九州大学学術情報リポジトリ
Kyushu University Institutional Repository

[003]速報 : 九州大学大型計算機センター

https://doi.org/10.15017/1516233

出版情報：九州大学大型計算機センター速報. 3, pp.1-15, 1969. 九州大学大型計算機センター
バージョン：
権利関係：

速　　報

No. 3

九州大学大型計算機センター

○$CONV,$NOCONV カードについて

FACOM 230－60計算機システムでは内部コードとして、すべてEBSDICコード系（IBM029 EL型穿孔機のコード）で処理しております。そのためHコード系（IBM026，IBM029H型穿孔機）で穿孔したカードで計算依頼をするためにはコード変換のコントロールカード（$CONV カード）を挿入しなければなりません。

EBSDICコード系とHコード系との異なるコードは次のものです。

Hコード		$CONVを挿入しないと	$CONVを挿入すると
＝	等号 ⟶	#	≠
(	右カッコ ⟶	%	(
)	左カッコ ⟶	＜	)
(8－4)	⟶	@	▼
＋	加算記号 ⟶	&	＋

Hコードで上記の6個の記号以外はELコードと同じです。（利用の手引64ページ参照）

Hコードで穿孔した上記のコードを使用していますと計算機の中では各々ELコードの対応する記号として取扱われ、エラーが出てしまいますので、Hコードで穿孔したカードデックの前には、$CONVカードを挿入しなければならないのは当然ですが特にEL型コードで穿孔したカードデックと混入している場合などは$CONVと$NOCONVを多数挿入しなければならず、面倒ですのでEL型コードで穿孔したカードデックを$CONVを入れたまま処理するとどのようなことが起るかということを考えてみましょう。結論から書きますと、FORTRANについては全く問題はありません。ALGOLについては論理式中の＆という記号が＋となるため＆を使用していない限り問題はないわけです。つまり、EL型で穿孔し$CONV

を挿入したときは、＆（ALGOLのANĐ記号）が＋と変換されるだけというわけです。正しく$CONV, $NOCONVカードを挿入したときと、$CONV のみしか挿入しなかつたときの差は処理時間が少々（CPU時間は変りません）長くなるだけです。

以上のことから利用者の皆様はALGOLで＆を使用している場合のほか（特にFORTRANにおいては）受付番号コードの次に$CONVを挿入しさえすれば$CONV, $NOCONVカードのTROUBLEを回避することができます。（新しくコード切替文を挿入することが望ましいのは言うまでもありませんが。）

○ALGŌLのĐYNAMIC ARRAYについて

コンパイル中にその大きさが定つていないよう配列(dynamic array)を使用する場合には、次のことに注意してください。

'ARRAY A(|1:N|), B(|1:N|);

と宣言しますと、コンパイル時には、Nは未定ですから記憶装置に余分の領域が確保されます。九大センターとしては標準として、256WŌRĐの領域がとられていますので、使用しているすべてのdynamic arrayの領域が256WŌRĐS を越える場合、制御データで、WŌRĐ数をWŌRK:で指定してください。

（ALGŌL 解説編 79ページ）

たとえば

'ARRAY A(|1:L|), B(|1:M|), C(|1:N|);

と宣言されていて、実行中には L＝330,

M＝250,

N＝500,

のようになつていれば、制御データで

WŌRK:＝1100;

と指定しておく必要があります。

○電話によるプログラム相談について

電話によるプログラム相談は適切な指導がうけられないことが多くなりますので、出来るだけ直接センタープログラム相談室で指導をうけてください。

また遠隔地の方でそれが不可能な方は、リスト・デック・利用の手びきマニュアル等、相談に必要なものをそろえて電話をかけてください。

○講習会について

大型計算機センターでは次のように利用者講習会を開きます。受講希望の方は、初級講習会は各連絡所に、中級講習会は地区協に申込んでください。

(1) 初級　FŌRTRAN

開催期日、連絡所名は次の通りです。内容は、センター利用の手続きとFŌRTRAN文法，演習を予定していますが、連絡所の希望によつてはその内容をALGŌLと変更することもありますので連絡所に問合わせてください。会場，時間割その他についても連絡所に問合わけてください。

6月	27(金)・28(土)	佐賀大学
7月	1(火)・2(水)	九州工業大学
	7(月)・8(火)	広島大学
	9(水)・10(木)	山口大学
	11(金)・12(土)	長崎大学
	14(月)・15(火)	熊本大学
	21(月)・22(火)	大分大学
	28(月)・29(火)	宮崎大学
	30(水)・31(木)	鹿児島大学

(2) 中級　FŌRTRAN

7月　17(木)・18(金)　　九州大学

(3) 中級　ALGŌL

7月　24(木)・25(金)　　九州大学

○FORTRANに関して

今回FŌRTRANコンパイラーのレベルアップが行なわれ、今まで障害となつていたものがかなり修正されましたので、以下にお知らせします。又、新しく制限が加えられたもの、現在まだ修正されずに制限されているもの等ありますので、加えてお知らせします。

(I) 今回のレベルアップの結果修正されたもの

(1) カードパンチについて

WRITE(7,f)を使つた場合、LP上に、カードパンチの内容が印字されたり、LPの内容を、消したりしている事がありましたが、今回修正されてよくなりました。

(2) SUBTRACE文について

SUBTRACE文を使つた場合、出力がおかしい事がありましたが修正されました。

(3) ŌPTIŌN NŌLISTがきいていなかつたのが修正されました。

(4) 実行時のエラー個数による打ち切り

今まで、I／Ōに関するエラーをカウントしていなかつたのを修正した結果、指定した個数(普通は50回)になつた時その計算を打ち切る様になりました。

(5) 基本外部関数　ERF, ERFCが使える様になりました。

(6) Đタイプ

(i) ĐO の拡張範囲が使える様になりました。

(ii) Logical IF で implied ĐO を使うとエラーとなつていたものが修正されました。

(例えば IF(I.EQ, 1)WRITE(6, 100)(A(I), I=1, 10))

(II) 新たに制限されたもの

(1) SUBCHK 文

現在I／O リストに現われた配列についてSUBCHK文を使用することはできません。

(※チエックしようとする配列を、入出力させたい場合には、他の配列に置きかえて入出力させて下さい。)

尚、次回のレベルアップで、入出力に関する配列は、チエックを行なわないように修正される予定です。

(2) 特殊型宣言文(TYPE文)で宣言された変数名は、I／O リストには使用できません。

(※この場合は、EQUIVALENCE 文を使つてデータの入出力を行なつて下さい。)

(3) ĐATA 文で、配列の要素に値を与える場合

添字は、ĐATA文の中で値が決まらなければなりません。

例えば

ĐATA((A(I, J), I=1, 2), J=1, 2)／4＊0.0／

は良いが

J＝1

ĐATA(A(I, J), I=1, 2)／2＊0.0／

という使い方は、できません。

(4) NAMELIST 付き REAĐ 文でデータを与える場合で、変数とデータ

の型が異なる時の処理について

解説編(Ⅱ) P28，文法書 P65 の中で、

「代入文の法則に従う」とありますが、次の様になります。

変数＼データ	I	R	Đ	C	ĐC	L	H
I	◎	○	○	×	×	△	×
R	○	◎	○	×	×	△	×
Đ	○	○	◎	×	×	△	×
C	×	×	×	◎	○	×	×
ĐC	×	×	×	○	◎	×	×
L	●	●	●	×	×	◎	×

I：integer
R：real
Đ：double precision
C：complex
ĐC：double precision complex
L：logical
H：hollerish

(注)
- ◎：正常な値が set される。
- ○：変数の型に変換される。
- △：どんな値が set されるか不明。
- ● ：integer の－1以外は false の値が set される。
- ×：許されない（エラーとなつて、計算を打ち切る）

(Ⅲ) 変更されたもの

(1) FORTRAN 実行時の完了コードが次の様に変更されました。

正常終了，モニタによる打ち切りの場合は、以前と同様それぞれ、完了コード000，511 ですが、何らかのエラーがある場合は次の様になります。

エラーレベル	完了コード
W	464
X	474
Z	494

(2) エラーメッセージのいくつかが、変更になりました。

(i) FT187W ĐŌ NŌ KAKUTYŌŌ HANI GA ARU

は

FT 187W m 'ĐŌ' NŌ KAKUTYŌŌ HANI MATAWA TŌBIKŌMI GA ARU n

ただし m；Internal Statoment Number

n；飛び込み先の文番号

となりました。このエラーメッセージはĐŌ ループの外から中に飛びこむ命令がある場合に出ます。

(ii) FT 122W 'MAIN／SUBRŌUTINE／FUNCTIŌN PRŌG. UNIT' GA HUKANZEN ĐE ARU

は

FT 122W 'STŌP／RETURN' BUN GA NAI

となりました。

(iii) 今まで 'C−ERRŌR'

or 'PRŌGRAM−ERRŌR'

と出ていたものが

'PRŌGRAM GA ŌKASII'

という、エラーメッセージに変更になりました。

(3) ĐATA 文で文字定数の割りつけ方法が変更になりました。

(i) ĐATA A(1), A(2)／4HABCĐ, 12HEFGHIJKLMNŌP／

とした場合

A(1)には ABCĐ

A(2) 〃 EFGH

A(3) 〃 IJKL

A(4) 〃 MNŌP

が入ります。　（A(5)は不定）

(ii)　ÐATA　A／20HABCÐEFGHIJKLMNŌPQRST／

とした場合

A(1)には　　ABCÐ

A(2)　〃　　EFGH

A(3)　〃　　IJKL

A(4)　〃　　MNŌP

A(5)　〃　　QRST

が入ります。

以上が修正された主な内容です。

この他に、障害及び制限事項として残されているもの及びプログラミング上の注意事項等を次にあげます。

(1)　以下の機能は、制限されていて使用できません。

(i)　BUFFER文

(ii)　入出力動作状態検査文

(iii)　基本サブルーチン　WAIT

(iv)　大記憶に対する　ENÐFILE文

(v)　Binary　（書式なし形式）でWriteされたI／Ō　に対する

BACK　SPACE文

(2)　Ðタイプで論理型の文関数は、使えません。（障害）

コンパイル中に

ÐATA　YŌMIÐASI　SINGAI

となつて、コンパイルを中止してしまいます。

(3)　MAIN　の第1ステートメントが継続行になつている（i，e，第6column に何か文字がある）場合も、

ĐATA YŌMIĐASI SINGAI

となつてコンパイルを中止しますので御注意下さい。

(4) STŌP n について

nが記号列の場合、記号列中に等号があるとSTŌP 文とみなされずレベル2のエラーとなつてそのジョブは実行されませんので御注意下さい。（障害）

(5) ソースプログラムの中にブランクカードが入つていると、レベル2のエラーとなつてそのジョブは実行されませんので御注意下さい。

(6) Đタイプで、I／Ōリストに implied ĐŌ を使用する場合

添字に式を書くことができません。（障害）

例えば

WRITE（6, 100）（A（I+K）, I=1, N）

（WRITE（6, 100）（A(I), I=1, N）はできます。）

(7) ĐATA 文で、8進定数あるいは、2進定数を与える場合

例えば、8進で12桁をこえる定数を与えると、倍精度の定数とみなされます。（REAĐ 文で与える場合と処置が異なります。）

例えば、Aが実数型変数名の場合

ĐATA A／Ō1234567012345 67／

とすると、Aには、8進で、0000000000123

という値が入ります。（4567012345 67は入りません）

この点御注意下さい。

（又、ĐATA 文は、コンパイル時に、変数に初期値を与えるためのものですから、同じ変数に、ĐATA 文で2度、定数を与えることはできません。）

(8) 整数型変数に文番号を代入する場合

配列要素に文番号を代表する事はできません。

FT 190X SIKI NI AYAMARI GA ARU

というエラーメッセージが出ます。

(9) SUBTRACE文について

SUBTRACE文を使用した場合、引数の値は、そのサブルーチン（又は、関数）が引用された時点での値が出力されます。

サブルーチンの中で、値が与えられる引数については、結果の値が出力される訳ではありませんので御注意下さい。

(10) FT999のエラーメッセージについて

FT999のエラーメッセージは次の5通りがあります。

(i) FT999 W F−ŌVER FLŌW

(ii) FT999 W ĐIVIĐE CHECK ERRŌR

(iii) FT999 Z NŌ INSTRUCTIŌN

(iv) FT999 Z INVALIĐ INSTRUCTIŌN

(v) FT999 Z NŌ MEMŌRY

i）は演算でオーバーフローがおこつたという意味で、ii）は、除算で分母が0になつたという意味です。(iii)～(v)のエラーの原因には、次の3通りが考えられます。

(a) 配列で宣言した範囲をこえて使用した場合

(b) 未定義の記号を使つた場合

(c) サブルーチン又は、関数で引数の使い方に誤りがある場合

これらのエラーが起つた場合は、もう一度プログラムを検討しなおして下さい。

○CPU時間打切り時の出力結果について

FACŌM 230−60 SYSTEM ではラインプリンターに出力するとき大記憶（Ðisk 又はDisk Pack）をBuffer として使用しております。利用者のプログラムでWRITE, PRINT等の命令を出しますとその結果が15桁にまとまつたときに大記憶に転送し、すべての計算が終つた後、大記憶からラインプリンターに打出すわけです。このため、CPU 時間切れで計算が途中で打切られたときは計算結果の内15桁に満たない端数は大記憶に転送されないということが起ります。利用者の皆様の中にはCPU 時間切れの結果の時には、計算をしているはずなのに結果が全然出ていない場合があつて不審に思われたことがあつたのではないでしようか。モニターに強制的に打切られたときのみ、このようなことが起ります。将来修正される予定ですが、当分の間は予定出力量、計算時間を考慮の上計算依頼の際JŌB 種別等を決定してください。

○SSL（科学用サブルーチンライブラリ）の呼び出し法について

(1) 使用方法について

現在、SSL はコンパイルされて、センター側ファイル（デイスクパック）に登録されていますので、いつでも、ソースプログラムの中から、呼び出すだけで使う事ができます。

(i) 例えば、FŌRTRAN では

```
¥NŌ
¥QJŌB
¥FŌRTRAN
      .....
      CALL  CELI1(K, R, EPS)   } ソースプログラム
      .....
      ENÐ
```

¥LIEÐRUN

（データ）

¥JENÐ

(ii) ALGŌLでは、プログラムの前の制御データで、

EXTERNAL := 'PRŌCEÐURE <呼び出し名>;

と宣言する必要があります。

例えば

¥NŌ

¥QJŌB

¥ALGŌL

EXTERNAL:= 'PRŌCEÐURE ACELI1;

'BEGIN

.....

ACELI1(K, CI1, EPS);

.....

'ENÐ

} ソースプログラム

¥LIEÐRUN1

（データ）

¥JENÐ

この様に、カードデックを組み込む必要は全くありません。

(iii) 資料について

SSL に関しては、FŌRTRAN, ALGŌL の解説編にその説明がありますが、呼び出し名、引数等、実際とは、かなり異なつている部分があります。

使用される場合は、プログラム相談室にSSL 使用方法解説書（仮

仕様書）及びソースリストがありますので、必ずそれらを参照して下さい。（学外の方は、センターに問合わせて下さい。）解説編だけに従つて書かれると、エラーとなつて実行されなかつたり、結果がおかしくなつたりしますので御注意下さい。

尚、富士通より、「SSL 使用方法解説書 FORTRAN 編」が発行されています。入手方法は他のマニュアル類と同様の手続きをとつて下さい。

ALGOL編も近く発行される予定です。

SSL に関して、一部不備な点があり、御迷惑をおかけしている事と思います。早急に改善する様に今後共努力いたしますが、お気付きの点がありましたら、センターまで御連絡下さる様お願いいたします。

○ハードウエアの改造について

大型計算機センターでは、去る6月7日にハードウエアの改造を実施いたしました。その結果、浮動小数点演算（real, double）が従来、切り捨てになつていたものが、round（0捨1入）に改良されました。

FACOM 230-60の浮動小数点のデータは、単精度の場合符号部1ビット、指数部9ビット、仮数部26ビットで表現されていますが、演算そのものは仮数部35ビットで行なわれ、結果は26ビットとられることになつています。

改造前は、27ビット以下を無条件に切り捨てていましたが、改造後は27ビット目が0であれば切り捨て、27ビット目が1であれば、26ビット目に1を加えるようになりました。

この結果、演算時間は多少増加しましたが、精度もかなりあがりました。特に同じデータに対して、何度も繰り返し演算を行なう場合にはかなりの

影響が出てきます。

例えば $\Sigma^{1}/_{N(N-1)}$ と $^{N}/_{N+1}$ をN＝５００，１０００の場合に比較してみますと、次のようになりました。

	改　造　前	改　造　後	
$\Sigma^{1}/_{N(N-1)}$	0.99800056	0.99800400	N＝ 500
	0.99899386	0.99900113	N＝1000
$^{N}/_{N+1}$	0.99800400	0.99800400	N＝ 500
	0.99900100	0.99900100	N＝1'000

また、この改造によつて、$^{1}/_{A}$ ＊Aが０．９９９９…… となつていたものが正確に１．０になります。

この改造により、計算結果（精度）が多少変わりますが、改造後で計算結果が大巾に異るプログラムに関しては計算方式そのものに問題があると思われますので再検討してください。

○計算結果等の運搬について

穿孔、計算依頼および結果等の運搬を受付（箱崎地区）とセンター（薬院）間に下記のとおり運搬車を運行しております。

記

	センター発	受付着	受付発	センター着
第１便	9.時20分	9.時50分	10.時10分	10.時40分
第２便	11.時00分	11.時30分	12.時30分	13.時00分
第３便	13.時30分	14.時00分	14.時20分	14.時50分
第４便	15.時10分	15.時40分	16.時00分	16.時30分

ただし土曜日は第２便まで、日曜、祭日運休

○再計算について

利用者は再計算の依頼をしようとする場合は、センター受付で再計算依頼書に必要事項記入の上、次の要領で申込むこと。

イ）デック、計算結果（ＬＰ用紙）を添えて提出すること。

（カードは原則的に入替してはいけない。）

ロ）ジョブ受付番号は前回の番号を使用する。

再計算の場合、明らかな操作ミス、システムエラーである場合を除き、その他はプログラム相談員に一応相談すること。

○保守日について

センターでは、Ｆ２３０－６０の定期保守を毎週木曜日の午前中行っていますので、木曜日午前中は計算業務は行いませんのでお含置き下さい。