

図形文書処理システム新版 (DOG R A C E)

情報処理センター 車古 正樹

図形文書処理システムDOG R A C Eは、従来のFCP, TXTFCP, DOGの順に改良した最新版である。ここでは主としてDOG R A C Eの概要について紹介する。

1. DOG R A C Eの起動方法

DOG R A C Eの起動は次の順に行う。

① 起動コマンドの入力

```
LOGON TSS AB9999 S(3072) TIME(10)
```

DOG R A C Eはメニュー方式によるため、主記憶容量が2 MB以上必要とするから、必ず Size オペラントで3072程度を指定する。

② 金沢大学プログラム開発支援機能システム (K P F D) 起動コマンドの入力

```
KPFD
```

DOG R A C EをK P F Dの一部としてシステムに組み込んだため、まずK P F Dを起動する。

③ K P F D初期選択画面でDOG R A C Eの選択

```
オプション ==> D
```

図1の選択画面が表示される。この画面でT S S処理か、バッチ処理かを選択する。

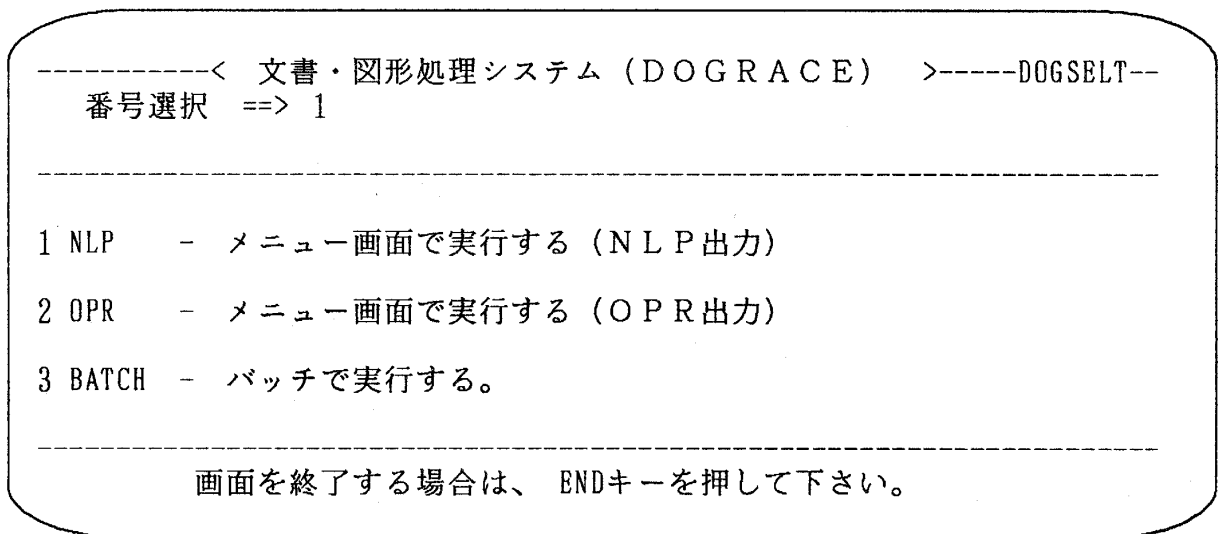


図1 DOG R A C Eのモード選択

④ TSSモードの選択

番号選択 ==> 1

図2のDOGRACE処理選択画面が表示される。図2の指定方法や実行方法については次節を参照するとよい。なお、番号選択欄に2を入力するとOPR出力用画面となる。

上記の②～④の操作の代わりに図2を表示することも可能であり、その場合はREADY状態で次のように入力する。

DOGRACE M

```
***** 文章・図形処理システム (DOGRACE) *****
-----処理選択画面-----
項番入力 ==>

項番  項目名      機      能
I   入力      プログラム・データセット ==> TEST.DATA (PAGEERR)
A   追加入力   上記プログラムの後ろに追加する
J   修正      入力用プログラム・データセットの編集を行う
U   修正1     修正用データセット名 ==> TEST.DATA
V   修正2     修正用データセット名 ==> TEST.DATA (BOX01)
G   実行      プログラムの実行を開始する      ↓実行モード ==> C
R   実行      入力と実行を行う      A:出力無し B:出力あり C:出力あり
D   削除      出力データセット名 ==> DOGRACE.OUTLIST
                   直接出力の場合 ==> K      (K 一階 N 二階)
O   出力      出力結果の入ったデータセットの印刷
                   同一ページの高速印刷部数 ==> 1      部数 ==> 1
E   終了      DOGRACEを終了する
                   出力データセットの削除 ==> YES      (YESまたはNO)
                   用紙の最大サイズ = 108文字×84行
X   TSS      TSSコマンドの実行を行う
                   TSSコマンド ==> LISTD TEST.DATA M
L   エラー   実行時の誤りの参照を行う
```

図2 DOGRACE処理選択画面

2. DOGRACE処理選択画面の操作

DOGRACEのコマンドの実行方法について以下に説明する。

① ユーザプログラムのローディング (入力)

プログラム・データセット欄に入力プログラムを指定し、項番入力欄にIと入力する。この後プログラムが実行可能となり、実行は項番入力欄にGと入力すればよい。なお、プログラム・データセット欄のプログラムのメンバ名のみを変更し入力を行う場合は、プログラム・データセット欄のメンバ名を変更せずに項番入力欄にI (メンバ名) と入力してもよい。同様に、プログラム・データセット欄のプログラム全体を変更し、入力を行う場合は、項番入力欄にI データセット名

(メンバ名) と入力すればよい。

② 2個以上のプログラムを同時に実行する場合 (追加入力)

1個目のプログラムをIで入力し、2個目以降のプログラムはA (メンバ名) で追加入力し、実行 (G) する。

③ プログラム・データセットの修正

項番入力欄にJと入力すると、プログラム・データセット欄の編集 (EDIT) 画面となる。後は、PFDのEDIT操作の要領で編集する。

④ プログラム・データセット以外のデータセットの修正

修正用データセット名欄が2個準備されている。修正1のデータセットの編集はUで、修正2のデータセットの編集はVで行う。

⑤ プログラムの実行

プログラムを実行する場合は、実行モード (A, B, C) と出力データセット名または直接出力の場合の出力クラス (K, N) を指定し、項番入力欄にGと入力する。NかKが指定されたならば出力図は直接日本語ラインプリンタ (NLP) に出力される。NかKを指定した場合は、印刷部数欄を必ず指定しなければならない。なお、直接出力欄にblankが指定されたならば出力データセットに印刷出力が保存される。実行モード欄でAを指定した場合は、出力図を掃き出さずプログラムのチェックと実際の図のサイズの表示を行う。Bの場合は出力図とサイズの表示、Cの場合は出力図のみとなる。なお、エラー等がある場合は、実行時に表示すると共に終了後エラーメッセージの編集画面となる。従って、エラーメッセージを確認し、行コマンドDDでエラーメッセージを消去し、EDIT画面を終了する。

⑥ 出力データセットの図形のNLP出力

実行時に直接出力せずにデータセットに保存した場合の図形の出力は、項番入力欄にOと入力する。この場合は直接出力欄にNかKを指定し、印刷部数欄に部数を指定する。なお、実行毎に以前の図に出力図が追加されるので、出力後不要になった図は項番入力欄にDを指定し、消去する。

⑦ 入力と実行

①と⑤の操作を同時に行う場合は、項番入力欄にRと入力する。

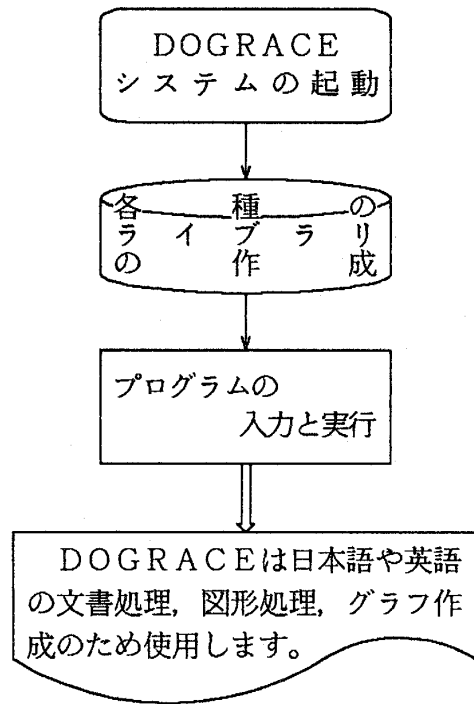
⑧ TSSコマンドの実行

図2の画面を表示中にTSSコマンドを実行したい場合は、TSSコマンド欄にコマンドを指定し、項番入力欄にXと入力する。

3. DOGRACE 概要

DOGRACEで描いた実例を図3に示す。図3を作成するプログラムは図4に示されるものである。

おおまかな文法については、このプログラムに添って説明されている。なお、プログラム中の「*」以後は注釈であり文として意味を持たない。また、文の最後の文字または注釈がある場合の注釈の直前の文字が-（マイナス）で次行の文字の先頭が*の場合は、文の**継続**を意味する。なお、これ以後に記載した図中で必要に応じ、その作画プログラムの入ったデータセット名（DSN=データセット）とメンバ名（MEM=メンバ名）を記載するので、プログラムを作成する場合の参考にとよい。



DSN=SYS9.DOGRACE.TEXT
MEM=EXSOC01

図3 実例1

```

INIT 108 84 M1 * 論理座標と物理座標の定義
    * ↑ M1は論理座標の単位が8分の1インチであることを表す
A:TELE 8.5/2.5 5:2 'DOGRACE': 'システムの起動' >> * 絶対座標
SET XFCT=5, YFCT=2 * 箱のサイズの設定 (デフォルト)
B:DISK '各種の'='ライブラリ'='の作成'=> * 座標やサイズの省略
C:PROC ¥+D, YFCT*3.0 - * 相対座標式
    * 'プログラムの', '入力と実行' < * 式による座標指定
SET DINT=1.5, DSTP=1, DMOD=0.8 * 文章の行間, 文字間などの指定
* 次の¥BOX名は文書BOXとなる。-は縦のサイズを実際の行数とする。
D:-¥DATA ¥C, D+D, YFCT*1.2 >D 8:20 * 方向指示子のある座標式
@DN=0.5@@PR=1.5, 0.5@DOGRACEは日本語や英語の文書処理,
図形処理, グラフ作成のため使用します。@NL@
¥ENDDATA * 文章データの終了
E:LP ¥D, C ¥D, F * 箱定数を用いた指定
ARROW ¥C, D, ¥D, U * 太矢印の作画
TEXT ¥E, D+D, 2 HGT=0.8 FORM=C - * 文字列の作画
    * 'DSN=SYS9, DOGRACE, TEXT'
TEXT ¥E, D+D, 3 HGT=0.8 FORM=C 'MEM=EXSOC01'
OUTPUT * 図形作画終了
END * プログラムの終了
  
```

DSN=SYS9.DOGRACE.TEXT MEM=EXPRTO1

図4 図3の作画プログラム

3. 1 プログラムの構成

プログラムはINIT文で始まり、プログラムの終了は必ずEND文で終る。また、図形出力の終了はOUTPUT文である。なお、図形の出力途中でページ換えはNEWPAGE文で行う。

◎INIT文

INIT文は図を描くための論理座標と物理座標を定義する。他に出力方向についても定義ができる。INIT文の実例を以下に示す。

```
INIT 108 84 M1
```

は図5に示すように、X軸とY軸の論理座標が108,84となりM1は倍率が8分の1インチを表し、実際の物理座標のサイズは各々13.5inchと10.5inchとなる。

mm単位で論理座標を指定するには次のように書く。

```
INIT 340 250 34 25
```

最初の340,250は論理座標であり、次の34,25はそれぞれの大きさを34cm,25cm(これはNLPの最大値である)として作画することを表す。

また、縦長として作画する場合は次のように指定する。

```
INIT 250 340 25 34 IANG=2
```

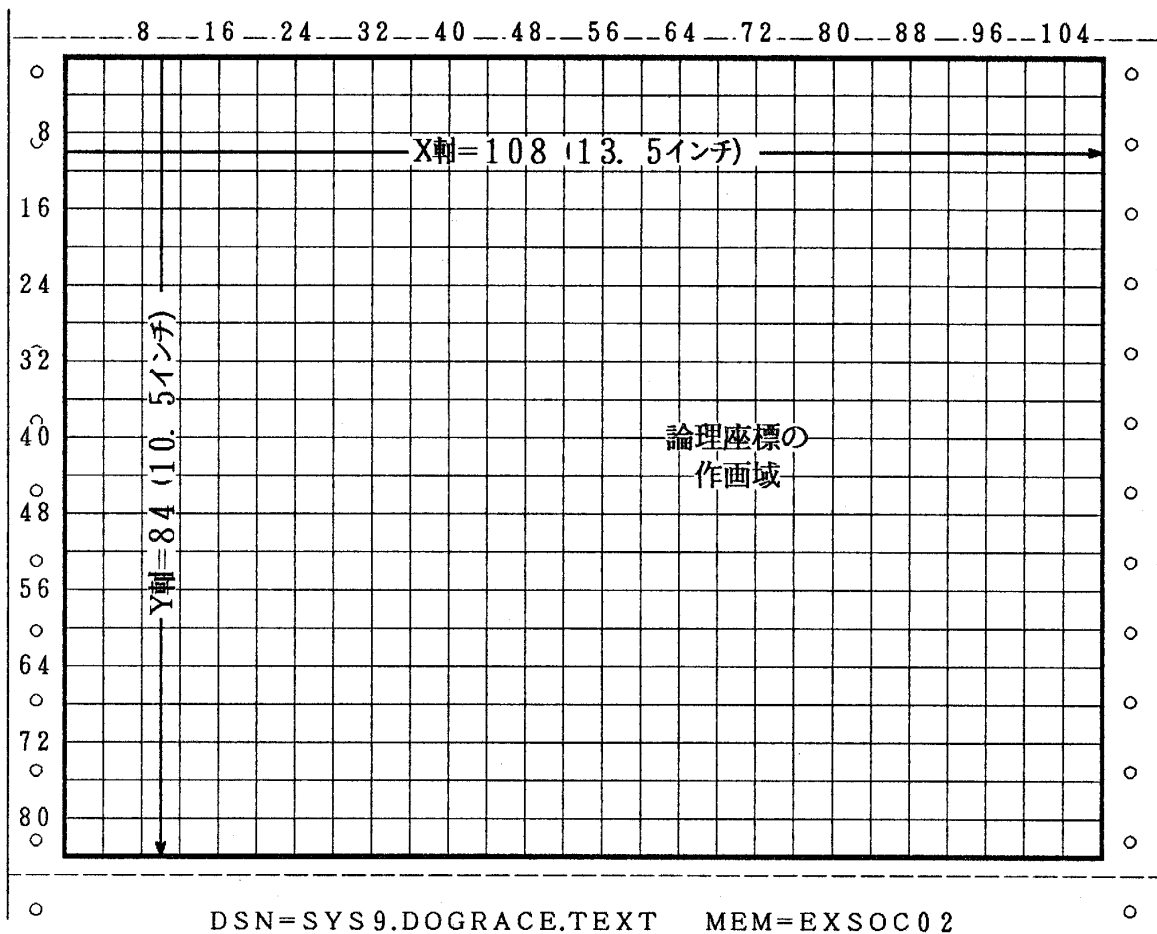


図5 日本語ラインプリンタ用紙上の論理座標とサイズ

IANG=2 は用紙を反時計方向に90度回転した座標系を示し、1~4までの整数のみ指定できる。任意の論理座標を指定するには次のように書く。

```
INIT -10:10 0:10 10 10
```

上記のINIT文はX軸に関する論理座標を-10~10、Y軸の論理座標を0~10とし、各々の物理座標のサイズを10cm×10cmとすることを表す。

3.2 BOX文の指定方法

DOGRACEではBOX(箱)文により、図を作成する。BOX文には標準BOX、ユーザBOX、線画BOX、文書BOXなどがある。標準BOX文について以下の実例で説明する。

```
A:TELE 8/2.5 5:2 'DOGRACE': 'システムの起動'>>
```

この文は図6に示すように、論理座標上の点(8, 2.5)を中心として、中心から横巾5、縦巾2の枠を描き、その枠中に' (引用符)で囲まれた文字列を作画する文である。この文の最初のA:はAというラベル名であり、後で説明するBOXに関する定数や飛び先ラベルとして用いる。次のTELEはDOGRACEで決められた標準BOXの1つであり、その種類については図7に示す。なお、DUMMY文は枠を書かない標準BOX文である。

次の'DOGRACE': 'システムの起動'>は'で囲まれた文字列をBOX内に出力する。出力する場合にその文字列が左揃えか、中央揃えかなど

を'文字列'の後に記号で指定する。この記号を字間出力制御子という。この場合は2行にわたる出力があり、行間についても制御が可能である。行間出力制御子は、出力文字列の前にFORM=行間出力制御子のように指定する。この例では最初の文字列は中央揃え(:)となり、2行目の文字列は適当な字間(横巾/字数-1)を開けて、中央揃え(>)となる。BOX内に縦書きに文字列を出力する場合は'の代わりに| (縦棒)で囲む。その他、反時計方向に90°傾いた字をBOX内に出力するには、システム変数KANGに2を指定(指定方法は後述のSET文かBOX文のオペランドで行う。)する。これらの組み合わせについて図8に示した。なお、最後の記号>は次に表れるBOXと矢印で結ぶ指示であり、BOX結合式という。単に線で結ぶときは_ (アンダーライン)を指定する。

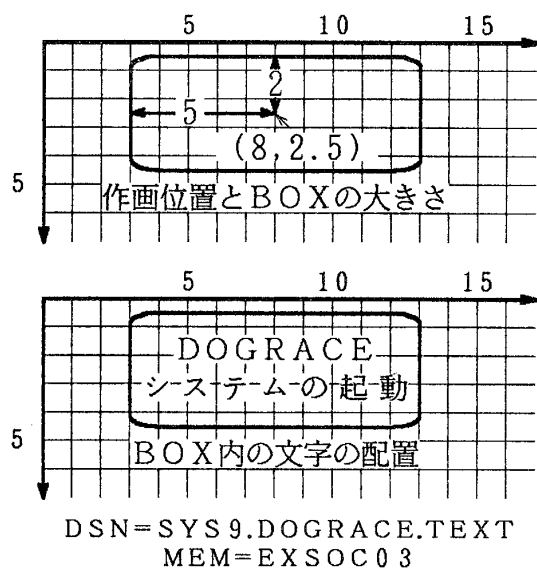


図6 BOX文による作画

3.3 BOX文の位置とサイズ

BOX文では位置やサイズを直接指定する方法

```
PROC 10/10 5:4
```

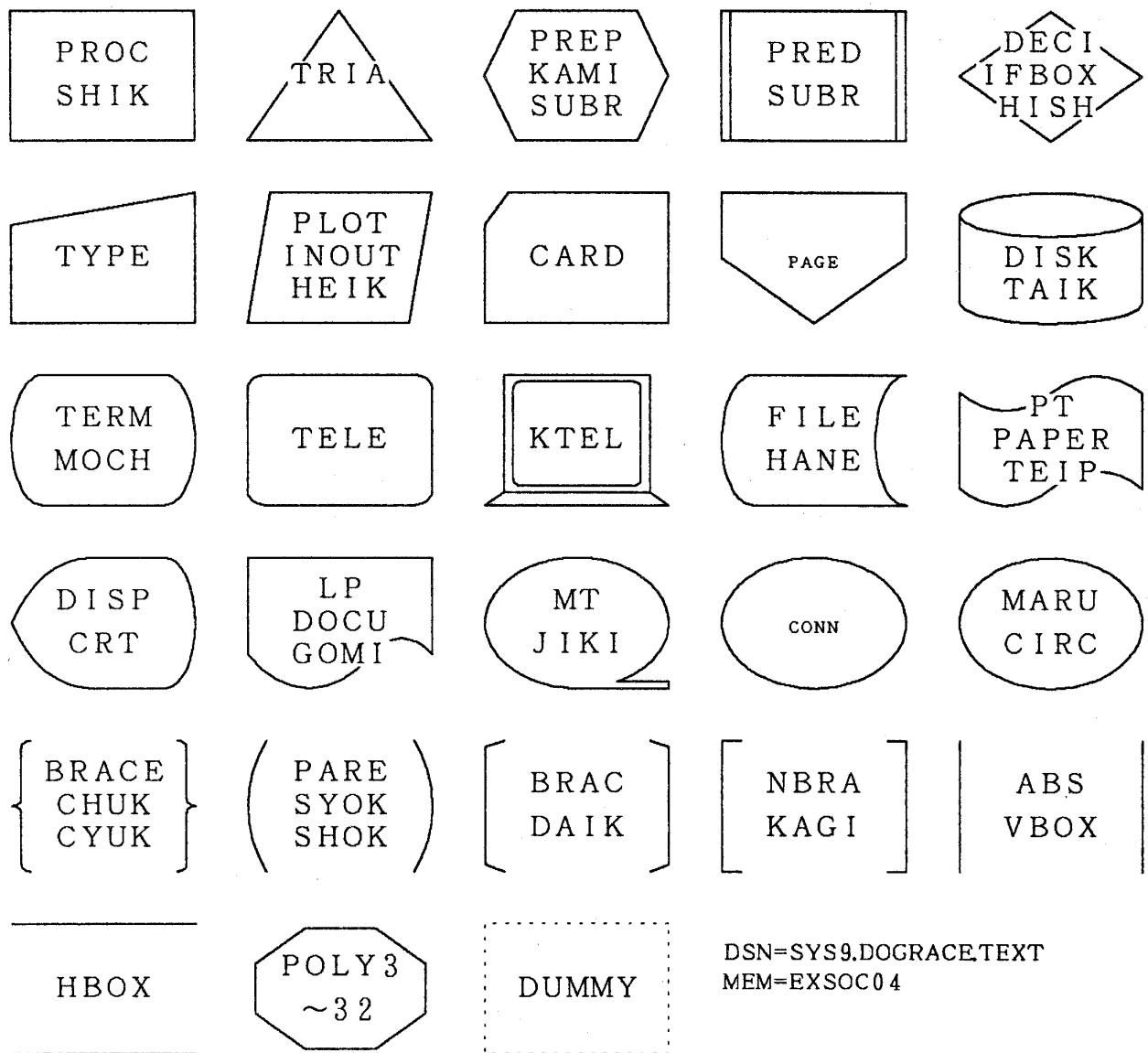


図7 標準BOXの種類

がある。また、サイズについて同じ大きさのものを何度も出力する場合は、次のような SET 文でサイズを指定し、BOX文ではサイズを省略してよい。

```
SET XFCT=5, YFCT=4
```

```
PROC 10/10
```

すなわち、XFCT (Xサイズ: BOXの横方向の半分の長さ), YFCT (Yサイズ: BOXの縦方向の半分の長さ) はシステム変数でBOX文のサイズ指定が省略された場合の横巾と縦巾のデフォルト値となる。また、図4の5行目のように

```
B:DISK ' 各種の '=' ライブラリ '=' の作成' =>
```

と、位置を省略した場合は、システム変数 HINT, VINT を使用し次のように計算される。

```
今回のX座標=直前のX軸の中心座標+ (直前のXサイズ+今回のXサイズ) ×HINT
```

```
今回のY座標=直前のY軸の中心座標+ (直前のYサイズ+今回のYサイズ) ×VINT
```

KANG=1
の場合

		字	間	出	力	制	御	子
		,	:	>	=	<		
子 御 制 力 出 間 行	Fか、	行間字間 出力制御	行間 字間 出力制御	行 間 字 間 出 力 制 御	行 間 字 間 出 力 制 御	行 間 字 間 出 力 制 御	行 間 字 間 出 力 制 御	行間字間 出力制御
	Cか：	行間字間 出力制御	行間 字間 出力制御	行 間 字 間 出 力 制 御	行 間 字 間 出 力 制 御	行 間 字 間 出 力 制 御	行 間 字 間 出 力 制 御	行間字間 出力制御
	Aか>	行間字間 出力制御	行間 字間 出力制御	行 間 字 間 出 力 制 御	行 間 字 間 出 力 制 御	行 間 字 間 出 力 制 御	行 間 字 間 出 力 制 御	行間字間 出力制御
	Bか=	行間字間 出力制御	行間 字間 出力制御	行 間 字 間 出 力 制 御	行 間 字 間 出 力 制 御	行 間 字 間 出 力 制 御	行 間 字 間 出 力 制 御	行間字間 出力制御
	Lか<	行間字間 出力制御	行間 字間 出力制御	行 間 字 間 出 力 制 御	行 間 字 間 出 力 制 御	行 間 字 間 出 力 制 御	行 間 字 間 出 力 制 御	行間字間 出力制御

KANG=2
の場合

		字	間	出	力	制	御	子
		,	:	>	=	<		
子 御 制 力 出 間 行	Fか、	行間字間 出力制御	行間 字間 出力制御	行 間 字 間 出 力 制 御	行 間 字 間 出 力 制 御	行 間 字 間 出 力 制 御	行 間 字 間 出 力 制 御	行間字間 出力制御
	Cか：	行間字間 出力制御	行間 字間 出力制御	行 間 字 間 出 力 制 御	行 間 字 間 出 力 制 御	行 間 字 間 出 力 制 御	行 間 字 間 出 力 制 御	行間字間 出力制御
	Aか>	行間字間 出力制御	行間 字間 出力制御	行 間 字 間 出 力 制 御	行 間 字 間 出 力 制 御	行 間 字 間 出 力 制 御	行 間 字 間 出 力 制 御	行間字間 出力制御
	Bか=	行間字間 出力制御	行間 字間 出力制御	行 間 字 間 出 力 制 御	行 間 字 間 出 力 制 御	行 間 字 間 出 力 制 御	行 間 字 間 出 力 制 御	行間字間 出力制御
	Lか<	行間字間 出力制御	行間 字間 出力制御	行 間 字 間 出 力 制 御	行 間 字 間 出 力 制 御	行 間 字 間 出 力 制 御	行 間 字 間 出 力 制 御	行間字間 出力制御

DSN=SYS9.DOGRACE.TEXT MEM=EXSOC05A

図8-(a) : 横方向'出力でKANG=1, 2の場合

KANG=1
の場合

	子	御	制	力	出	間	字	:	御	=	子	<
	Fか、	出力制御	行間字間	出力制御	行間字間	出力制御	行間字間	:	出力制御	行間字間	出力制御	行間字間
	Cか、	出力制御	行間字間	出力制御	行間字間	出力制御	行間字間	:	出力制御	行間字間	出力制御	行間字間
	Aか>	出力制御	行間字間	出力制御	行間字間	出力制御	行間字間	:	出力制御	行間字間	出力制御	行間字間
	Bか=	出力制御	行間字間	出力制御	行間字間	出力制御	行間字間	:	出力制御	行間字間	出力制御	行間字間
	Lか<	出力制御	行間字間	出力制御	行間字間	出力制御	行間字間	:	出力制御	行間字間	出力制御	行間字間

KANG=2
の場合

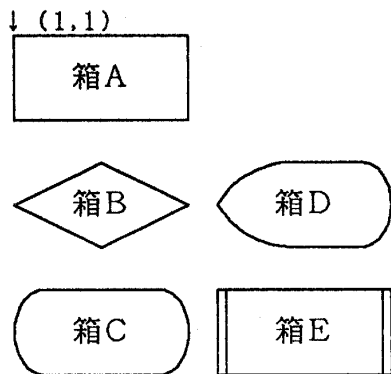
	子	御	制	力	出	間	字	:	御	=	子	<
	Fか、	出力制御	行間字間	出力制御	行間字間	出力制御	行間字間	:	出力制御	行間字間	出力制御	行間字間
	Cか、	出力制御	行間字間	出力制御	行間字間	出力制御	行間字間	:	出力制御	行間字間	出力制御	行間字間
	Aか>	出力制御	行間字間	出力制御	行間字間	出力制御	行間字間	:	出力制御	行間字間	出力制御	行間字間
	Bか=	出力制御	行間字間	出力制御	行間字間	出力制御	行間字間	:	出力制御	行間字間	出力制御	行間字間
	Lか<	出力制御	行間字間	出力制御	行間字間	出力制御	行間字間	:	出力制御	行間字間	出力制御	行間字間

DSN=SYS9.DOGRACEN TEXT MEM=EXSOC05B

図8-(b) |縦方向|出力でKANG=1, 2の場合

この場合のHINT, VINTをSET文で指定していない場合は、0と1.5が与えられる。

図を作成する場合、BOXの位置を直接指定するよりもBOXとBOXを相対的な位置で指定した方が後の修正が容易である。BOXの相対的な指定方法について図9に添って以下に述べる。

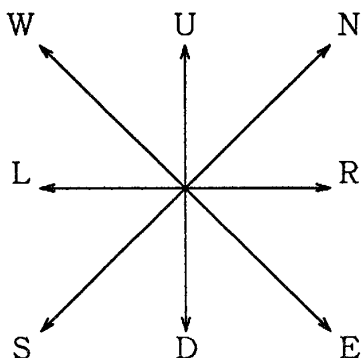


BOX間の相対表現

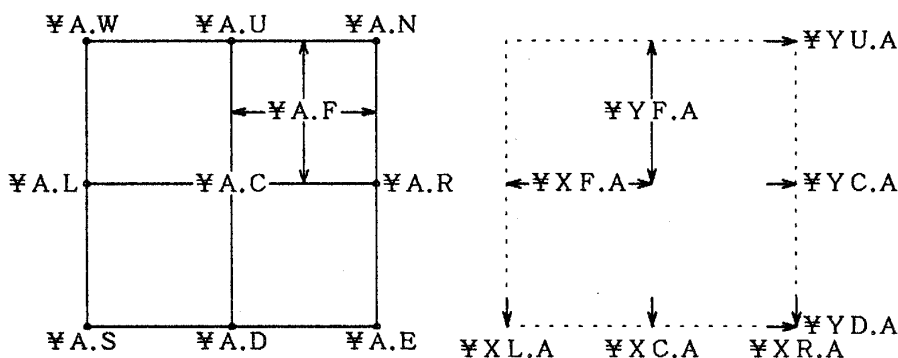
DSN=SYS9.DOGRACE.TEXT
MEM=EXSOC06

```
INIT 108 45 M1
SET CHGT=1 XFCT=3 YFCT=1.5
A:PROC 1/1>E LHGT=0.8 '箱A':('¥.W+U.0.1+L0.4) ↓(1,1)'
B:IFBOX ¥.D+D.YFCT*2 '箱B': * 座標式
C:TERM ¥.D+D.YFCT>D '箱C': * 方向指示子のある座標式
D:DISP ¥B.R+R.CHGT>R '箱D': * 良く使用する座標式
E:SUBR ¥XC.D/¥YC.C ¥XF.D:¥YF.C '箱E':
F:DUMMY (¥C.S+¥E.E)*E.0.5+D.1>D (¥XR.E-¥XL.C)*0.5:YFCT -
* 'BOX間の相対表現':
SET DSTP=1 DINT=1.3 * これより文章出力
TXTLIB SYS9.DOGRACE.TEXT
¥EXSOC06 ¥(¥XR.E/¥YU.A)+R.2>E 20:20 CONV OFF
TEXT ¥F.S+D.2 KHGT=0.8 STEP=0.62 'DSN=SYS9.DOGRACE.TEXT'
TEXT ¥F.S+D.3 KHGT=0.8 STEP=0.62 'MEM=EXSOC06'
OUTPUT
END
```

図9 BOXの相対的な位置指定



方向演算子
方向指示子



箱AのBOX座標定数
座標定数 ¥ (数式/数式)

箱AのBOX要素定数

DSN=SYS9.DOGRACE.TEXT MEM=EXSOC07

図10 座標式に用いる方向と定数

箱Aの座標は1/1>Eであり、これは座標(1,1)の右下側方向EにBOXを描けという指定である。このように>の後に、図10のような方向指示子を指定しBOXを配置できる。なお、>Eの代わりに、=Wと書いてよい。=Wは指定した位置が箱の左上として描く指示となる。

箱Bの座標は¥.D+D.YFCT*2であり、このような書き方を座標式という。これは直前のBOX座標の下側(¥.D)からさらに下方(D.)へYFCT×2の位置を中心として配置される。¥.Dは¥A.Dと書くのと同じで、直前のBOXラベルの場合はラベル名を省略できる。¥A.DをBOXの座標定数といい、図10に示したものである。なおD.YFCT*2のD.は方向演算子であ

り、座標式中で用いることができる。また、文中の . は省略可能である。

箱Cや箱Dの座標式は箱Aや箱Bの方法を組み合わせで作成したものである。

箱Eの座標は $\$XC.D/\$YC.C$ であり、これはX座標を箱DのX方向に関する中心 ($\$XC.D$) とY座標は箱CのY方向に関する中心 ($\$YC.C$) を箱Eの中心として配置する。このような $\$XC.$ や $\$YC.$ をBOXの要素定数といい、図10のようなものがある。この場合の箱の大きさも要素定数を用いた。

箱Fの座標は $(\$C.S+\$E.E)*E0.5+D.1>D$ であり、これは箱Cの左下側 ($\$C.S$) と箱Eの右下側 ($\$E.E$) の中点 ($*E0.5$) から下方へ1下がり ($D.1$)、その下側に ($>D$) 箱を配置する。なお、DUMMYは枠のないBOX文である。E0.5は座標定数 $\$(0.5,0.5)$ と書いたのと同じであり、S0.5は $\$(-0.5,0.5)$ と同じになる。各々の軸方向に注意するとよい。なお、この場合のX方向の箱のサイズは $(\$XR.E-\$XL.C)*0.5$ であり、箱Cの左側から箱Eの右側までのサイズとなる。また、箱Aの文字列出力に

' ($\$.W+U0.1+L0.4$) ↓ (1,1) '

と書かれている。このように最初の ' や | の直後に方向指示子のない (座標式) がある場合は、補助出力文字列となり、指定された位置に描かれる。

☒ これ以後、作画プログラム中にTEXTLIB データセット名と $\$$ メンバで表わされ文章BOXのあるものはプログラムの入ったデータセットとメンバ名である。TEXTLIB文から $\$$ メンバ名まではプログラムを出力するための文である。この利用方法は後述する。

これらの位置指定の他に、表を作成する場合に便利な表座標式がある。表座標式については図11に添って述べる。

図11の箱A2を作画しているPROC文は箱A1の次の文であり、位置とサイズ指定が同時に行われる表座標式

| $\$.R$ | が使用されている。表座標式は | で囲んで書く。これは直前のBOXの右側 ($\$.R$) に同じサイズの大ききで描くことを指定している。直前のBOXのラベルがA1であるから | $\$A1.R$ | と書いても良い。Rの他に、L左、U上、D下の4種類がある。箱A3は箱A2の右横にX軸方向のサイズを変えて描いたものであり、表座標式は | $\$.R X1$ | と指定されている。X1がX軸方向のサイズ変更でblankやコンマで区切って書く、この場合は3となる。箱B1のようにY軸方向に箱を出力すると2番目のY1はY軸方向のサイズの変更を表す。なお、ラベルFRの箱座標は | $\$A1.W\&\$B3.E$ | である。このように座

標式&座標式と書いた場合は、2点を対角 (コーナ) とするBOXを描く。なお、その後のNPEN

箱A 1	箱A 2	箱A 3
箱B 1	箱B 2	箱B 3

```
INIT 108 84 M1
SET X1=3 Y1=0.75 NPEN=0
A1:PROC 1/1>E 2:1 '箱A 1':
PROC |$.R| '箱A 2':
A3:PROC |$.R X1| '箱A 3':
B1:PROC |$A1.D Y1| '箱B 1':
PROC |$.R| '箱B 2':
B3:PROC |$.R X1| '箱B 3':
PR:PROC |$A1.W&$B3.E| NPEN=3
TEXTLIB SYS9.DOGRADE.TEXT
SET DSTP=1.0 DINT=1.3
$EXSOC08 $B1.S+D.1>E 20:20 -
* CONV OFF
OUTPUT
END
```

図11 表座標式

=3 は線の太さの一時的な変更を表す。この場合は、外側が太線となっている。このようなNPENをキーオペランドという、キーオペランドについては次節で述べる。

3. 4 標準BOX文のキーオペランド

標準BOX文では前例のNPENを含め以下のキーオペランドが使用可能である。それらの働きは図12を参照するとよい。なお、アンダーラインの付いたキーオペランドはSET文で設定できる。キーオペランドで指定した場合は、その文のみで有効となる。

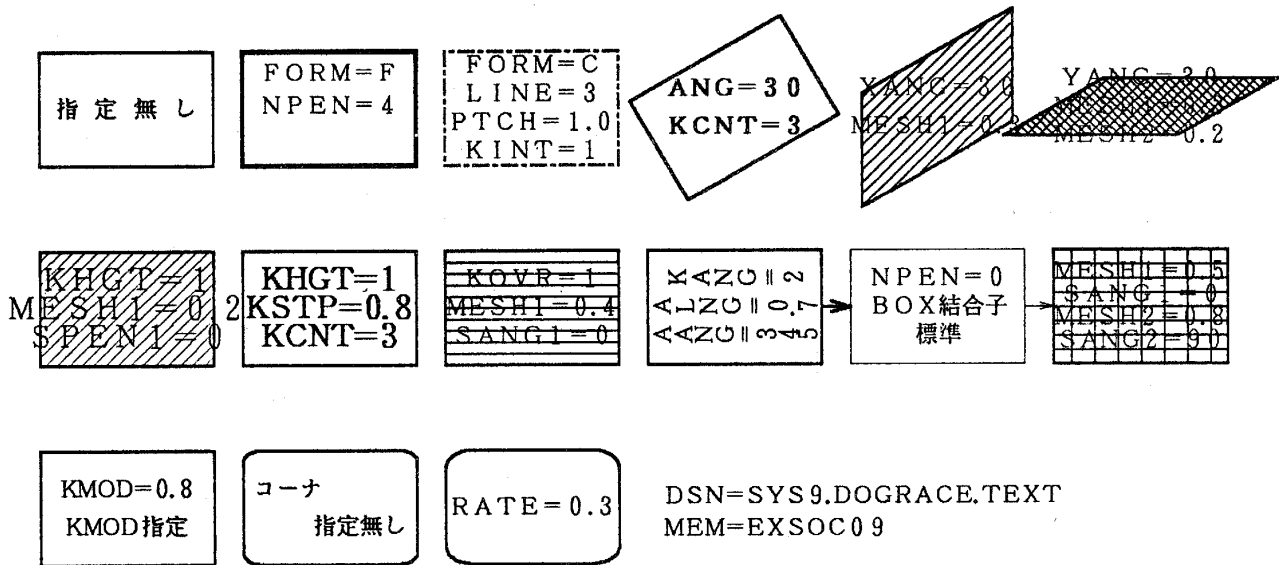


図12 BOX文のキーオペランドと結果

NPEN=0, 1, 2, 3, 4

枠の線の太さを指定する。

KOVR=1, 2

BOX内の文字列を1ならば枠と重ね合せ、2ならば文字列と重なる部分の枠や文は描かない。

KCNT=1 ~ 3

BOX内の文字列の太さを指定する。

KANG=1 ~ 4

BOX内の文字を90° × (回数-1) 反時計方向に回転する。

KHGT (又は HGT) = 高さ

BOX内の文字の高さを指定する。なお、SET文でKHGTが0 (省略時) の場合はCHGTの値となる。この場合は文字列がBOX内に納まらない場合は、BOX内に納まるような値に修正される。文字の高さは0.16mm, 0.24mm, 0.30mm, 0.40mm, 0.64mm, 1.28mmの6種類である。なお、論理座標の倍率が8分の1インチ単位の場合は、0.30mmを1として換算せよ。

KSTP (又は STEP) = 字間

BOX内の文字列の字間を指定する。右揃え, 左揃え, 中央

	揃えの場合に有効となる。KSTP=0の場合は CHGT の値となる。
<u>KBLK</u> =ブランク率	ブランクを出力する場合の字間 KMOD に対する割合を指定する。
<u>KMOD</u> =字間率	EBCDICコード文字列の字間を指定した字間に KMOD の値を乗じたものとする。
FORM=F, C, L, A, B	行間出力制御子であり、行間の制御を行う (図8参照)。
<u>KINT</u> =行間	BOX内の文字列の行間を指定する。F, C, Lの場合に有効となる。
<u>LHGT</u>	補助出力文字列の文字の高さを指定する。
<u>LSTP</u>	補助出力文字列の文字の字間を指定する。
LANG=1 ~4	補助出力文字列の文字を90° × (回転-1) 反時計方向に回転する。
<u>LINE</u> =0 ~99	枠の線種を指定する。0は実線, 1は破線, 2は点線, 3は一点鎖線であり、99までの値を書ける。なお、小数点以下の数を指定した場合は、小数点以下の値を10倍した値が、PTCH に乗せられる。
<u>PTCH</u> =間隔	LINEの値が1~3の場合のピッチを指定する。
ANG= 回転角	枠をBOXの中心を軸にして回転させる角度を指定する。なおラジアン角で指定する場合は RAD= ラジアン角とする。なお、角度が負の場合は、回転後BOXの中心のX軸回りに180° 回転する。
XANG=X軸の傾斜角	BOXのX軸の方向のみを傾ける。なお、ラジアン角で指定する場合はXRAD=ラジアン角とする。
YANG=Y軸の傾斜角	BOXのY軸の方向のみを傾ける。なお、ラジアン角で指定する場合はYRAD=ラジアン角とする。
<u>RATE</u> =0.01 ~0.99	BOXでコーナがカットされているもの (図12の TELE など) について、そのカットする割合を指定する。0の場合はシステムで決めた値となる。
MESH=斜線間の長さ	BOX内に斜線を引く場合に、その斜線間の距離を指定する。但し、閉じたBOXのみ有効である。
<u>MLIN</u> =斜線の線種	斜線の線種を指定する。線種は LINE と同じである。
MESH2=斜線間の長さ	BOX内の2度目の斜線について指定する。
<u>SANG</u> =斜線の角度	MESHで指定された線の角度を指定する。

省略時は45°となる。

SANG2=斜線の角度

MESH2で指定された線の角度を指定する。

省略時はSANG-90°となる。

SPEN=斜線のペンの太さ

MESHで指定されたペンの太さを指定する。

SPEN2=斜線のペンの太さ

MESH2で指定されたペンの太さを指定する。省略時はSPENの値となる。

ALNG=矢印の長さ

BOXを次のBOXと結合する場合の矢印の長さを指定する。

AANG=矢先の角度

BOXを次のBOXと結合する場合の矢先の角度を指定する。

180°を超える場合は、360°から引いた値とし矢先を塗りつぶす。

3.5 TEXT文

TEXT文は文字列の作画に用いる。TEXT文の実例を図13に示した。

TEXT文は指定された位置に文字列を出力する文である。TEXT文は次のように書く。

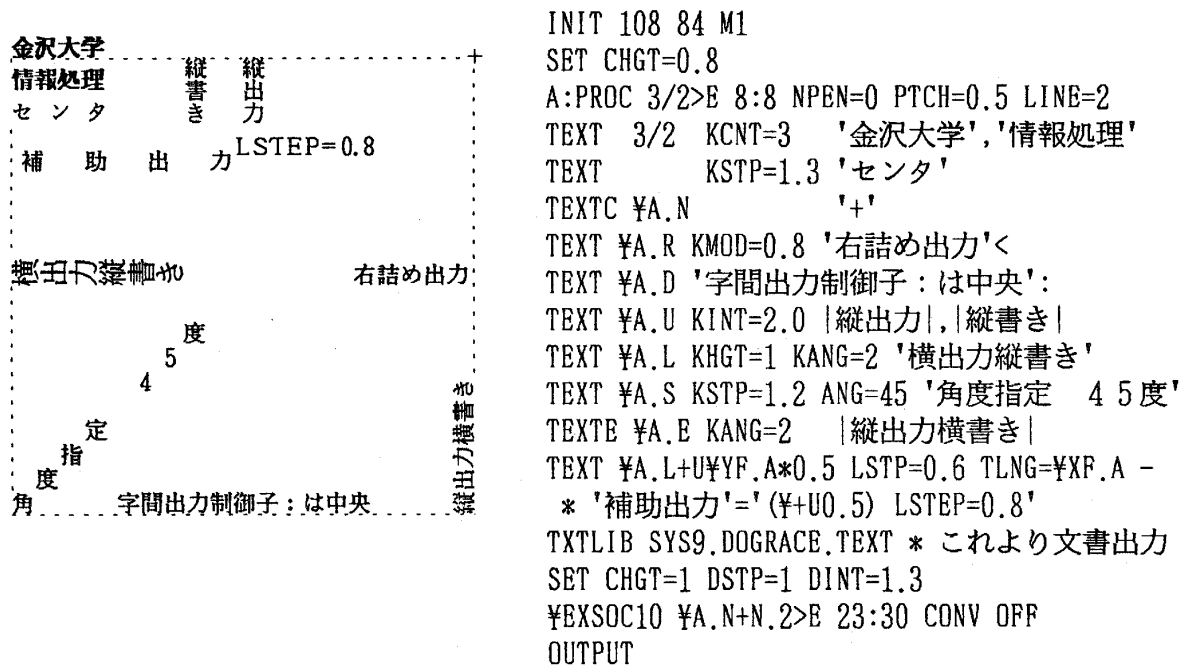


図13 TEXT文の出力とそのプログラム

TEXT [位置] [キーオペランドの並び] '文字列' 字間出力制御子

または |文字列| 字間出力制御子の並び

位置はBOX文で説明したX軸/Y軸または方向指示子のない座標式で指定する。文字列は指定された位置から'で囲まれた場合は横方向に、|で囲まれた場合は縦方向に書く。なお、キーオペランドについて以下に示す。

KHGT, KANG, KSTP, KBLK, KMOD, KCNT, KOVR, LHGT, LSTP はBOX文と

同じ意味を持つ。

ANG (または RAD) = 角度 文字列を指定された角度に従って出力する。

TLNG=出力長 出力文字列の出力の長さを指定する。字間出力制御子>, <, =, : のとき有効となる。

また、TEXT文では横書きの場合は、文字の左下が指定された位置となるが、文字の中央や左上などへ変更する場合は、TEXTC, TEXTL, TEXTU, TEXTW, TEXTN, TEXTD, TEXTS, TEXTEなどのTEXT文で指定できる。補助出力文で直前の文字列の最後の位置から出力する場合は、(¥+ . . .)のように、¥のみで指定する。

☒ X軸とY軸の倍率が異なる場合やグラフ座標系では、BOX文やTEXT文で文字の高さ、字間、行間を指定する場合、必ずY軸に関する論理座標系で指定すること。これらのものをキャラクター・ユニットという。

☒ 文字列中に次のような指定があれば文字列の一部が変換される。

◎数値→文字変換

@数式 ([Z] In, In.k, Fn.m, En.m, Gn.m, RL, RS [CまたはB])

nはカラム数、Kは打出し桁数、mは少数点以下の桁数、最後のCまたはBはコンマまたはブランクの挿入であり、I, F, E, GはFORTRANの欄記述子と同じである。Zは最初のブランクを除く。RL, RSはローマ数字に変換する。但し、ローマ数字は1~20までである。

その他に@DATE, @TIMEがあり、それぞれYY/MM/DD, TT:MM:SSが与えられる。また@S@, @Y@, @M@, @D@は年(昭和)、年(西暦の下2桁)月、日が2桁で与えられる。

◎変換子

以後のEBCDICコードで書かれた英小文字に変換する。次の#で元に戻る。

? 以後のEBCDICコードで書かれた英小文字をギリシャ文字に変換する。次の?で元に戻る。

% 文字の高さを1サイズ大きくする。次の%で元に戻る。

* 以後の文字を濃く出力する。次の*で元に戻る。

" 以後の文字を上付きとする。&で元に戻る。

_ 以後の文字を下付きとする。&で元に戻る。

これらの文字を出力するには##のように2個並べて書く。

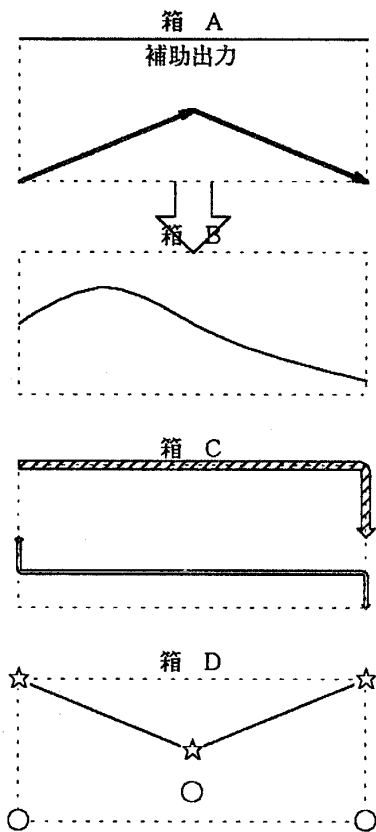
3.6 線文

線等を描く文には以下のものがある。なお、作画例とプログラムを図14に示した。

LINE文 指定された座標点間を直線で結ぶ。

DLINE文 指定された座標点間を直線で結び、両端に矢印を書く。

CLINE文 指定された座標点間を曲線で結ぶ。



```

INIT 108 84 M1
SET KHGT=0.8 KMOD=0.8 KSTP=KHGT XFCT=6 YFCT=2.5
SET LHGT=0.8 LSTP=0.8 XS=1 YS=1 PTCH=0.3
A:PROC XS/YS>E LINE=2 NPEN=0 '(%U+U0.2) 箱 A':
  LINE XS/YS XS+XFCT*2/YS '(%D1) 補助出力':
  LINE NPEN=3 AANG=345 %A.S,%A.C>%A.E>
B:PROC %A.D+D.YFCT>D LINE=2 NPEN=0 '(%U+U0.2) 箱 B':
  CLINE %B.L,%+U.YFCT*0.5+R.XFCT*0.5,%B.C,%B.R+D2
  ARROW WIDTH=2 %A.D,%B.U
C:PROC %B.D+D.YFCT>D LINE=2 NPEN=0 '(%U+U0.2) 箱 C':
  XARROW MESH=0.3 %C.W,.H,%C.R
  DARROW WIDTH=0.2 %C.L,.Y,%C.E
D:PROC %C.D+D.YFCT>D LINE=2 NPEN=0 '(%U+U0.2) 箱 D':
  LINE %D.W %D.C %D.N
  SYMBOL '☆' %D.W %D.C %D.N
  SET X01=%XL.D X02=%XC.D X03=%XR.D
  SET Y01=%YD.D Y02=%YD.D-1 Y03=%YD.D
  SYMBOL '○' REP(1:3)(X,Y)
  SET DINT=1.3 DSTP=1 CHGT=1
  TXTLIB SYS9.DOGRACE.TEXT
  %EXSOC12 %A.N+R2>E 40:40 CONV OFF
OUTPUT

```

図14 線文による作画例とプログラム

- | | |
|---------|-----------------------------|
| ARROW文 | 指定された座標点を片方向太矢印で結ぶ。 |
| DARROW文 | 指定された座標点を両方向太矢印で結ぶ。 |
| XARROW文 | 指定された座標点を閉じた片方向太矢印で結ぶ。 |
| VBAR | グラフ座標系で指定された座標点までの縦棒グラフを描く。 |
| HBAR | グラフ座標系で指定された座標点までの横棒グラフを描く。 |
| SYMBOL | 指定された座標点に指定されたシンボル記号を描く。 |

なお、ARROW文、XARROW文、DARROW文、VBAR、HBARはBOXで紹介したMESHに関するオペランドの指定ができる。

線文は次のように書く。

線文 [SYMBOLの場合'シンボル記号'] [キーオペランドの並び] 座標点・区切記号・[補助出力・文字列]の並び

線文に共通なキーオペランドには以下のものがある。

NPEN, LINE, PTCH, KOVR, LHGT, LSTP, LANG はBOX文と使用方法が同じである。

◎ARROW文などに有効なキーオペランド

MESH, SPEN, SANG, MESH2, SPEN2, SANG2 はBOX文と使用方法が同一である。

WIDE (またはWIDTH) ……太矢印または棒グラフの幅を指定する。

◎SYMBOL文に有効なキーオペランド

KHGT ……シンボルの高さを指定する。

◎座標点の指定方法

座標点の指定方法には次のようなものがある。

数式/数式……例えば、 $1/5$, $\%XL.A+3/YS$ など

方向指示式のない座標式……例えば、 $\%A.E$, $\%A.D+D.YFCT$, $\%+R.2$ (但し、この場合の $\%$ は直前に指定された座標点), $\%(XS/YS)+D.2$

REP (n1:n2:n3) (x,y) ……変数の繰り返し (n1 から n2 まで n3 増しに) 例えば、REP (1:3) (A,B) は A01/B01, A02/B02, A03/B03 と同じ。n_i は数式が許され、1 から 99 までの値である。

座標点結合演算子…… .V 直前の座標点と垂直で、直後の座標点と水平な座標点。

.H 直前の座標点と水平で、直後の座標点と垂直な座標点。

.X 直前の座標点と直後の座標点の midpoint の水平方向の座標要素を有し、直前の座標点の垂直方向の座標要素および直後の垂直方向の座標要素を有する 2 つの座標点。

.Y 直前の座標点と直後の座標点の垂直方向の座標要素を有し、直前の水平方向の座標要素および直後の水平方向の座標要素を有する 2 つの座標点。

なお、これらの区切り記号として、, やブランクを用いれば良い。LINE文, DLINE文, CLINE文の座標点の区切り記号として、記号>を用いた場合は矢印を、記号<を用いた場合は半弧を描く。矢印の長さや角度は ALNG と AANG で半弧の大きさは WIDE で指定する。

データセットからの座標点の入力は

DSN= データセット名 (メンバ名) [FIRST=n1] [LAST=n2] [INTERVAL=n3] [COL=m]

で指定する。省略時は各々が n1=1, n2= 最終値, n3=1, COL=80 となる。

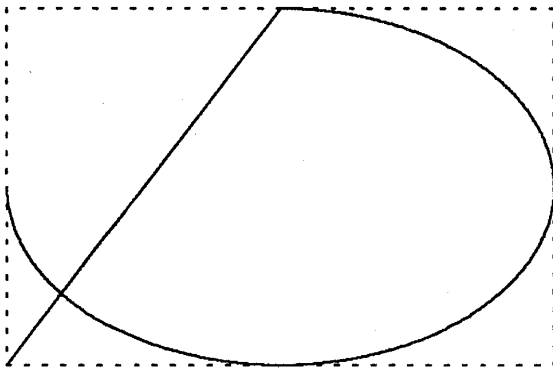
◎線文補助出力文字列

' または | で囲み、' ($\%+ \dots$) 文字列' : のように書く。この場合の $\%$ は直前の 2 個の座標点の左 (上) 側の座標点を表し、文字列を出力する長さが 2 個の座標点の距離となり、出力長さが指定された TEXT 文と同じ解釈がなされる。但し、文字の高さや字間は LHGT, LSTP で指定する。

3. 7 その他の線文

線を描く文には、3. 6 で説明したもの以外に ELLIPES 文と MOVE 文がある。実例を図 15 に示した。

ELLIPES 文は楕円を描く文である。ELLIPES について以下に示す。



```

INIT 200 100 20 10
A:PROC 5/5>E 30:20 -
    * LINE=2 PTCH=2
MOVE ¥A.S 3 * ペンの位置移動
MOVE ¥A.U 2 * ペンの移動作画
ELLIPES ¥A.C ¥A.F -
    * ANG1=90 ANG2=-180 >
SET CHGT=3 DSTP=CHGT DINT=CHGT*1.3
TXTLIB SYS9.DOGRACE.TEXT
¥EXSOC13 ¥A.N+R10>E 80:80 CONV OFF
OUTPUT

```

図15 MOVE文とELLIPES文の例

ELLIP 中心座標点 X軸長:Y軸長 キーオペランドの並び [>]

中心座標点およびX軸長:Y軸長の指定方法はBOX文の位置座標と大きさの指定と同じである。

キーオペランド

NPEN, LINE, PTCH についてはBOX文と同じである。

ANG1 (またはRAD1) …作画開始角度を与える。省略時は0度。

ANG2 (またはRAD2) …作画終了角度を与える。省略時は360度。

ANG (またはRAD) …中心座標点を中心とし回転する角度を指定する。

最後の記号>は終了点で矢印を描く場合に指定する。この場合ALNG, AANGの指定が可能である。

◎ MOVE文は座標点の移動であり、PLOT サブルーチンの働きをする。

MOVE 座標点 ペン番号 [NPEN=n]

座標点は数式/数式または方向指示子のない座標式で指定する。

ペン番号が3ならば座標点移動、2ならば直前の座標点から指定された座標点までを実線で結ぶ。

3.8 各種のBOX文

標準BOX文以外にユーザの定義したBOX (利用者定義BOX) を標準BOXと同様に利用することができる。

線画BOX文 …地図などの線画を描く文

イメージBOX文…他のソフトウェアで作画した図の取込み

ソースBOX文 …完成済のプログラムを図の一部に組み込む場合

上記のBOX文については、別の機会に解説する。

3. 9 作画領域制御BOX文

作画領域を制御するBOX文には以下のようなBOX文がある。実例を図16に示した。

- FIELD文 …作画可能領域を設定する。
- ENDFIELD文 …作画可能領域の設定を解除する。
- SINIT文 …論理座標と作画方向を設定する。
- ENDSINIT文 …論理座標と作画方向を元に戻す。
- ERASE文 …作画された図形を消去する。

◎FIELD文は作画可能範囲を設定するBOX文であり、次のように書く。

FIELD 位置 サイズ [OUTまたはIN]
位置やサイズの指定方法はBOX文と同じである。キーオペランドOUTは指定されたBOXの外側を作画可能域とし、INは内側を作画可能域とする。省略時はOUTが指定されたものとする。なお、解除はENDFIELD文で行う。

◎SINIT文は論理座標の原点と作画方向を再設定するBOX文であり、次のように書く。

SINIT 位置 サイズ [IANG=n]
位置やサイズの指定方法はBOX文と同じである。キーオペランドIANGは作画方向の指定であり、INIT文で指定した方向からさらに反時計方向に90度×(n-1)回転した方向に座標系が設定される。なお、解除はENDSINIT文で行う。

◎ERASE文はすでに作画されているものを消去する文であり、次のように書く。

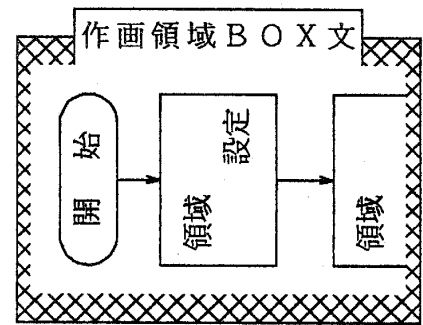
ERASE 位置 サイズ
位置やサイズの指定方法はBOX文と同じである。

3. 10 グラフ座標

グラフを描く場合はグラフ座標を設定することにより、簡単にグラフを描くことが可能となる。グラフ座標の設定は次のように書く。

GRAPH 位置 サイズ X軸開始値：X軸終了値 Y軸開始値：Y軸終了値 [X=LOG10] [Y=LOG10]

例えば、次の様に指定した場合は、図17のような座標系となる。



```
INIT 100 70 M1
SET CHGT=1
A:PROC 3/2>E 5:1 -
  * '作画領域BOX文'>
B:FIELD ¥A.C ¥A.F OUT
C:PROC ¥A.C>D ¥XF.A+2:5 -
  * MESH=0.5 MESH2=0.5
ENDFIELD
D:ERASE ¥C.C ¥C.F-¥(0.5,1)
SET XFCT=3 YFCT=2
SINIT ¥D.C ¥D.F IANG=2
  TERM 1/1>E 3:1 '開始'>>
  PROC '領域','設定'<>
  PROC '領域','解除'<
ENDSINIT
SET DINT=1.3,DSTP=CHGT
TXTLIB SYS9.DOGRACE.TEXT
¥XSOC14 ¥C.S+D1>E -
  * 30:30 CONV OFF
OUTPUT
END
```

図16 作画領域制御BOX文の例

GRAPH 6/3>E 10:8

-3.14:3.14 -1:1

なお、Y軸を対数グラフにする場合は、 $Y=LOG10$ のように指定する。グラフ座標系の解除は、ENDGRAPH文である。

㊦ グラフ座標系では、論理座標系で作成されたボックス定数等は使用できない。なお、論理座標系の単位をグラフ座標系の単位に変換するには、システム変数 GRPX, GRPY をX軸方向、Y軸方向の値に乗ずるとよい。

また、文字に関するキーオペランドはグラフ座標系であっても、キャラクタ・ユニットで指定しなければならない。

◎軸作画文

グラフ座標で軸を作画する場合は、X軸はXAXIS文でY軸はYAXIS文で描く。それらの書き方を以下に示す。また、グラフの作画例を図18に示した。

XAXISまたはYAXIS キーオペランドの並び

キーオペランドには以下のものがある。NPEN, LINE, PTCH はBOX文と同じである。

POSITION	軸を作画する位置を、XAXISの場合はY軸のグラフ座標系で、YAXISの場合はX軸のグラフ座標系で指定する。省略時は開始位置。
START	軸メモリの作画開始位置を指定する。省略時は開始位置。
END	軸メモリの作画終了位置を指定する。省略時は終了位置。
MAJOR	軸メモリを作画する分割数を指定する。省略時は5となる。なお、値が1以下の場合には作画しない。
LMAJOR	軸メモリの長さをキャラクタ・ユニットで指定する。省略時はCHGTとなる。なお、値が正と負により作画方向が異なる。値が正の場合、XAXISの場合は軸に対して時計回り方向となり、YAXISの場合は軸に対して反時計回り方向となる。
MINOR	MAJORで分割された1区間を細分割する。分割数を指定する。1以下の場合には作画しない。省略時は1となる。

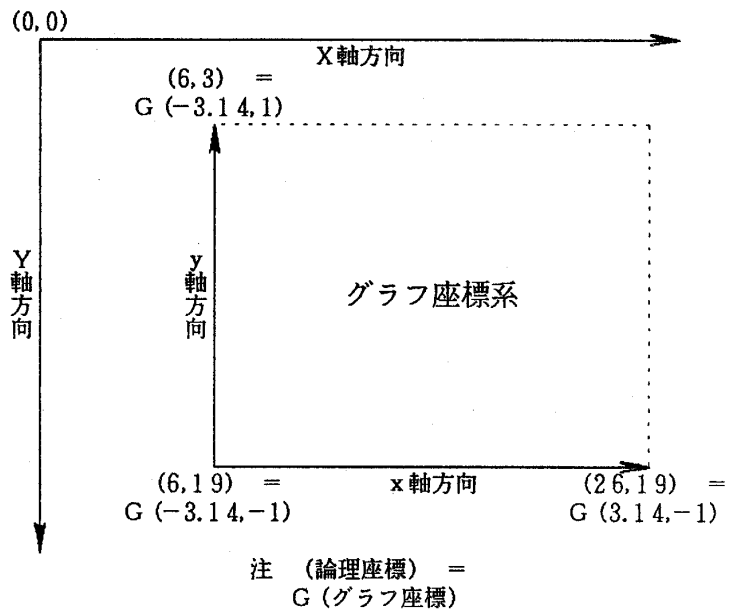
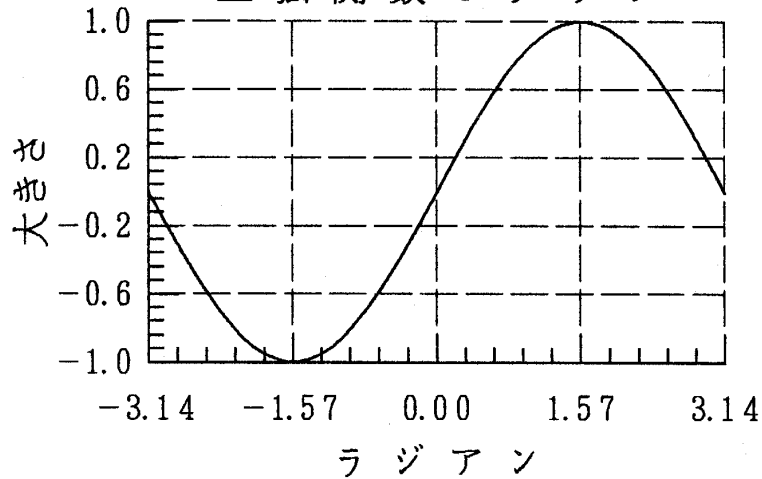


図17 グラフ座標系

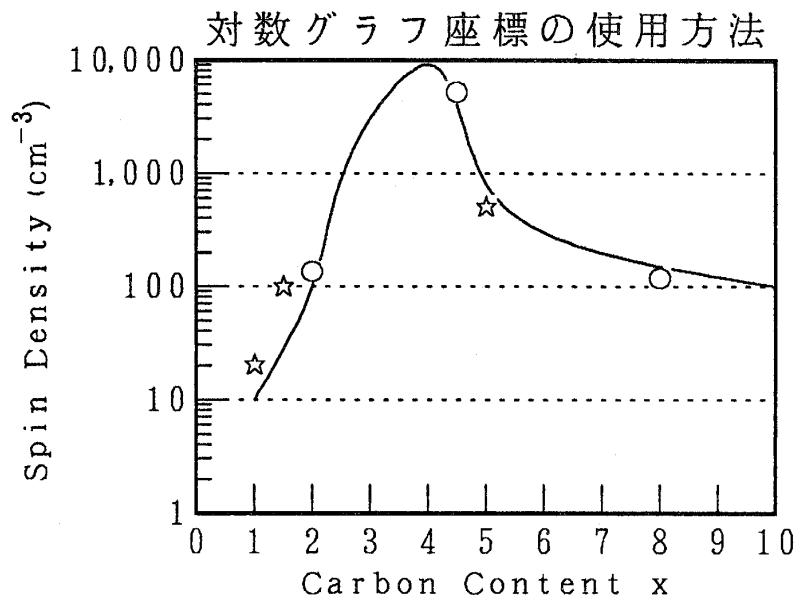
正弦関数のグラフ



```

INIT 100 76 M1
A:GRAPH 25.5/2>D 10:6 -3.14:3.14 -1:1
XAXIS POSITION=-1 - * POS=YS 軸の作画位置
* START=-3.14 - * STA=XS 軸メモリの作画開始値
* END=3.14 - * END=XE 軸メモリの作画終了値
* MAJOR=4.0 - * MAJ=5 軸メモリの分割数: >1で有効
* LMAJOR=-1. - * LMA=CHGT 軸目盛の長さ* (*標準座標値)
* MEMORY='(F5.2)' - * --- 軸目盛値の作画 (書式の指定)
* LHGT=1 - * LHGT=CHGT 目盛値の高さ*
* LSTP=0.8 - * LSTP=LHGT 目盛値の字間*
* MINOR=5 - * MIN=5 軸目盛間の分割数: >1で有効
* LMINOR=-0.5 - * LMI=LMA/2 細分割目盛の長さ*
* TITLE='ラジアン': - * ---- 軸タイトル
* KHGT=1.4 - * KHGT=CHGT 軸タイトルの高さ*
* KSTP=2 * KSTP=KHGT 軸タイトルの字間*
YAXIS POSITION=-3.14 - * POS=YS 軸の作画位置
* START=-1 - * STA=XS 軸メモリの作画開始値
* END=1 - * END=XE 軸メモリの作画終了値
* MAJOR=5 - * MAJ=5 軸メモリの分割数: >1で有効
* LMAJOR=-1.0 - * LMA=CHGT 軸目盛の長さ* (*標準座標値)
* MEMORY='(F4.1)' - * --- 軸目盛値の作画 (書式の指定)
* LHGT=1 - * LHGT=CHGT 目盛値の高さ*
* LSTP=0.8 - * LSTP=LHGT 目盛値の字間*
* MINOR=5 - * MIN=5 軸目盛間の分割数: >1で有効
* LMINOR=-0.5 - * LMI=LMA/2 細分割目盛の長さ*
* TITLE='|大きさ|>' - * ---- 軸タイトル
* KHGT=1.4 - * KHGT=CHGT 軸タイトルの高さ*
* KANG=2 *
XREF MAJOR=4 NPEN=0 LINE=1 * リファレンス線作画
YREF MAJOR=5 NPEN=0 LINE=1 * リファレンス線作画
LINE DSN=SYS9.DOGRACE.TEXT(SINDT) * グラフ座標による作画
ENDGRAPH * グラフ座標の終了
TEXT ¥A.U+U0.5 HGT=1.4 STEP=2 '正弦関数のグラフ':
TXTLIB SYS9.DOGRACE.TEXT
SET DSTP=1 DINT=1.1 CHGT=1
¥EXSOC16 ¥A.D+D4>D 20.0:30 CONV OFF
OUTPUT
END
    
```

図18 グラフの作画例(1)



```

INIT 108 84 M1
TTL:DUMMY 28/0>D 10:1 KHGT=1.4 '対数グラフ座標の使用方法'>
GRP:GRAPH ¥TTL.D>D 10:8 0:10 1:1.E4 Y=LOG10
* Y=LOG10 Y軸が対数である
XAXIS POS=1 MAJ=10 LMAJ=-1 MEM='(I2)' LHGT=1 LSTP=0.8 -
* TITLE='C#ARBON #C#ONTENT X':KHGT=1 KSTP=0.8
YAXIS POS=0 MAJ=4 LMAJOR=-1 MINOR=10 LMINOR=-0.5 -
* MEM='(I7C)' LHGT=1 LSTP=0.8 -
* TITLE='|S#PIN #D#ENSITY(CM"-3&)|': KHGT=1 KSTP=0.8 KANG=2
YREF MAJOR=4 PTCH=0.5 LINE=2 * リファレンス線作画
CLINE 1/10 2/100 3/3010 4/9.1E3 5/810 6/3.E2 8/150 10/100
SYMBOL '☆' HGT=0.8 1/20 1.5/98 5/500
SYMBOL '○' HGT=0.8 2/130 4.5/5E3 8/115
ENDGRAPH * グラフ座標の終了
PROC ¥GRP.C ¥GRP.F NPEN=3
TXTLIB SYS9.DOGRACE.TEXT
SET CHGT=1 DSTP=1 DINT=1.2
¥EXSOC17 ¥GRP.D+D4>D 18:40 CONV OFF
OUTPUT
END

```

図18 グラフの作画例(2)

- LMINOR** 細分割した軸メモリの長さを指定する。作画方向はLMAJORの指定方法と同じ。
- MEMORY** MAJORで分割された位置に軸メモリを作画する場合に、その書式を指定する。書式は以下のように書く。
- '(In,Fn.m,Gn.m,En.m [,BまたはC])'
- LHGT** メモリ値の文字の高さを指定する。
- LSTP** メモリ値の字間を指定する。
- TITLE** 軸タイトルを指定する。タイトルは'タイトル'字間制御子で指定する。
- KHGT** タイトルの文字の高さを指定する。

KSTP タイトルの字間を指定する。

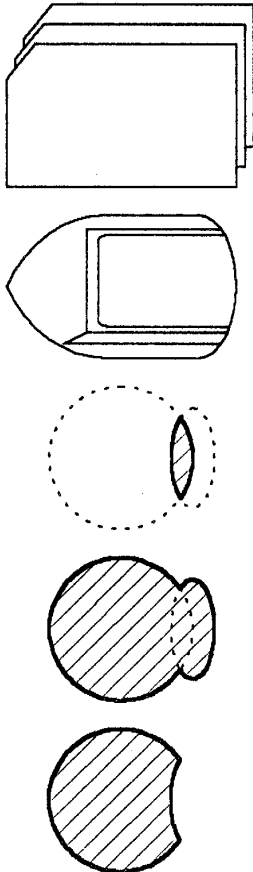
KANG タイトルの文字の傾きを指定する。

◎リファレンス線の作画

リファレンス線の作画は以下のように指定する。

XREFまたは**YREF** キーオペランド

キーオペランドとして**NPEN**, **LINE**, **PTCH**, **START**, **END**, **MAJOR**が指定できる。



```
INIT 100 70 M1
  SET XFCT=4 YFCT=2.5
A: CARD 1.5/1.5>E
  +CARD ¥+¥(0.3,-0.7) * カードの重ね合せ
  +CARD ¥+¥(0.3,-0.7) * 外側に作画
C: DISP ¥A.D+D1=U
D: -KTELE ¥C.R+L2 4:2 * 内側に作画
  SET XFCT=2.5 YFCT=2.5 PTCH=0.4
E: MARU ¥C.D+D1=U LINE=2
  &MARU ¥E.R ¥E.F*¥(0.3,0.7) LINE=2 * 論理積
  LGCBOX NPEN=3 SPEN=0 MESH=0.5 * 論理BOXの作画
F: MARU ¥E.D+D1=U LINE=2
  |MARU ¥F.R ¥F.F*¥(0.3,0.7) LINE=2 * 論理和
  LGCBOX NPEN=3 SPEN=0 MESH=0.5 * 論理BOXの作画
G: ~MARU ¥F.D+D1=U LINE=2 * 作画しない
  ~.MARU ¥G.R ¥F.F*¥(0.3,0.7) LINE=2 * 論理差
  LGCBOX NPEN=3 SPEN=0 MESH=0.5 * 論理BOXの作画
TXTLIB SYS9.DOGRACE.TEXT
SET DINT=1.3
¥EXSOC25 15/1>E 20:30 CONV=OFF
OUTPUT
END
```

図19 論理BOX文

3.11 論理BOX文

標準BOX文と利用者定義BOX文では、BOX文の前に演算子を付けることができる。演算子には次のものがある。なお、使用例は図19に示した。

- ¬ BOXを作画しない。
- & 直前の論理BOXと指定されたBOXの内側の枠を論理BOXとして記憶する。
- | 直前の論理BOXと指定されたBOXの外側の枠を論理BOXとして記憶する。
- + 直前の論理BOXの外側の領域に指定されたBOXを描く。
- 直前の論理BOXの内側の領域に指定されたBOXを描く。
- 直前の論理BOXから指定されたBOXの差を論理BOXとして記憶する。

論理BOXの作画はLGCBX文で行う。

4. 文書BOX文

このシステムの特長は図形のみならず文書処理（日本語・英語）が可能なことである。図の中に文書を入れたり、文書の中に図を挿入することが可能である。文書は、文章中に日本語（JEFコード）や英語（EBCDIC）と制御記号や制御文字列により構成する。簡単な使用例は図3、図4を参照するとよい。

4.1 短文出力のための文書BOX文

段落のない短文出力の文書BOX文には2通りある。1つはプログラム中に文章を書くもの、もう1つは文章をデータセットから読み込む方法である。文書BOX文の使用例を図20に示した。

文書の出力を任意の大ききさで出力するにはX軸方向の大ききさのみを決め、Y軸は大ききくとるとよい。

出力サイズ

指定サイズ

-¥DATAとしたときは、指定サイズが出力サイズにBOX定数が変更される。

実際のBOXサイズ

出力サイズが決められている場合は、-で出力しDLNGを調べDINTを決め、再度出力するとよい。

```
INIT 108 84 M1
SET DHGT=1,DSTP=1.1,DINT=2
SET KHGT=1,KSTP=1,LHGT=0.8,LSTP=0.8
SET NPEN=1,PTCH=0.3 XF=8 YF=6
A:¥DATA 1/1>E XF:YF
文書の出力を任意の大ききさで出力する
にはX軸方向の大ききさのみを決め、Y
軸は大ききくとるとよい。
¥ENDDATA
PROC ¥A.C ¥A.F '(¥.D+D.1) 指定サイズ':
PROC ¥A.U>D ¥XF.A:DLNG*0.5 -
    * '(¥.D+D.1) 出力サイズ':
B:-¥DATA ¥A.D+D2>D XF:YF -
    * '-¥DATAとしたときは、指定サイズが-
*出力サイズにBOX定数が変更される。'
PROC ¥B.C ¥B.F -
    * '(¥.D+D.1) 実際のBOXサイズ':
TXTLIB SYS9.DOGRACE.TEXT
C:-¥EXSOC18D ¥B.D+D.3>D XF:YF DINT=1
D:¥EXSOC18D ¥C.U>D XF:YF DINT=5//DLNG
PROC ¥D.U>D XF:5//2
¥EXSOC18 ¥A.N+R.3>E 40:40 DINT=1.2 CONV=OFF
OUTPUT
END
```

図20 短文出力の例

◎プログラム中に文章を書く文書BOX文は、以下のように指定する。

¥DATA 位置 大きき キーワードオペランドの並び '文章'

または、文章が数行になる場合は、

¥DATA 位置 大きき キーワードオペランドの並び

文章

¥ENDDATA

◎データセットから文章を読み込む文書BOX文

TXTLIB データセット名

¥メンバ名 位置 大きさ キーワードオペランドの並び

上記の2通りの位置、大きさの指定方法はBOX文と同じである。

キーワードオペランドには以下のものがある。

<u>DHGT</u>	文書出力をする場合の文字の高さを指定する。省略時は CHGT となる。
<u>DSTP</u>	字間の指定を行う。省略時は DHGT となる。
<u>DMOD</u>	日本語に対する英語の字間を DSTP の率で指定する。省略時は 0.8 である。
<u>DINT</u>	文書を出力する場合の行間を指定する。
<u>DCHR</u>	1 のとき大きさが字数と行数とみなし、2 のとき位置と大きさが共に字数と行数とみなす。
<u>DXSK</u>	文書中に図を取り込んだ場合の図の両サイドに空白をあける字間 DSTP の数を指定する。省略時は 3 となる。
<u>DYSK</u>	文書中に図を取り込んだ場合の図の下方に空白行数を指定する。省略時は 1 となる。
COL	データセットから入力する場合の入力カラム数を指定する。
CONV	OFF と指定した場合は、データセットの内容をそのままの形で出力する。
REC	OFF を指定した場合の入力カラム数を指定する。このとき同時に COL を指定すると COL は出力する文字数となる。なお DMOD は無視され、常に 0.5 となる。
<u>ADJS</u>	DHGT=3mm, DSTP=3mm のとき、字間調整を行う単位を 10 分の 1mm 単位で指定する。

㊦ 文書の出力後、システム変数 DLNG に実際に使用した Y 軸方向の長さが与えられる。なお、¥DATA 文や ¥メンバ名文の前にラベルと (マイナス) を付けて A: -¥DATA のように書くと Y 軸方向に関するラベル定数は実際に使用された値となる。また、- の代わりに $\bar{\quad}$ を付けると文書も出力されない。従って決ったサイズの BOX 内に文書を BOX 一杯に配置するには、 $\bar{\quad}$ で調べてから行間の計算をし、実際に出力するとよい。

4. 2 長文出力のための段落BOX文

長文 (段落やページが 2 以上の文) を出力する場合は、段落BOX文で段落を指定し、出力する。段落の指定は次のように行う。なお、使用例は図 2 1 に示した。

¥PAGE m.n 位置 サイズ キーワードオペランドの並び [DATA]

長い文章は用紙内に段組を設定し、その段組のどこから出力するか指定する。出力文書は自動的に段組内に収められる。なお、段毎にページ等挿入するには段定義文にDATA

と指定し、後ろに必要な文を置き、文の最後に ¥ENDPAGE 文を置く。

P AGE=1.1

P AGE=2.1

```
INIT 108 76 M1
SET DHGT=1 DSTP=1.0 DINT= 1.5 ADJS=3
A:¥PAGE 1.1 1/1>E 10:3 DATA * 段の設定
  PROC ¥A.C ¥A.F NPEN=0 LINE=2 PTCH=0.3
  TEXT ¥A.D+D.2 KHGT=0.8 'PAGE=1.1': * 段出力後の処理
  ¥ENDPAGE * 段の終了
B:¥PAGE 2.1 ¥A.R+R3>R ¥A.F DATA
  PROC ¥B.C ¥B.F NPEN=0 LINE=2 PTCH=0.3
  TEXT ¥B.D+D.2 KHGT=0.8 'PAGE=2.1':
  ¥ENDPAGE
  TXTLIB SYS9.DOGRACE.TEXT
  ¥EXSOC19D PAGE=1.1 * 文章の出力段の指定
  ¥EXSOC19 ¥A.S+D3>E 20:30 DINT=1.2 CONV=OFF
OUTPUT
END
```

図 2 1 段組を用いた出力例

m.n の m は段種別であり、n はクラスを表す。m, n とともに 1~9 までである。位置、サイズの指定方法は短文出力の文書 BOX 文と同じ。キーワードオペランドは DCHR が指定されたとき DHGT, DSTP, DINT が有効となる。

DATA は段内文書出力後、文書 BOX 文以外の図形出力を行うとき、それらの文をこれ以後に書き、終りに ¥ENDPAGE 文を置く。実際の出力方法は、短文出力の位置やサイズの代わりに、出力段落を指定する。

TEXTLIB データセット名

¥メンバ名 PAGE=m.n キーワードオペランドの並び

出力は指定された段落から開始され、BOX に文書が一杯になった時点で m+1 の段落に移る。次の段落がない場合はページを換え、最初の段落に戻る。なお、クラス n の変更は文章中で指定できる。例えば、以下のように定義されている場合

TEXTLIB データセット名

¥PAGE 1.1 1/1>E 10:30

B:¥PAGE 2.1 30/1>E 10:30 DATA

SET PG=PG+1

TEXT ¥B.D+D2 ' -@PG(Z3)- '

¥ENDPAGE

¥メンバ名 PAGE=1.1

まず文書が1. 1段に、次に2. 1段となり、2. 1段が終了すると¥PAGE文に2. 1段のDATA が指定されているから、¥ENDPAGE までの文を実行した後、新しいページに移り、1. 1段から出力される。

4. 3 出力文字と入力方法

文書中に出力する文字に日本語文字と英論文用の文字の2種類がある。それらの入力は各々、日本語文字（JEFコード）と英数字（EBCDICコード）で行う。

日本語文字の字種には以下の大きさのものがある。

16ドット, 24ドット, 30ドット, 40ドット, 64ドット, 128ドット

この字種の指定はDHGTで決まる。DHGTの指定の目安は30ドットが8分の1インチあるいは3mmである。文章中で指定された文字サイズDHGTより、ひとまわり大きい字を出力するには%で囲む。また、濃く出力するには*で囲む。

英文用の文字には、ローマタイプ、イタリックタイプ、ギリシャ文字等の大小文字とボールドタイプがあり、大きさは24ドットと30ドットの2種類である。入力方法は英数字で行う。タイプ等の指定は特殊記号を用いる。たとえばギリシャ文字のαは?#A#?と入力する。この?がギリシャ文字への変換、#が小文字への変換の切り換えを表す。使用できる変換記号とその書体を表1に示した。

表1 変換記号と書体

変換記号	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	1	2
	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	1	2
#	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	1	2
*	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	1	2
*#	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	1	2
!	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	1	2
!#	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	1	2
!*	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	1	2
!*#	a	b	c	d	e	f	g	h	i	j	k	l	m	n	o	p	q	r	s	t	u	v	w	x	y	z	1	2
?	A	Β	Δ	E	Φ	Γ	H	I	Ɔ	K	Λ	M	N	O	Π	§	P	Σ	T	Υ	Ψ	Ω	Ξ	X	Z	1	2	
?#	α	β	θ	δ	ε	φ	γ	η	ι	ς	κ	λ	μ	ν	ο	π	§	ρ	σ	τ	υ	ψ	ω	ξ	χ	ς	1	2
?*	A	Β	Δ	E	Φ	Γ	H	I	Ɔ	K	Λ	M	N	O	Π	§	P	Σ	T	Υ	Ψ	Ω	Ξ	X	Z	1	2	
?*#	α	β	θ	δ	ε	φ	γ	η	ι	ς	κ	λ	μ	ν	ο	π	§	ρ	σ	τ	υ	ψ	ω	ξ	χ	ς	1	2

また、記号としてキーボード上の記号+、., =*...など使用してよいが、次のものは制御記号として使用されるため、出力する場合は2個並べて1文字となる。

" _ & ? % * ! 「 」 | _ \$:

したがって、?を出力する場合は??と入力する。なお、数式中で取り扱う記号を表2に示した。たとえば±という記号は¥02¥と入力する。

表2 数式で用いる記号

1の位	0	1	2	3	4	5	6	7	8	9
10の位										
0		-	±	∓	×	÷	=	≠	≡	>
1	+	≠	≠	≧	≦	≧	≡	≫	≪	≡
2	<	:	∞	∥	∥	#	↕	⊥	⊥	∞
3	≠	≠	△	~	∈	∉	⊂	⊄	⊃	⊄
4	→	≠	≧	≧	∪	∩	∨	∧	↔	↗
5	≡	∞	∂	Δ	∅	∀	∃	+	·	∑
6	↘	△	□		L	#		○	○	○
7	∠	[]	{	}	()	丸	え	丸
8	°	□	△	▽	∇	∇	∇	†	§	‡
9	丸	°	°	°	°	°	°	°	°	∴
10	°	°	°	°	°	°	°	°	°	°
11	∴									
20	()	<	>	{	}	{	}	/	
その他	$\forall P I \forall = \prod$ $\forall S I \forall = \sum$ $\forall C I \forall = \oint$ $\forall I N \forall = \int$									

4. 4 文書制御子について

出力文書は文書出力を行うための文章と制御子で構成する。制御子について以下に説明する。なお、以後のnやmは特に断らない限り実数で指定できる。

- @ FM= {L, C, R} @ 揃え制御を行う。Lは左揃え, Cは中央揃え, Rは右揃えである。
- @ PR=n [, m] @ パラグラフ制御を行う。nは先頭行のシフト文字数, mは2行目以後のシフト文字数である。nやmが省略された場合は直前に指定された値となる。また、@ PR ·· @が指定された場合、出力が先頭に位置されている場合は改行せず、1文字でも出力された後であれば改行される。
- @ PR1 @ パラグラフ制御子の第1パラメータ nを変更する。文章中に@PR1 @と置くと、現在位置付けられているカラムがnの値となる。
- @ PR2 @ パラグラフ制御子の第2パラメータ mを変更する。文章中に@PR2 @と置くと、現在位置付けられているカラムがmの値となる。
- @ NL [=m] @ 改行を行う。mは改行の数である。省略時は1となる。なお、この後に続く文章はパラグラフ制御子@ PR ·· @で指定された第2パラメータの位置から出力される。
- @ RC [=記号] @ 一部右詰めを行う。文章中に@ RC @とある場合に、この制御子から@ NL @または@ PR @までの文章が右詰めで出力される。この場合に@ RC=記号@が表れたときに現在位置付けられているカラムから

直後の文章までの間が指定した記号で埋められる。記号を省略した場合はブランクとなる。

- @ PS= n @ 出力カラムを指定する。これ以後の文章の出力カラムを n で指定する。なおマイナスで指定した場合は行末からのカラム数+1となる。
- @ CL [= n] @ 字間の一時的な変更を指定する。 n は現在の字間に対する割合を指定する。たとえば字間を1.2倍にするには@CL=1.2@と指定する。元に戻すには@CL@と指定する。
- @ LN [= n] @ 行間隔の一時的な変更を指定する。 n は現在の行間に対する割合を指定する。
- @ SF [= n] @ 出力する行の位置を一時的に移動する。 n はcm単位で指定する。負の場合は上方へ、正の場合は下方へ移動する。
- @ SP [= n [, m]] @ スペースを空ける場合にその数を指定する。 n はスペースの数、 m は1ならば日本語の字間、2ならば英語の字間としてスペースを空ける。
- @ UL [= n] @文章@ UE [= m] @ アンダラインを引く。 n は線の数を10の位に、太さを1の位に指定する。省略時は11となる。
- @ DN [= n] @ 字上げ文字数を指定する。行末に出力しないカラム数を指定する。たとえば@DN=2@と指定した場合は、これ以後、後2文字分を残して改行される。元に戻すには@DN@と指定する。
- @ NB @ 改段を行う。
- @ NB2 [= n , m] @ , @ NB3 [= n , m] @ 改段の制御を行う。 n は残り行数、 m は改行の数である。@NB2=5,2@と指定した場合に残り行数が5行以上あるならば2回改行し出力する。5行未満であるならば、次段の先頭から出力する。 n , m が省略された場合は直前の値が有効となる。

4.5 英文文章制御子

英文や数式を取り扱う場合の出力制御子について以下に示す。

- @ FM=A @ 英文出力の場合の両端揃えを行う。
- @ WD @ 単語制御は行わない。@WD=0@とすると単語が行中に入らない場合は単語として次行の先頭から印刷される。
- @ AT [=2] @ 字間調整を行う。@AT=2@とした場合、英文のプロポーシオン処理を行う。元に戻す場合は@AT@とする。
- @ TM [= n] @ 英小文字入力の指定を行う。 n が1以上の場合は英小文字入力が可能となり、変換記号#を必要としない。
- @ DL [= n] @ 微小空白記号\$の長さの指定を行う。微小空白を挿入する場合は\$を

- @CF記号 [=n, m]@ 入力する。長さの指定は@ DL=n @で与える。nはcm単位である。位置揃えを行う。指定された記号が表れた位置が出力カラムnの位置になるように前後を調整し出力する。m=1 のとき日本語カラム数、m=2 のとき英語カラム数となる。
- @AL@ 文章中に分数等が表れたとき、改行を自動調整する。

4. 6 数式の取り扱い

数式を取り扱うため各種制御コードが準備されている。以下にそれらについて説明する。

◎括弧の取り扱い。

各種括弧は@ (=n @文章@)=m @で囲み指定する。なお、括弧内に指定できる文章の長さは1行以内に納まらなければならない。括弧の種数を表わすnとmは異種の番号を用いることができる。括弧の種類については表3に示す。なお、9番はダミー括弧である。

上記の他にキーマークを描く@KS@文書@KE@とルートを描くための@RS@文章@RE@がある。

表3 各種括弧について

n	@ (=n@	m @) =m@	n	@ (=n@	m @) =m@
省略	文章	省略 文章	5	文章	5 文章
0	(文章	0 文章)	6	[文章	6 文章]
1	文章	1 文章	7	[文章	7 文章]
2	<文章	2 文章>	8	文章	8 文章
3	文章	3 文章	9	文章	9 文章
4	{文章	4 文章}			

その他 @RT@文章@RE@ $\sqrt{\text{文章}}$ @KS@文章@KE@ $\boxed{\text{文章}}$

◎添字の取り扱い。

上付文字は"上付文字列&で指定する。また下付文字は_下付文字列&で指定する。

例 $A^2 \dots A^"2&$, $a_0 \dots \#a_0&\#$, $x^{i_n} \dots \#X^"I_N&\&\#$

最後の例の&&の&はダミー文字である。&&と描くと&という文字が直接出力されるので必ず&を間に置くこと。

天地付文字列は「文章|"天付文字列&」, 「文章|_地付文字列&」で指定する。

例 $\sum_{i=1}^n \dots \text{「}\yen SI \yen | _ \# I=1 \& | \text{」} N \# \&$

$\lim_{x \rightarrow \infty} \dots \text{「}\# LIM | _ X \yen 40 \yen \yen 51 \yen \# \&$

◎分数の取り扱い。

分数は@OS @分子@OV @分母@OE @と指定する。

例 $x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$ …… # X = @OS @ -B ¥ 02 ¥ @RT @B " 2 & -4AC @RE @
@OV @2A @OE @

分数の入れ子は3重までである。なお、これらの使用例を図22に示した。

$$y_i = x_i^2 + e^{-x_i} \quad \# ! Y _ I \& \$ = \$ X _ I \& " 2 \& + E " - X _ I \& \& \& ! \#$$

$$\sum_{i=1}^n \lim_{x \rightarrow \infty} \quad @AL @ \# ! \text{ ¥ } S I \text{ ¥ } | _ I = 1 \& | " N \& \text{ ¥ } ! \quad \text{ ¥ } L I M | _ I X \text{ ¥ } 4 0 \text{ ¥ } 5 1 \text{ ¥ } \& \text{ ¥ } ! \#$$

$$D = \sqrt{B^2 - 4AC} \quad @AL @ D = @RT @ B " 2 \& - 4AC @RE @$$

ENTER @AL @ @KS @ENTER @KE @

$$\frac{1 + \frac{1}{A}}{1 - \frac{1}{\sqrt{B}}} \quad @AL @ @OS @ 1 + @OS @ 1 @OV @ A @OE @ @OV @ 1 - @OS @ 1 @OV @ @RT @ B @RE @ @OE @ @OE @$$

図22 数式の例

◎マトリックス等の取り扱い。

マトリックスや場合分けを行う場合は@MT @文章@ME @で指定する。出力文書は1行内に入らなければならない。なお、マトリックス等を描く括弧は@BX_n @要素_{1,1} : 要素_{1,2} : ……要素_{1,n} ; 要素_{2,1} : ……要素_{n,m} @BE_m @と指定する。ただし、入れ子は許されない。なお、nとmは表3の各種括弧と同じである。使用例を図23に示した。実例の@MF=LL@等は列の出力制御であり、Lは左詰め、Cは中央、Rは右詰めの指定であり、列の個数だけ定義する。

4.7 文書の取り込み

文章中に図形やデータセットから別の文章を取り込むことができる。

◎文章の取込み

文章を出力するプログラムにTXTLIB文でデータセット名を指定して、文章中に次の制御子を置く。

@RD=メンバ名@ [COL=N] [CONV OFF]

◎図形の取込み

文章を出力するプログラムにSOCLIB文およびZHNLIB文でデータセット名を指定して、文章

$$Y = A + B \begin{vmatrix} A_{1,1} & A_{1,2} & A_{1,3} \\ A_{2,1} & A_{2,2} & A_{2,3} \\ A_{3,1} & A_{3,2} & A_{3,3} \end{vmatrix} \quad (1-a)$$

$$\begin{cases} a'_{k,j} = a_{k,j} / a_{k,k} & j=1,2,\dots,n \\ a'_{i,j} = a_{i,j} - a'_{k,j} \cdot a_{i,k} & k \neq i ((j=1,\dots,n), i=1,\dots,n) \end{cases} \quad (1-b)$$

```
@PR=2,2@
@MF=CCC@MT@Y$=$A$+$B@BX8@A_1,1&:A_1,2&:A_1,3&
A_2,1&:A_2,2&:A_2,3&;A_3,1&:A_3,2&:A_3,3&@BE8@
@RC@(1-#A#)@ME@
@AT=2@MF=LL@MT@BX4@#A'_K,J&=A_K,J&/A_K,K&:J=1,2,\dots,N;
```

図23 マトリックスの例

中に次の制御子を置く。

@SK= [行指定], [カラム指定], [タイトル指定], [図指定] @

行指定 (A, T, B, I) 行数を指定する。Aは任意の位置に図を挿入する。Tは次の段の先頭に図を挿入する。Bは段の最後に図を挿入する。Iは図が挿入されるまで以後の文章を出力しない。A, T, B, I を省略した場合はAとなる。行数は図とタイトルが入る数を指定する。省略した場合は段の行数となる。

カラム指定 (L, R)カラム数を指定する。Lは左に、Rは右に図を挿入する。省略した場合はLとなる。カラム数は図が入る文字数で指定する。省略した場合は段の文字数となる。

タイトル指定 (H, B) '文章' を指定する。Hはタイトルを図の上側に挿入する。Bはタイトルを図の下側に挿入する。省略した場合はBとなる。

図指定 (S, Z)=メンバ名を指定する。Sは図がソースプログラムの場合である。Zは図が図版(イメージ)の場合である。図指定が省略された場合は内部空白となる。

その他文字の字種(大文字, ローマン)を初期化するには@CI@を指定する。

4.8 文書出力禁止式の設定

文章出力プログラム中に次の文を置くと文書出力禁止域を設定できる。

DFIELD 位置 大きさ

ただし、ページが変わったとき解除される。DFIELD文は8個まで可能である。

5. プログラム実行制御文

図形出力を行うために、その図形出力文を繰り返したり、流れを制御したり、データを入力する文がある。ここで、それらについて説明する。

5.1 SET文とGET文

◎代入文

変数に数値や文字を代入するSET文は次のように書く。

SET {変数名=数式, &文字変数名='文字列'} の並び

このシステムで取り扱う変数はすべて実数とする。ただし!頭文字として!記号のついたものは、IF文や文字列変換の場合に限り整数化される。なお、変数名は4文字以内である。演算子には+, -, *, //, **があり、また関数としてMOD, ATAN2, SIGN, NINT, SQRT, EXP, LOG, SIN, COS, TAN, ABS, RND, LOG2, LOG10 が使用できる。

◎入力文

変数に数値や文字列を端末やデータセットから入力するGET文は次のように書く。

・端末から入力する場合

GET [@] { 変数名, &文字変数名, REP (n1, n2, [n3])
({ 配列名, &文字配列名 } の並び) }

・データセットから入力する場合

OPENn データセット名

GETn [@] { 変数名, &文字変数名, REP (n1, n2, [n3])
({ 配列名, &文字配列名 } の並び) }

CLOSEn

nは1から5まで良く、それぞれ対応して使用する。

最初の@は入力行のデータ数と入力並びの数が同一でなく、入力行のデータを順次解釈する場合であり、@がなければGET文が実行される毎に新しい行が入力される。入力カラムを指定するには、変数名の後に¥(n1, n2)と指定する。n1は開始カラム。n2は終了カラムである。文字列入力データ中に空白やコンマを含む場合は'で囲む。

☒ 入力並びの数が入力データの数より多い場合(@指定のないとき)やデータセットが終了した場合は、変数には-30000、文字変数には*が代入される。

5.2 DO文とSHIFT文

繰り返し実行する文にはDO文とSHIFT文がある。

◎繰り返し文

DO 変数名=n1,n2 [,n3]

.....

ENDDO

n1,n2,n3 は数式が可能でn3が1の場合は省略できる。

◎図形移動文

SHIFT i=n x,y

.....

ENDSHIFT

i は変数名で、nは繰り返し数であり、繰り返しの途中回数が変数iに与えられる。x,y は図形に対する移動量であり、数式で与える。SHIFT文からENDSHIFTまでの図形はX軸方向およびY軸方向に対して、各々x(i-1)とy(i-1)移動した位置に描かれる。同じような絵を同一ページに何枚も描くとき便利である。

5.3 IF文とGOTO文

◎IF文

IF文は次のように書く。

IF 論理式 THEN

.....

[ELSE]

.....

ENDIF

または、次の一行式

IF 論理式 [THEN または ELSE] 文

論理式は、 l_1 (=, \neq , >, \geq , <, \leq) l_2 と書き、 l_1 や l_2 に数式か '文字列' を指定する。また、論理式 (| , &) 論理式 と書くこともでき、| は OR を、& は AND を表す。

◎GOTO文

GOTO文はGOTO ラベル名 で指定する。

5.4 文の解釈

このシステムでは、文の実行前に文の翻訳を行う。すなわち、次の文字変換機能を行う。

◎添字付文字列の使用法

文中に次のような添字付文字列を書く。

&文字列 (数式)

たとえば次のように

& P (I) : DUMMY & ¥ P (I - 1) . R > R ' 文字列変数 '

と書いた場合に I=2 のとき、次のように翻訳される。

P02 : DUMMY ¥ P01 . R > R ' 文字列変数 '

ただし、P02 や P01 が文字変数であれば、さらにその値が代入される。すなわち、P01 = ' A ' , P02 = ' B ' という値であるとき、上の文は次のように翻訳される。

B : DUMMY ¥ A . R > R ' 文字列変数 '

◎文字変数の代入

文字変数を文中に代入するには次のように書く。

&文字変数名.

たとえば、次のように

PROC X/Y ' & A . ' & PR .

と書いた場合に A = ' 文字変数 ' , PR = ' : ' のとき、次のように翻訳される。

PROC X/Y ' 文字変数 ' :

◎数値文字変換

数値を文字に変換する文は次のように書く。

&@数式 (欄記述子)

欄記述子は文字列作画のときの変換と同じである。

たとえば、次のように

IF ' A & @ I (Z I 2) ' = ' A 1 ' THEN

と書いた場合に I=2 のとき、次のように翻訳される。

IF ' A 2 ' = ' A 1 ' THEN

おわりに

簡単ではあるが、主として文法について述べてきた。実際に使用する場合は、身近にいる DOGRACE の利用者のプログラムを複写し、この解説を照らしながら修正し、まず実行してみることを薦める。なお、利用方法としては、次のものがある。◎論文等の文書処理 ◎グラフの作成 ◎表の作成 ◎フローチャートの作成

これらのものをよく手書きで作成している人は、一度 DOGRACE に挑戦することを期待する。