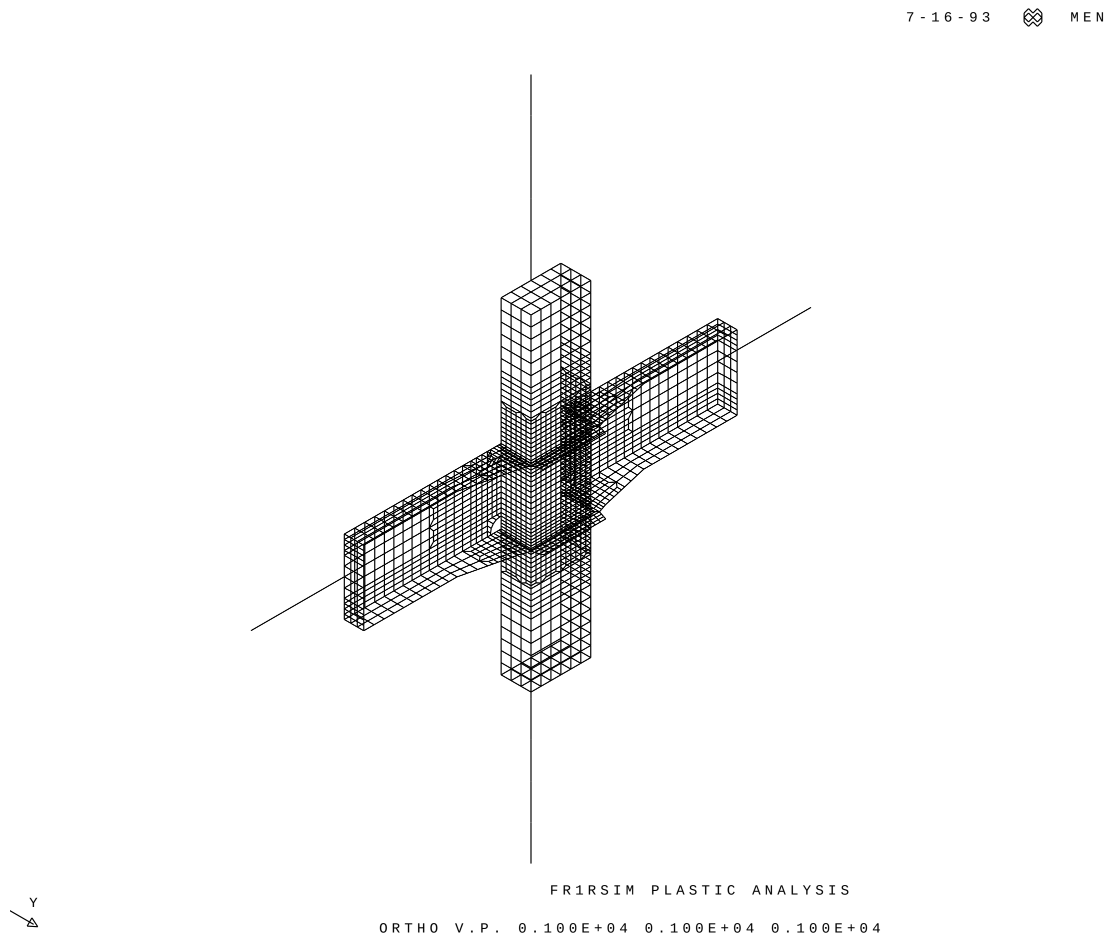
【参考文献】

- Y.Mito and M.Kamimura , Prog. Theor. Phys. 56, No.2, pp.583–598, 1976.

15

有限要素法

ライブラリ名	特徴	MSP	WS
AACOUST	建築音響解析ライブラリ	○	
MARC	構造解析ライブラリ	○	
MENTAT	MARC のプリ / ポスト処理，SUN4 上の動作		○



15.1 AACOUST

建築音響解析ライブラリ AACOUST

プログラム形式	サブルーチン副プログラム
作成者 作成年月日	大鶴 徹 (大分大学工学部建設工学科) 1993 年 3 月 1 日
ソースの公表	当面公表しない
使用 OS	MSP(M1800, VP2600)

【概要】
AACOUST(subroutine program library for Architectural Acoustics) は，建築音響関連分野での使用を主眼とする有限要素解析のためのサブルーチンプログラムライブラリ．現在の Ver.1.0 には，室や弾性壁体に関わる基礎的な解析に用いる各マトリクス算定サブルーチン 3 種と，室の外形から各要素の節点の位置情報を算定するサブルーチン 1 種，並びに，それらの算定の際に用いるスレーブサブルーチンから構成される．複雑なメッシュ分割の処理等に関しては，ワークステーションやパソコン等グラフィクスによるインターフェースの充実した機器を用い，要素のメッシュデータ等を作成した後，そのデータと本ライブラリを利用して VP 上で大型数値解析を実施してもよいでしょう．しかし単純な 2 または 3 次元メッシュの解析では，本ライブラリのサブルーチンのみによった解析の方が容易となる場合もある．

サブルーチン名	機能
KEMEP4	4 節点 12 自由度板曲げ要素の算定
KEAIR8	8 節点 8 自由度音響要素の算定
MEAIR8	8 節点 8 自由度アイソパラメトリック音響要素の算定
NODE8	6 面体の等分割メッシュ生成

AACOUST サブルーチンの機能

【注意事項】

- 本プログラムを使用して得られた結果を公表する場合，プログラム名を明記すること．

【参考文献】

- 「建築音響解析ライブラリ AACOUST・利用の手引」大鶴 徹，九州大学大型計算機センター広報, Vol.26, No.5, pp.549–554, 1993.

15.2 MARC, MENTAT

【概要】
MARC プログラムは，NASTRAN, ADINA と同様，今日世界的に広く利用されている汎用有限要素法解析プログラム．構造解析をはじめとして，熱伝導解析，音響解析，静電場解析などの機能をもっており，豊富なライブラリ群から必要なものを選択して自由度の高い解析処理を行なうことが出来る，特に非線形解析が精度良く行なえるのが特徴．

MENTAT システムは MARC プログラムのための会話型プリ・ポストプロセッサであり，グラフィック・ディスプレイ上でインタラクティブな入力データの作成・編集および解析結果の表示が出来る．MARC プログラムはベクトルプロセッサに，MENTAT システムはワークステーションにインストールされているので，両機間でのデータ転送が必要である．

【処理環境】

- MSP/VP2600
  cataloged procedure : `MARCGO`, `MARCCLG`
- qviss(SUN4)

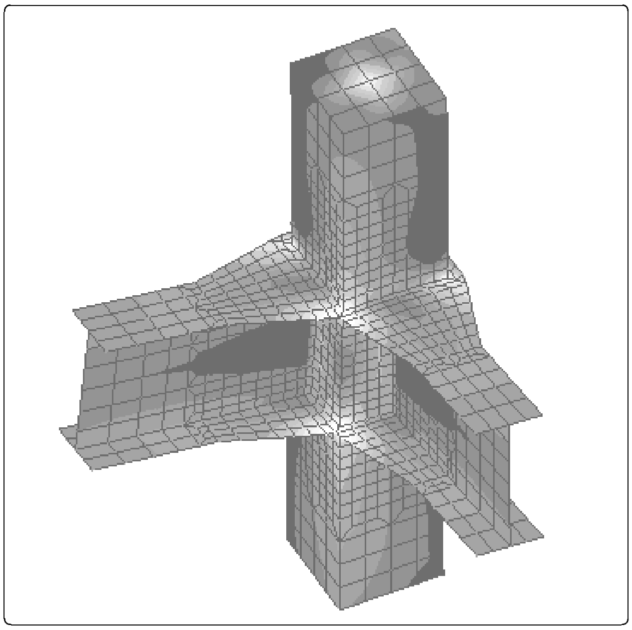
【使用例】

- `MARC1.DATA` よりデータを入力して `MARC` プログラムを起動する．

```
//A79999AA JOB CLASS=V
//   EXEC MARCGO
//MARC.SYSIN DD DSN=A79999A.MARC.DATA,DISP=SHR
//
```

- 利用者のサブルーチン `MARC.FORT` を利用して `MARC` の解析を行う．作業用ファイルとして `SSU(System Storage Unit)` を割り当てる．

```
//A79999AB JOB CLASS=V
//    EXEC MARCCLG
//FORT.SYSIN    DD DSN=A79999A.MARC.FORT,DISP=SHR
//MARC.SYSIN    DD DSN=A79999A.MARC.DATA,DISP=SHR
//MARC.FT11F001 DD UNIT=SSU,SUBSYS=(VPCS,'SPACE=10M'),DISP=(NEW,DELETE)
//MARC.FT12F001 DD UNIT=SSU,SUBSYS=(VPCS,'SPACE=15M'),DISP=(NEW,DELETE)
//MARC.FT13F001 DD UNIT=SSU,SUBSYS=(VPCS,'SPACE=10M'),DISP=(NEW,DELETE)
//
```



MARC/MENTAT による構造解析

【提供】

- 日本 MARC 株式会社

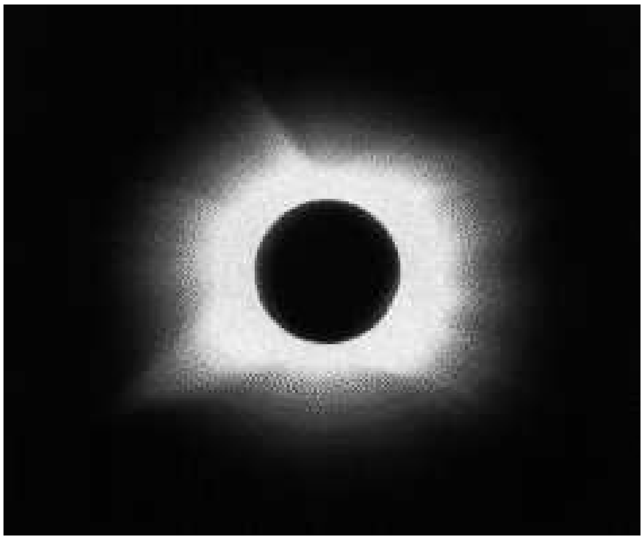
【参考文献】

- 「S.S.Rao, The Finite Element Method in Engineering 2nd Edition, Pergamon Press, 1989」
- 「日本マーク，MARC プログラムユーザ・マニュアル（日本語）A 編 – プログラム機能，B 編 – 要素ライブラリ，C 編 – 入力データ，D 編 – ユーザ・サブルーチン，1990」
- 「MARC/MENTAT 利用の手引き」山成 實, 九州大学大型計算機センター広報, Vol.25, No.1, pp.1-31, 1992.
- 「SSU の MARC による性能評価」 山成 實, 九州大学大型計算機センター広報, Vol.26, No.4, pp.485-487, 1993.
- 「利用の手引・バッチジョブ編」九州大学大型計算機センター・ライブラリ室, 1992.

16

Other Application

ライブラリ名	特徴	MSP	UXP
BOB73	Ge(Li) ガンマ線スペクトル解析	○	
FSPICE	電子回路解析	○	
FSPICE/VP	FSPICE の VP 版	○	
PROPATH	流体の熱物性値プログラム・パッケージ	○	○
UNICS	結晶構造解析プログラムシステム	○	



16.1 BOB73

Ge(Li) ガンマ線スペクトルの解析 (The Analysis of Ge(Li) Gamma-Ray Spectra)

プログラム形式	コンプリートプログラム
作成者 作成年月日	馬場 宏，岡下 宏，馬場 澄子 (日本原子力研究所) 1973 年
改訂者 作成年月日	高島 良正，中山 祐輔，百島 則幸，大崎 進（九州大学理学部） 1978 年 2 月 1 日
使用言語 ソースの公表 使用 OS	Fortran 'QS.LIBSC(Y0BOB73)' MSP(M1800)

【概要】

- Ge(Li) 半導体検出器によるγ線スペクトルデータの解析用プログラムである．
- ガウス曲線に最小自乗近似を行なっている．
- 日本原子力研究所製造部 馬場 宏 らが作成した．
- 九州大学理学部 高島 良正らが九州大学大型計算機センター M1800 で利用できるように改訂した．
- 原プログラムの計算は全て可能である．
- 計算機の機種および使用機器の機種の相違のため，一部修正した．
- 原プログラムに含まれていないデータセットからの測定データ入力を可能にした．

使用方法に関しては，参考文献を参照，

【使用例】

- データを直接入力する

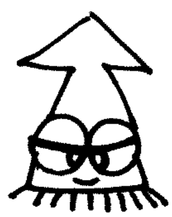
```
//A79999A1 JOB CLASS=A
//  EXEC GO,PROG=BOB73
//GO.SYSGO DD *
      データの入力
/*
//
```

- 測定データをデータセット D1.DATA から入力する．

```
//A79999A1 JOB CLASS=A
//  EXEC GO,PROG=BOB73
//GO.FT01F001 DD DSN=A79999A.D1.DATA,DISP=SHR
//GO.SYSGO DD *
      計算条件設定のデータ入力
/*
//
```

【参考文献】

- Usage of the BOB7-series Programs for the Analysis of Ge(Li) Gamma-Ray Spectra, 日本原子力研究所放射性同位体製造資料 -1200, JAERI-M70117, March, 1977.



16.2 FSPICE, FSPICE/VP

【概略】

FSPICE(Fujitsu Simulation Program for Integrated Circuit Evaluation) は電子回路解析のためのアプリケーションプログラム．カルフォルニア大学で開発された回路解析プログラム SPICE にいくつかの機能を追加している．解析結果は GRAPHMAN によって図形出力される．`FSPICE` コマンドは M1800 のみの動作であるが，バッチジョブによって VP2600 上で高速な処理が可能である．

【処理環境】

- MSP/M1800, MSP/VP2600
  command : `FSPICE`
  catalogued procedure : `FSPICE, FSPICEVP`

【使用例】

- 図形出力を行う出力制御文により出力された解析結果の図形を `PDB` に格納する．`FSPICE` での処理後，`GRAPHMAN` を起動して図形出力を行う．

```
 READY
FSPICE GRPH GRA1(G1.DATA) GRA2(G2.DATA) M(MACGO) IND(IN.DATA) +    ↵
続きを入力してください．
F(FILE.DATA) INC(INC.DATA)    ↵

<<<FSPICE>>>
     -
     -
   解析結果
     -
 READY
```

【提供】

- 富士通株式会社

【参考文献】

- 「FACOM OSIV/F4 FSPICE 解説書」(78AR-7700)，富士通株式会社．
- 「FSPICE（汎用電子回路解析プログラム）の公開について」九州大学大型計算機センター広報, Vol.22, No.5, pp.517–524. 1989.
- 「利用の手引・バッチジョブ編」九州大学大型計算機センター・ライブラリ室, 1992.
- 「利用の手引・ MSP コマンド編」九州大学大型計算機センター・ライブラリ室, 1993.

16.3 PROPATH

PROPATH は関数型の汎用熱物性値プログラムパッケージ．利用者が自身のプログラム単位において物質名を指定すれば，その物質に対する各種の熱物性値が指数関数や三角関数のような簡便さで求められる．1984 年にプロパス第 1.1 版を公開し 1993 年に 8.1 版が公開された．熱物性値や定数を与えるプログラムはすべて関数である. それらの引数は単精度実数か文字型であり, 関数の値は単精度実数か文字型である.

プログラム形式	関数型プログラム
作成者	PROPATH Group 代表：伊藤 猛宏（九州大学工学部）
作成年月日	1984 年 (1.1 版)
使用言語	Fortran
使用 OS	MSP, UXP(M1800)

PROPATH 第 8.1 版では，新たな物質として，水 (IFC 1967-ITS 1990)，六ふっ化硫黄，FC-C318(フロン C318)， CFC-113(フロン 113)， HCFC-123(フロン 123)， HFC-134a(フロン 134a)，ハロン 1211(フロン 12B1)，フロン 500，フロン 503，湿り空気 (理想気体混合物) および湿り空気 (実在流体) の関数を追加し，第 7.1 版までに収録されていた物質についても関数の数を若干追加した．第 8.1 版のライブラリ・データセット名およびシングルショット・プログラムのソース形式のデータセット名は，次のようになっている．

ヘリウム 4	Helium 4	`LIB.P8HE4.LOAD`
ネオン	Neon	`LIB.P8NE.LOAD`
アルゴン	Argon	`LIB.P8ARG.LOAD`
クリプトン	Krypton	`LIB.P8KR.LOAD`
キセノン	Xenon	`LIB.P8XE.LOAD`
n- 水素	n-Hydrogen	`LIB.P8H2.LOAD`
塩素	Chlorine	`LIB.P8CL2.LOAD`
窒素	Nitrogen	`LIB.P8N2.LOAD`
酸素	Oxygen	`LIB.P8O2.LOAD`
空気	Air	`LIB.P8AIR.LOAD`
二酸化炭素	Carbone Dioxide	`LIB.P8CO2.LOAD`
水 (IFC 1967-IPTS1968)	Water(IFC 1967-IPTS1968)	`LIB.P8H2O.LOAD`
水 (IFC 1967-ITS1990)	Water(IFC 1967-ITS1990)	`LIB.P8H2O90.LOAD`
水 (IAPS 1984)	Water(IAPS 1984)	`LIB.P8HGK.LOAD`
重水	Heavy Water	`LIB.P8D2O.LOAD`
アンモニア	Ammonia	`LIB.P8NH3.LOAD`
六ふっ化硫黄	Sulfur Hexafluoride	`LIB.P8SF6.LOAD`
メタン	Methane	`LIB.P8CH4.LOAD`
エチレン	Ethylene	`LIB.P8C2H4.LOAD`
エタン	Ethane	`LIB.P8C2H6.LOAD`
プロピレン	Propylene	`LIB.P8C3H6.LOAD`
プロパン	Propan	`LIB.P8C3H8.LOAD`
FC-C318(フロン C318)	FC-C318(RC318)	`LIB.P8RC318.LOAD`
CFC-12(フロン 12)	CFC-12(R12)	`LIB.P8R12.LOAD`
CFC-113(フロン 113)	CFC-113(R113)	`LIB.P8R113.LOAD`
CFC-114(フロン 114)	CFC-114(R114)	`LIB.P8R114.LOAD`
HCFC-22(フロン 22)	HCFC-22(R22)	`LIB.P8R22.LOAD`

HCFC-123(フロン 123)	HCFC-123(R123)	`LIB.P8R123.LOAD`
HFC-134a(フロン 134a)	HFC-134a(R134a)	`LIB.P8R134A.LOAD`
ハロン 1211(フロン 12B1)	Halon 1211(R12B1)	`LIB.P8R12B1.LOAD`
ハロン 1301(フロン 13B1)	Halon 1301(R13B1)	`LIB.P8R13B1.LOAD`
フロン 500	Refrigerant 500	`LIB.P8R500.LOAD`
フロン 502	Refrigerant 502	`LIB.P8R502.LOAD`
フロン 503	Refrigerant 503	`LIB.P8R503.LOAD`
湿り空気 (理想気体)	Moist Air (Ideal Gas)	`LIB.P8MAIG.LOAD`
湿り空気 (実在流体)	Moist Air (Real Fluid)	`LIB.P8MARF.LOAD`
純物質のシングルショット・プログラム	Single Shot PROGRAM for Pure Substance	`LIB.P8PURESS.FORT`
湿り空気 (理想気体) のシングルショット・プログラム	Single Shot PROGRAM for Moist Air as Mixture of Ideal Gas	`LIB.P8MAIGSS.FORT`
湿り空気 (実在流体) のシングルショット・プログラム	Single Shot PROGRAM for Moist Air as Real Fluid	`LIB.P8MARFSS.FORT`

【マニュアルについて】

PROPATH 第 8.1 版の利用の手引は，LaTeX のソースファイル，DVI ファイルで公開している（ASCII 日本語 TeX）．センター 2 階のオープン室にフロッピーディスク (MS-DOS format) が置いてあるので各自でコピーして結構である．ソースファイル，DVI ファイルの処理方法は，`README.DOC` を参照．ただし，フロッピーの LaTeX は図表の部分が取り込まれていません．必要な方は，ライブラリ室まで御連絡下さい．コピーを送付致します．

【参考文献】

- 「PROPATH 流体の熱物性値プログラム・パッケージ 第 8.1 版」プロパス・グループ, 1993.
- 「PROPATH 流体の熱物性値プログラム・パッケージ 第 8.1 版, 新版公開のお知らせと大型計算機利用者のための補遺」 プロパス・グループ, 1993.
- 「UXP/M for the Working Scientists」佐藤 周行, 1993.

16.4 UNICS

UNICS(Universal Crystallographic Computation Program System) は日本結晶学会が全国の大学，研究所で開発されたプログラムを統合整備して作成した結晶構造解析用ユニバーサル・プログラムシステム．九州大学版は，移植者による追加開発分を含む．

プログラム形式	コンプリート，サブルーチンプログラム
作成者	日本結晶学会
改訂者	河野 重昭（九州女学院短大）
作成年月日	1983 年
使用言語	Fortran
ソースの公表	'QS.LIBSC(Y3*****)'
使用 OS	MSP(M1800)

【使用例】

- プログラム FILEIN により，CRYSTAL DATA FILE を作成する．

```
 READY
ALLOC F(FT01F001) DA(XTAL.DATA) ↵
ALLOC F(FT02F001) DA(RD.DATA) ↵
FREE F(FT05F001) ↵
ALLOC F(FT05F001) DA(PO.DATA(FILEIN)) ↵
UNICS FILEIN ↵
```

【参考文献】

- 「UNICS II の使用法について」河野 重昭, 九州大学大型計算機センター広報 Vol.13, No.1, pp.39–50, 1980.
- 「UNICS III の使用法について」河野 重昭, 九州大学大型計算機センター広報 Vol.16, No.2, pp.113–154, 1983.


GOOD
Night