

SPIDER — ポータブルな画像処理

サブルーチン・パッケージ

田村秀行[†] 坂根茂幸[†] 富田文明[†] 横矢直和[†] 坂上勝彦[‡] 金子正秀[‡]

[†]電子技術総合研究所 [‡]東京大学生産技術研究所

1. はじめに

デジタル画像処理技術は、この10年位の間に急速に発展し、ハードウェアの進歩とともに様々なアルゴリズムが発表されてきた。しかしその多くは特定の応用分野に依存したもので、一般的な図形・画像情報を扱い得る技術体系を確立することが望まれている。このためには標準的な画像データを揃えて、その上で各種アルゴリズムを比較検討し、体系的整理を行なう必要がある。

こうした考えの下に、情報処理学会旧イメージプロセッシング研究連絡会では、標準化活動として、画像データ交換用の標準データフォーマットと画像入出力装置の較正法を起草した^[1]。次いで、基本的なアルゴリズムをプログラムの形で収集・蓄積・配布するための仕様を検討した^[2]。それには、移植性(*transportability*)のあるプログラム仕様でなければならぬことは勿論だが、現実にプログラムを収集したり、あるいは新たに作成する過程でこの仕様を修正して現実的なものにして行く必要がある。またライブラリの流通と新たなルーチンの登録を期待するには、初期セットとして主要なルーチンを含んだパッケージを配布するのが望ましい。

電総研では、大型プロジェクト「パターン情報処理システム」の一環として、文字認識・図形認識・物体認識の研究開発を過去8年間行なってきた。これら画像関連の研究は、基礎的な前処理・特徴抽出の段階では共通する部分が多く、個々の実験サブシステムで重複して作成したプログラムもかなり

ある。それらを整理・統合し1つのサブルーチン・ライブラリにするに際し、上記のIP研究連絡会の意向を反映したものとなるよう配慮した。こうして出来たのが、画像処理サブルーチン群と若干のサービス・プログラムから成るパッケージSPIDER (*Subroutine Package for Image Data Enhancement and Recognition*)である。

本稿はこのパッケージの公開にあたり、設計方針とプログラム仕様を解説したものである。

2. SPIDER開発の基本方針

2.1 移植性の達成についての考え方

各種画像処理アルゴリズムをライブラリの形で蓄えることは、従来から多くの画像処理システムで行なわれていた。実際、画像処理専用のソフトウェア・システムの具体例も多く、それらの比較評価もなされている^[3]。にもかかわらず、数値計算ライブラリや様々な統計処理パッケージのように暗箱的に用いられるソフトウェアは画像処理分野では流通していない。これは、画像処理ソフトウェアの構造が画像処理システムのハードウェア構成に大きく依存した形で設計されていることが多いからであると思われる。

画像処理システムの特徴は、通常の計算機システムには非標準である特殊入出力装置や、大容量の画像データの管理を含むことである。大型計算機システムのためのオペレーティング・システムをこのために改造することは容易ではなく、能率中心の独自の画像処理システムがミニコンなどを中心として構成されることが多い。(文献[2]の表1

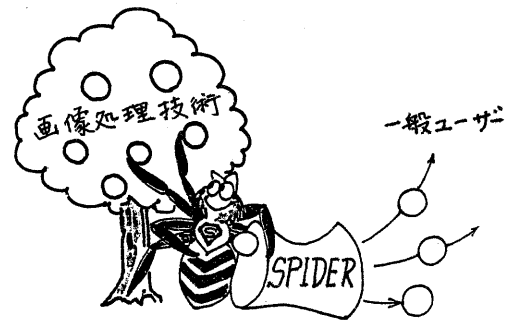
を考慮されたい)。

こうした画像処理ソフトウェアをポータブルなものにするには、単に通常のプログラミング言語の仕様と処理系の問題にとどまらず、画像ファイルの構成やタスク実行時の画像データ構造をも考慮に入れた対策が必要である。

完全な移植性を望むならば、画像処理操作手順を容易に記述でき、画像ファイルの管理がインプリシットに行なわれる画像処理専用共通言語とその処理系を開発すればよいであろう。勿論、この際、画像処理手続き概念の抽象化と構造化が予め図られていなければならない。コンピュータ・グラフィックスの分野では、基本概念の標準化を行なった上で、汎用的なパッケージを作りつつある^[4]。しかし、画像処理の分野では、まだここまでの標準化はなし得る段階ではないと思われる。

岳しくも機を同じくして、米国でもポータブルな画像処理ソフトウェア作りの計画が進められている^[5]。この計画では、まずソフトウェア工学の立場から一般的なポータブル・パッケージの在り方を検討し、その1例として画像処理の場合を考えるアプローチをとっている。具体的には、ファイル操作や直接I/OなどのOSとの接点をサブルーティン化し、これを標準化している。このポータブル・パッケージを利用するには、標準サブルーティンと各機種種のOS間のインターフェイスは各自が作成せねばならない。コンピュータ・ネットワークが発達していない我が国では、これは各ユーザーにとって大きな負担であろう。実際、米国でも1, 2の機種に試みただけで、画像処理ソフトウェアはこれからその上に作られて行くという段階である。

これに對して我々は、もっと実現し易い移植性達成策を採った。現時点での流通を考えて、あらゆる画像処理ル



カット(協力): 塩見佳久

ーチンをFORTRANのサブルーティンとし、移植を妨げ易い入出力を扱わないようにした。その他詳しい仕様は後述するが、サブルーティン仕様の制限を厳しくすることによって移植性を高めている。画像データの準備は各ユーザーの責任であるが、能率と画像の大きさを問わなければ、大抵のFORTRANコンパイラで一応の画像処理実験が可能である。

2.2 SPIDERの趣旨と目的

以上のように、プログラム仕様の美しさの点から移植性を考えず、現実にはアルゴリズムを収集し、パッケージが利用されることを第一義とした。このSPIDERがめざすものは、次の通りである。

- (1) デジタル画像処理の基礎的な手法をモジュール化して集め、ソフトウェアの財産として蓄積・交換することを目的とする。
- (2) 研究実験用のライブラリとしての充実をめざす。使用頻度の高い低レベルの処理(例、ヒストグラム変換、平滑化)、これまで同一主旨で少しずつ異なったアルゴリズムが数多く発表されているもの(例、エッジ検出、細線化)、やや複雑で各個人ではなかなか揃えられなかったもの(例、弛緩法、Huecke / オペレータ、

- Split & Merge法)などを積極的に集め、真に有効なものの選択やより高度なアルゴリズムの開発に役立てる。
- (3) 能率よりも汎用性、移植性を優先させる。非専門家がそのまゝ用いて実験できるような配慮する。
- (4) 能率を重視する個別システムへの完全な移植は望めないので、修正して登録し易いよう単純明確なプログラムを作る。

3. パッケージの構成

画像処理アルゴリズムとして何を入れるかについては、国内の約40研究グループに対してアンケートを実施し、画像処理実験体制の現状、提供してもらえるプログラム、登録して欲しいアルゴリズム等を調査した。

現在、国内で比較的多くのルーチンを有していて、他の計算センターへの移植実績を持つのは名古屋大学のSLIP (Subroutine Library for Image Processing)^[6]である。SPIDERの開発計画当初から、このSLIPを念頭において仕様を検討してきた。アルゴリズムの収集については、数多くのグループからプログラムのソースリストが公開されたが、名大からはSLIPのほぼ全てのリストの提供の上に、筆者らの希望のルーチンを整理・統合してSPIDER仕様に変換する作業の協力を得た。

さて、第一期開発としてSPIDERに含まれる画像処理ルーチンは附録に示したアルゴリズムとその下位ルーチンから成るFORTRANのサブプログラム群(サブルーチンと関数副プログラム)であり、このうちマニュアルに記載されていてユーザーから利用できるものは348個である。ただし、類似のものも相当数ある。

アルゴリズムによっては、大きな画像データの受け渡しの都合上、いくつかのサブルーチンに分けたもの、多数

の下位ルーチンを含んでいるが見かけ上1つのサブルーチンの引用で済むものの寄、形態は様々である。下位ルーチンについては、画像処理的意味があり、ユーザーが何らかの形で利用する可能性のあるものは極力マニュアルに記載した。

内容的には、直交変換以外の符号化技術と投影からの画像の再構成を除く画像処理のほぼ全般に及んでいる。SLIPから変換されたルーチンは63個であり、両ライブラリを利用したときの混乱をさけるためサブルーチン名はすべて変更してある。

これらのアルゴリズムの最終的な選択は、製作に当たった筆者らのグループの独断によるもので、必ずしも登録されていない手法があるという訳ではない。各分野ごとにサーベイを行ない、種々の方法を整理したが、時間的・予算的な制限からコーディングできなかったものも多い。また各分野の軽重も同じ理由によるものである。

FORTRANの組み込み関数以外の数値計算ルーチン(逆行列計算、連立方程式解法、等)については、何らかのライブラリが利用できるものとして、SPIDERには入れなかった。筆者の予もとで利用できるサブルーチンが呼ばれているので、勿論、この部分は各ユーザーが入れ換える必要がある。リスト処理については用意されていないことが多いので、ACMのSIGPLANから公開されているサブルーチン群L_{PN}^[7]をSPIDERに入れ、マニュアルにも附録として解説を記載した。(上記のサブプログラム数には、これは含まれていない)。

この他に、後述のMT形式Aに対して動作するサービス・プログラムが3個、主プログラムの形で用意されている。機能は次の通りである。

(1) SPI—指定の、もしくは全部の

(圖1 參照)

図 1. SPIDER ソースプログラム配布テープの形式 (9トラックに限る)

SPIDERに登録すべきサブルーチンの満足すべき条件を以下に述べる。ここで「なるべく」と記した項目は、第1期の開発令ではほとんど実施されている。この仕様は、パッケージの製作中に検討を重ねて決定したものである。

る。

4.1 FORTRAN 言語レベル

現在では、大型機からミニコンまで、ほとんどのFORTRANコンパイラはJIS 7000レベルを大きく超えています。SPIDERとしては、国産計算機のMシリーズ、ACOSシリーズ、COSMOシリーズの中位機種程度のレベルを認めることにした。JIS 7000レベルを超える機能、JISでは規定していない項目について、全機種についての共通集合部分を軒し、FORTRAN 77^[8]で縮小される機能はJIS 7000より厳しくする方針をとった。主に注意すべき点は次の通りである。

- ザーラの型と長さ :
 - 実数 — 倍精度 (8 バイト) まで
 - 複素数 — 単精度 () まで
- 文字型定数 ---- "x₁x₂...x_n" を許す。
- 配列の最大次元数 ---- 7
- 継続行の最大行数 ---- 19
- D₀ の入れ子の深さ ---- 20 以下
- 使用してよい文字 :
 - JIS 7000 の 47 文字の他に、
 - 英字 ---- 羊を追加
 - 特殊文字 ---- 〰 と 〴 を追加
- 混合演算 ---- 許す。
 - (なるべく避けた方がよい)
- 計算形 G₀ T₀ 文 ----
 - 整数式の値が行き先の範囲を越えることは禁止する。
- D₀ 文の最小繰り返し ----
 - 初期値が終値を越えた時の実行を故意に利用する用法は使わない。
 - (上記 2 つに対しては FORTRAN 77 で対応されているが SPIDER では避ける)。
- D₀ の拡張範囲 ---- 禁止する。
 - (これは FORTRAN 77 に従う)。
- D₀ の制御パラメータ ----
 - JIS 7000 と同じく算術式を許す。

ない。

・配列の添字----

JIS 7000と同じ制限つきで整数型の式とする。

(上記2項目については、HITAC-Mシリーズの中・下位機種のみがJIS 7000並みになっているためである。他のコンパイラに慣れたユーザーにはかなり厳しい条件なので、やがてSPIDERもFORTRAN 77並みに任意の整数型算術式を認めてよいかもしれない)。

この他にもJIS 7000を超える機能があるが、ほとんど入出力並びや宣言文等に関するものであり、むしろ以下のSPIDER仕様では禁止している。

4.2 サブルーチン設計方針

SLIPの様子は文献[6]に詳細に報告されている。ここでは紙数の制限からSLIP仕様との主な相違点のみを列挙する。

(1) ページ内サブプログラム間での引用(呼び出し)を許す。サブルーチンの階層化を許し、下位ルーチンもなるべく独立したプログラムとして利用できるように汎用性をもたせる。このために下位ルーチンでのみ利用するパラメータが上位ルーチンの引数となるのは止むを得ない。

(2) COMMONによるプログラム間の情報伝達は原則として認めないが、例外としてリスト処理ルーチンとの

間では用いている。

(3) 整数型、実数型両タイプの準備は当面、特に必要な場合に限る。画面内の一部を処理するウィンド処理や参照プレーンによるマスク処理(図2参照)についても別ルーチンを作るが、これも特に必要なもののみにしどめる。

(4) 画像やヒストグラム等の打ち出しルーチン等の例外を除いて、サブルーチン内部では入出力文は一切認めない(エラーメッセージも禁止)。

(5) エラー処理を行なう場合でも、サブルーチン内部では停止させず、必ずエラーコードを引数として返す。
0: 正常, 正整数: nonfatal error, 負整数: fatal error とする。

(6) 入力画像と出力画像の仮引数は別にし、入力画像を保存できるようにする。(可能なものについては実引数を同じにして差しつかえない)。

(7) 主記憶に一度に入らない大画像に対して、小領域を順次分割処理する方式^[9]で不都合な演算について対策を考える。

a) 反復演算については、1反復ごとに制御できるようにする。

b) 累積量を訂数する場合(ヒストグラム等)は、この量を引数に累積して行く形のルーチンを作る。

c) fatal error コードが受け渡された時、これをまず判別し何も処理

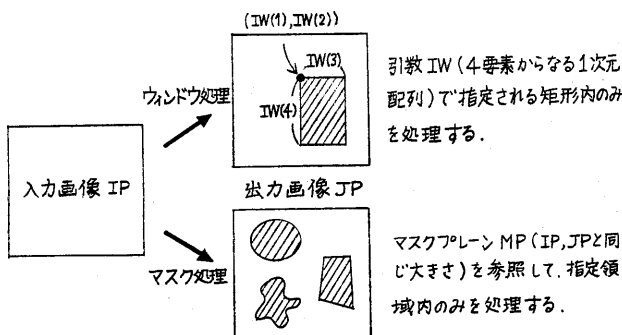


図2 ウィンドウ処理とマスク処理

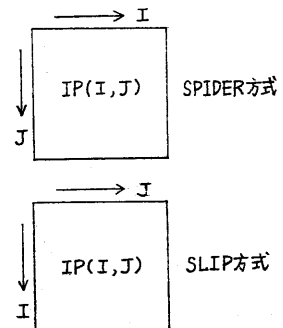


図3 二次元配列と画像の対応

せずに return する。

- (8) サブルーチン名の文字のうち機能を表す英字は4字以内(SLIPでは5字も可)とする。残りの文字は同種内での識別用数字(通し番号でなくても可)と次の特定の型指定に用いる。

I: 整数型(特にRと区別するときのみ)

R: 実数型, W: ウィンド処理,

M: マスク処理, A: 累積型。

- (9) 画像用2次元配列の第1添字と第2添字の画像への対応が、SLIPとSPIDERでは図3のように異なる。SLIP式は行列要素の表現と同じだが、SPIDER式は通常の水平ラスタースキャンでの点系列とFORTRANでの語順が一致する利点がある。両方式を混用すると不都合なことが起こり得るので、いくつかのルーチンを取り出して使ったり、新しいルーチンを登録する際には注意を要する。

4.3 コーディング上のとり決め

これもSLIPとの相違点のみ述べる。

- (1) 4.1でも述べたようにデータの制御パラメータや添字式の制限はSLIPよりかなり厳しい。
- (2) 引数の型はすべて暗黙の型宣言に随う。また、仮引数はなるべく次の頭文字を冠して使う。

I, A ---- 入力専用。

J, B ---- 出力専用。

K, L, C, D ---- 入出力兼用。

M, N, E, F ---- 定数用。

- (3) よく使う仮引数はなるべく次のようなものを用いる。

・画像配列 ---- P をつける。

IP1, JP3 等(ごく一部の例外を除いてKPはあり得ない)。

・画像の大きさ ---- ISX, ISY。

・作業領域 ---- IWORK。

・しきい値 ---- ITHD, FTHD 等。

・ウィンド ---- IW。

・エラーコード ---- JERR, KERR。

- (4) 原文献中の記号にならべく近い変数名を用いる。

- (5) 10進数以外の定数(8進数、16進数、文字等)はDATA文で指定し、実行文には書かない。

- (6) FORMAT文以外の文には、なるべく文番号をつけないようにし、IFやGOTOの行き先はCONTINUEで受けるようにする。特にD文の始めと終わりには文番号を断さない。終りは必ずCONTINUEとする。

- (7) D文の入れ子やIFによる分岐先はなるべくindentationを行なって見やすくする。

- (8) D文の順序は、第一添字から順に内側の入れ子にして回す。

- (9) 計算機の語長に依存する演算は行わない。従って整数型と実数型の語長を同じと仮定した処理を行ってはならない。ただし、最大(最小)を求める際の初期値などで定数として大きな(小さな)数を与えることにより語長に関係することがあるが、これはマニュアルに特記事項として注意を与える。

- (10) コメント文は、いくつかのグループに分け、テキストエディタやプロダラムで検索正分して出力できるように文頭の文字を規定する(附録2参照)。

4.4 マニュアル記入項目

SPIDERのマニュアルは、次の3部に大別されている。

- ・第1部 --- SPIDERの趣旨と方針(本稿を拡張したもの)。
- ・第2部 --- 画像処理分野別の各サブルーチンの位置づけと用途の解説。
- ・第3部 --- 各サブルーチンの説明書。この第3部に収納すべき各説明書には次の項目を記入する。

1. 目的・機能(なるべく簡潔に)。

2. Calling Sequence.
3. 各引数の説明。
4. 引数についての条件。
5. 使用上の注意。
6. アルゴリズム。
7. エラー処理。
8. 特記事項(ここには各機種ごとに若干修正して使う場合の注意事項を述べる)。
9. 実験例。

5. 今後の運営と課題

SPIDERは現コンピュータビジョン研究連絡会に貸与され、学術目的であれば誰にでも公開される。東大生研多次元画像情報処理センター、筑波大学学術情報処理センター等を通じて配布される予定である。

現在、予定したプログラムはすべて一応の虫取りを終えているが、厳密な意味では言語仕様と合致しているか確かめられていない。製作に当たったFORTRANのレベルが高いためである。一般公開に先立って、いくつかのコンピュータに通してみる必要がある。

マニュアルは約600ページ(予定)の大部であり、印刷校正に手間どっている。仕様決定やプログラミングよりもむしろキュメンテーションの方が大変、というのがこうしたパッケージ作りの経験から得られた教訓である。

SPIDERの製作期間は、計画より現段階まで1年足らずである。SLIPのように長い年月をかけて蓄積されたものでないので、かなりの初期不良があると思われる。特に、各サブルーチン内のコメントやマニュアルでのアルゴリズム解説は十分整理されたものではない。エラー処理や実験例も、もっと充実させるべきであろう。

こうした点は、多くのユーザーの使用を通して順次改良されて行くことが望まれる。漏れている重要なルーチン

○
ユーザーからの追加登録
熱烈歓迎



や新規開発アルゴリズムの追加、よりスマートなプログラムとの入替が頻繁に行なわれることが、このパッケージ本来の目的である。新しい追加ルーチンについては、命名の問題があるので、統一した改訂が何度か必要となるだろう。

6. むすび

画像処理アルゴリズムの蓄積と流通をめざしてサブルーチンパッケージSPIDERを開発した。各ルーチンは画像データ管理とは独立にし、仕様を厳しくすることで移植性を高めた。

製作に従事した者は、各自の研究の側面ボランティアとしてこの計画に参加したので、出来ばえには不満足な点も多い。しかし、第1期セットとしてはかなりの数のアルゴリズムを収録できたと自負している。

プログラムの収集と新規製作については、ここに書き切れない程多くの方々の御協力を得ました。特に、東大生研の尾上教授、高木教授からは計画全般にわたる御指導を、石大の島崎助教授からはSLIPの一部を交換して頂く御好意と御協力を得ました。また、西野PIPSプロジェクト・リーグによる御理解と御支援、JBAの手塚雅久氏の超人的なプログラミング能力によって始めて本計画が遂行できました。

参考文献

- [1] 高木幹雄, "画像の標準フォーマットと入出力装置の校正法", イメージプロセッシング・シンポジウム予集, pp. 5-36 (昭54).
- [2] 田村秀行, "画像処理用アルゴリズムソフトウェア", 同上, pp. 37-49 (昭54).
- [3] L. Borriello et al., "An Image Processing Support Software Study", vol. 0-5, CSATA Repo., Bari, Italy (April 1978).
- [4] Special issue on "Graphics Standards", ACM Comput. Surveys, vol. 10, no. 4, pp. 363-502 (Dec. 1978).
- [5] R. G. Hamlet and A. Rosenfeld, "Transportable image-processing software", Proc. NCC, vol. 48, AFIPS Press, pp. 267-272 (1979).
- [6] 島崎他, "画像処理サブルーティンライブラリSLIPについて", 信学技報 PRL 78-69 (昭54-01).
- [7] D. W. Bray, "Dynamic storage routines for Fortran programs", SIGPLAN (Sept. 1974).
- [8] 西村忠彦, "FORTRAN 77の特徴", 情報処理, vol. 20, no. 5, pp. 432-437 (昭54-05).
- [9] 田村 森, "画像処理基本プログラムの互換性と大画像の分割処理方式", 昭53信学総大会, no. 1010.

付録1. SPIDER第1期開発主簿ルーチン

直交変換(1次元及び2次元, 順変換及び逆変換)

FFT(基数 $\dots 2$, 混合位相), WHT, Haar変換, Slant変換, 離散的Cosine変換(ハタフライ型式, FFT型式)
 関連ルーチン... フーリエスペクトル計算, WHTスペクトル計算(2種)
 たみこみ(直接法, FFT法), 相関(直接法, FFT法), パワースペクトル面での分布計算(半径方向, 角度方向)と窓処理(矩形, カウス, ハミング, ハニング), 周波数領域での微分フィルタ(差分, gradient, Laplacian), その他

位置合わせ

相関法(粗サーチ, 精サーチ), SSD(固定又は傾斜しきい値, しきい値の指定又は自動決定, 粗サーチは精サーチ), アフィン変換(4点線形補間, 9点2次補間, 補間なし)とパラメータ決定(回転角, 参照点), 一般2次変換(アフィン変換と同じ3種)

強調と平滑化

ヒストグラム変換(平坦化, 双曲線化, 一般), 反復による強調とノイズ除去(2種), ヒステリシス平滑化(標準型, 対称型), E-フィルタ, 高速メタエアンフィルタ, エッジ保存平滑化

復元

逆フィルタ(4種), ウィナフィルタ, 制限つき最小2乗フィルタ, SVDによる擬似逆変換(3種)

エッジと線の検出

差分型(差分, gradient, Laplacian, Roberts, Sobel, Prewitt), テンプレートマッチング型(Prewitt, Kirsch, Robinson, Frei & Chen), Hueckelオペレータ(Hueckel, Meró & Vassy), 反復型(Kasvand), Hough変換(Duda & Hart, Perkins & Binford), ヒュリスティック探索法, その他

弛緩法

基本ルーチン群, 線の強調(係数の指定又は自動決定2種), エッジの強調, ノイズ除去

テクスチャー解析

cooccurrence行列, 差分統計量, 局所的極値, ランレングス, 自己回帰モデル, 自己相関, フーリエ記述, テクスチャーエッジ検出, テクスチャーエッジ保存平滑化

領域分割

KS検定法, ヒュリスティック法, 反復合併法, 反復しきい値処理法, Split and Merge法

幾何学的特徴の処理

ラベル付け, ラベルの通し番号化, 取捨選択, 境界線抽出, 膨張と収縮, 細線化(4種), 縮退化(5種), WPM法, 距離変換とスケルトン, 境界線の記述(チェイン符号, 傾き, 曲率, フーリエ記述), 形状的特徴(始点, 重心, 外接長方形, 面積, 周囲長, サイズ, 伸長度, エメント, フーリエ記述)

基本的処理

線形フィルタ, 正規化, ヒストグラム作成, 断面・投影の計算, しきい値処理(各種)としきい値選択(4種), 基本統計量, 型変換, 画像定数間・画像間演算, 量子化, 代入, データ転送

その他

図形(ラスターパターン)の発生, LPへの画像出力, サービスプログラム(3種)

付録2. コメント文についての指定

CS (Sequence)	コーリング・シーケンス
CP (Purpose)	目的・機能
CK (Key words)	キーワード (区切りは,)
CD (Data)	作成年月日と作成者 (updateおときの指標)
CA (Arguments)	引数の説明
CV (Variables)	その他の変数の説明
CE (Error)	エラー・コードの説明
CN (Note)	引数の条件や使用上の注意等
CM (Maintenance)	

修正・保守用の特記事項の説明

以上は SUBROUTINE 文の後, 宣言文の前にまとめて入れる。
 ただし, CMは修正すべき文々の直前に入れてよい。

以下のコメントはプログラム途中に随時入れてよい。

CF (Flow)	アルゴリズムの大略のフローをなされるようなコメント
C*	その他の雑コメント
C	プログラムを見やすくするための区切り
_____	オールブラック
_____	この他に
C - - - - -	などとも好みによって区切りとして便法よい