The University of Osaka
Institutional Knowledge Archive

Title	プログラム・ノート　大阪大学大型計算機センターニュース　No.1
Author(s)	
Citation	大阪大学大型計算機センターニュース. 1968, 1, p. 31-50
Version Type	VoR
URL	https://hdl.handle.net/11094/65106
rights	
Note	

The University of Osaka Institutional Knowledge Archive : OUKA

https://ir.library.osaka-u.ac.jp/

The University of Osaka

◎ プログラム・ノート

1. 科学計算用 FORTRAN ライブラリー一覧表（NEC 提供）

注 1. センターでのチェックは完了していないから使用に際しては各自よく検討されたい。
2. リストおよびカード・デックはセンターに準備している。
3. ドラム用ライブラリーを除外している。

No.	ライブラリー名	内	容
I	線型計算		
1.	MINVSP	逆行列（I）	掃出法により逆行列を求める。
2.	MINVSR	逆行列（II）	階数減少法で逆行列を求める。
3.	MINVSD	逆行列（III）	分割法により逆行列を求める。
4.	MLNEQ 1	連立一次方程式（I）	AX=B（A, B, Xは行列でA：n×n, B：n×1, X：n×1）なる連立一次方程式を Gauss-Seidel の方法で解く。
5.	MLNEQ 2	連立一次方程式（II）	連立一次方程式 BX=C を消去法で解く。
6.	MLNEQ 3	連立一次方程式（III）	AX=B なる連立方程式を三角行列を利用して消去法で解く。（磁気テープを用いる。）
7.	MDETRM	行列式	掃出法で，行列式の値を求める。
8.	MARTH	行列要素の四則演算	行列の各要素ごとの加，減，乗，除，を行なう。
9.	MSCLR	スカラーと行列の乗算	(スカラー)×(行列)
10.	MPRDCT	行列の乗算	(行列)×(行列)（A×B→C）
11.	MEIGNJ	固有値，固有ベクトル	ヤコビの方法により全ての固有値および固有ベクトルを求める。
12.	MEIGNP	固有値，固有ベクトル	累乗法により，絶対値最大の固有値と，それに対応する固有ベクトルを求める。
II	高次代数方程式		
1.	ALGEQ 1	実係数一元高次代数方程式（I）	Bairstow 法によって実係数一元高次方程式の根を求める。
2.	ALGEQ 2	実係数一元高次代数方程式（II）	Bairstow 法の変形である McAuley 法によって実係数，一元高次方程式の根を求める。
III	数値積分		
1.	INTEG 1	数値積分（I）	与えられた関数の Simpson 公式による積分
2.	INTEG 2	数値積分（II）	GAUSS 公式により与えられた関数の数値積分を行なう。
3.	INTEG 3	数値積分（III）	Chebyshevの多項式を積分する方法で与えられた関数の数値積分を行なう。
IV	微分方程式		
1.	DIFEQ 1	連立常微分方程式（I）	$y'_i=f_i(x, y_1, \cdots y_n)$, $(i=1, \cdots, n)$ なる常微分方程式を Runge-Kutta-Gill の方法で解く。
2.	DIFEQ 2	連立常微分方程式（II）	$y'_i=f_i(x, y_1, \cdots y_n)$, $(i=1, \cdots, n)$ なる常微分方程式を Milne の方法で解く。
3.	DIFEQ 3	常微分方程式	初期条件 $y_i(x_0)$ で $y'=f(x, y)$ $(i=1, \cdots, n)$ なる常微分方程式を Runge-Kutta 法により解く。

No.	ライブラリー名	内	容
V	特殊関数		
1.	BESIKO	Bessel 関数 $I_0(X)$, $K_0(X)$	Bessel 関数 $I_0(X)$, $K_0(X)$ を求める。 (Ⅰ) $M \leqq 0$ ならば, $K_0(X)$ が求まる。$X>2$ なら漸近展開を使用 $2 \geqq X>0$ なら級数展開を使用 (Ⅱ) $M>0$ ならば, $I_0(X)$ が求まる。 $X>3.75$ なら漸近展開を使用 $3.75 \geqq X \geqq -3.75$ なら級数展開を使用
2.	BESIK 1	Bessel 関数 $I_1(X)$, $K_1(X)$	Bessel 関数 $I_1(X)$, $K_1(X)$ を求める。 (Ⅰ) $M \leqq 0$ ならば $K_1(X)$ が求まる。 $X>2$ なら漸近展開を使用 $2 \geqq X>0$ なら級数展開を使用 (Ⅱ) $M>0$ ならば $I_1(X)$ が求まる。 $X>3.75$ なら漸近展開を使用 $3.75 \geqq X \geqq -3.75$ なら級数展開を使用
Ⅵ	補間・平滑化		
1.	IPOLAT	補　　間	x とその函数値f(x) のテーブルを用いた, Lagrange の方法による補間
2.	SMOOTH	データの平滑	観測誤差をもつ観測値の相続く n 個を順にとって, 2次式を最小自乗法であてはめ, 観測値の平滑化を行なう。
Ⅶ	統　　計		
1.	VARAL 1	分散分析(Ⅰ)	二元配置法による分散分析
2.	VARAL 2	分散分析(Ⅱ)	一元配置法による分散分析
3.	VARAL 3	分散分析(Ⅲ)	乱塊法による分散分析
4.	VARAL 4	分散分析(Ⅳ)	ラテン方格法による分散分析
5.	VARAL 5	分散分析(Ⅴ)	グレコ・ラテン方格法による分散分析
6.	VARAL 6	分散分析(Ⅵ)	つりあい型不完全配備法による分散分析
7.	CORAL 1	相関分析(Ⅰ)	二次元の相関分析に使用出来るように, 相関係数 t 分布, 正規分布および母相関係数の95%信頼区間の上限および下限の値を計算する。
8.	CORAL 2	相関分析(Ⅱ)	相関係数の差の検定のための正規分布の値を出力する。
Ⅷ	そ の 他		
1.	RANDM 1	一様乱数	乗算型合同法により一様乱数を発生する。(Lehmer の方法)
2.	RANDM 2	正規乱数	一様乱数と一定の数表によって正規乱数を発生する。
3.	PLOT	プロッティング	与えられた条件の下で, 曲線を近似する点を, LP用紙上に印字するために, その位置を定めるルーチン

2. FORTRAN 比較表—I

機能		NEAC シリーズ 2200 FORTRAN K	HITAC 5020 HARP	IBM 7090/7094 FORTRAN IV	IBM 7090/7094 FORTRAN II	UNIVAC 1107 FORTRAN IV
データ・タイプ	INTEGER	○	○	○	○	○
	REAL	○	○	○	○	○
	COMPLEX	○	○	○	○[A]	○
	LOGICAL	○	○	○	○	○
	DOUBLE LENGTH INTEGER	×	○**	×	×	×
	DOUBLE PRECISION REAL	○	○	○	○[B]	○
	DOUBLE PRECISION COMPLEX	×	○**	×	×	×
	QUADRUPLE PRECISION REAL	×	○**	×	×	×
	数式ステートメント	○	○	○	○	○
	論理式ステートメント	○	○	○	○[C]	○
	GO TO n	○	○	○	○	○
	GO TO (n_1…, n_r), i	○	○	○	○	○
	IF (e) n_1, n_2, n_3	○	○	○	○	○
	IF (e) S	○	○	○	×	○
	DO n I=m_1, m_2, m_3	○	○	○	○	○
	DO の拡張領域	○	○	×	×	○
	CONTINUE	○	○	○	○	○
	CALL SUB	○	○	○	○	○
	CALL SUB (a_1, …, a_n)	○	○	○	○	○
	STOP	○	○	○	○	○
	STOP n	octal n≦777777	octal n≦177777	octal n≦77777	octal n≦77777	○
	RETURN	○	○	○	○	○
	RETURN i	×	×	○**	×	○**
	RETURN o	×	×	×	×	○**
	GO TO i, (n_1, …, n_n)	○	○	○	○	○
	ASSIGN k TO i	○	○	○	○	○
	PAUSE	○	○	○	○	○
	PAUSE n	octal n≦777777	octal n≦177777	octal n≦77777	octal n≦77777	○

	機　　能	NEAC シリーズ 2200 FORTRAN K	HITAC 5020 HARP	IBM 7090/7094 FORTRAN Ⅳ	IBM 7090/7094 FORTRAN Ⅱ	UNIVAC 1107 FORTRAN Ⅳ
入出力ステートメント	READ (u, f) k	○	○	○	×	○
	WRITE (u, f) k	○	○	○	×	○
	READ (u) k	○	○	○	×	○
	WRITE (u) k	○	○	○	×	○
	READ INPUT TAPE u, f, k	○	○	×	○	○
	WRITE OUTPUT TAPE u, f, k	○	○	×	○	○
	READ TAPE u, k	○	○	×	○	○
	WRITE TAPE u, k	○	○	×	○	○
	READ f, k	○	○	○	○	○
	PRINT f, k	○	○	○	○	○
	PUNCH f, k	○	○	○	○	○
	NAMELIST	×	×	○**	×	×
	READ(u, f, END=n_1, ERR=n_2)	○**	×	×	×	×
	REWIND u	○	○	○	○	○
	BACK SPACE u	○	○	○	○	○
	END FILE u	○	○	○	○	○
	READ DRUM v, k	×	○**	×	○**	×
	WRITE DRUM v, k	×	○**	×	○**	×
	READ DRUM PROCEED v, k	×	○**	×	×	×
	WRITE DRUM PROCEED v, k	×	○**	×	×	×
	CALL DMTEST (i)	×	○**	×	×	×
フォーマット・フィールド指定と改行命令	F	○	○	○	○	○
	E	○	○	○	○	○
	I	○	○	○	○	○
	D	○	○	○	○	○
	G	○	×	×	×	×
	スケール因子 nP	○	○	○	○	○
	〃 nQ	×[E]	○	×	×	×
	O	○	○	○	○	○
	L	○	○	○	×	×
	A	○	○	○	○	○
	H	○	○	○	○	○
	X	○	○	○	○	○
	T	○**	~~○~~	×	×	×
	Character String	○	×	×	×	×
	1H1	○	○	○	○	○
	1H△	○	○	○	○	○
	1H0	○	○	○	○	○
	1H+	×[F]	○	×	○	○
	1H2～9	○	×	×	×	Single Space

	機　　能	NEAC シリーズ 2200 FORTRAN K	HITAC 5020 HARP	IBM 7090/7094 FORTRAN Ⅳ	IBM 7090/7094 FORTRAN Ⅱ	UNIVAC 1107 FORTRAN Ⅳ
スペシフィケーション・ステートメント	DIMENSION A (i), B (i)	○	○	○	○	○
	次　元	3	3	7**	3	7**
	Adjustable array	○	○	○	○	○
	COMMON	○	○	○	○	○
	COMMON/lbl/a, b, c	○	○	○	×	○
	EQUIVARENCE	○	○	○	○	○
	EXTERNAL	○	○	○	×	○
	タイプ ステートメント	○	○	○	×	○
	IMPLICIT	○**	×	×	×	×
	DRUM DIMENSION	×	○**	×	×	×
	DRUM COMMON	×	○**	×	×	×
	データー イニシャリゼーション	○	○	○	×	○
	FREQUENCY	×	×	○**	×	×
	ABNORMAL	×	×	×	×	○**
その他	END	○	○	○	○	○
	END ($i_1 \cdots, i_{15}$)	×	×	×	○**	×
ハードウェア・IF・ステートメント	CALL OVERFL (j)	○	○	○	×	○
	CALL DVCHK (j)	○	○	○	×	○
	CALL SLITE (n)	○	○	○	×	○
	CALL SLITET (n, j)	○	○	○	×	○
	CALL SSWTCH (n, j)	○	○	○	×	○
	IF ACCUMULATOR OVERFLOW n_1, n_2	×	○	×	○	○
	IF QUOTIENT OVERFLOW n_1, n_2	×	○	×	○	○
	IFDIVIDE CHECK n_1, n_2	×	○	×	○	○
	IF (SENSELIGHT i) n_1, n_2	×	○	×	○	○
	IF (SENSE SWITCH) n_1, n_2	×	○	×	○	○
	SENSE LIGHT i	×	○	×	○	○

	機　　能	NEAC シリーズ 2200 FORTRAN K	HITAC 5020 HARP	IBM 7090/7094 FORTRAN Ⅳ	IBM 7090/7094 FORTRAN Ⅱ	UNIVAC 1107 FORTRAN Ⅳ
サブプログラム・ステートメント	ステートメント関数	○	○	○	○	○
	関数サブプログラム	○	○	○	○	○
	Intrinsic 関数	○	○	○	○	○
	Basic external 関数	○	○	○	○	○
	サブルチン・サブプログラム	○	○	○	○	○
	ENTRY ステートメント	×	×	○**	×	×
	BLOCKDATA	○	○	○	×	×
デバッギング・その他のステートメント	CALL DUMP (a, b, fi)	○	×	○	○	×
	CALL PDUMP (a, b, fi)	○	×	○	○	×
	CALL MDUMP(a, b, fi)	○	×	×	×	×
	BEGIN TRACE k	○	×	×	×	×
	END TRACE k	○	×	×	×	×
	BEGIN FLOW	○	×	×	×	×
	END FLOW	○	×	×	×	×
	DEBUG VALUE n_1, n_2, a	×	○	×	×	×
	DEBUG FLOW n_1, n_2	×	○	×	×	×
	DEBUG DUMP n_1, a_1…	×	○	×	×	×
	DEBUG TERMINATE n	×	○	×	×	×
	CALL CLOCK (i)	×D	○	×		×
	CALL CHAIN (R, T)	×	×	×	○**	×
	CALL EXIT	○	×	○	○	×
	CALL REREAD (i)	○				
	CALL RENAME (U, ：symu：)	○				

注）A　カラム 1 nI をパンチすることにより使用可能
　　B　カラム 1 nD をパンチすることにより使用可能
　　C　Boolean 数式ステートメント
　　D　external な組込み関数として使用可能
　　E　昭和44年 3 月より改訂の予定
　　F　昭和43年11月より改訂の予定
　　**　その FORTRAN 特有のもの

3. FŌRTRAN 比較表—II

項目			NEAC シリーズ 2200 FŌRTRAN K	HITAC 5020 HARP	IBM 7090/7094 FŌRTRAN IV
1) エレメント式関係					
使用文字			48文字 0～9，A～Z， △，+，−，*，/，(，) ・，'，=，:，通貨記号	47文字 FŌRTRAN K から，： を除く	47文字 FŌRTRAN K から，： を除く
データーの型			5つのタイプ {整数型 実数型，倍精度実数型 複素数型 論理型	8つのタイプ FŌRTRAN K に加えて {倍長整数型 4倍精度実数型 倍精度複素数型	5つのタイプ FŌRTRAN K と同様
character string*1			可	不可	不可
精度*2	整数		$\leq 2^{23}-1$	$\leq 2^{31}-1$	$\leq 2^{35}-1$
	倍長整数			$\leq 2^{63}-1$	
	実数	仮数	$\leq 2^{35}-1$	$\leq 2^{23}-1$	$\leq 2^{27}-1$
		指数	$-616 \leq e \leq 616$	$-38 \leq e \leq 38$	$-38 \leq e \leq 38$
	倍精度実数	仮数	$\leq 2^{70}-1$	$\leq 2^{55}-1$	$\leq 2^{54}-1$
		指数	$-616 \leq e \leq 616$	$-38 \leq e \leq 38$	$-29 \leq e \leq 38$
	4倍精度実数	仮数		$\leq 2^{119}-1$	
		指数		$-38 \leq e \leq 38$	
混合演算	四則演算 (+，−，*，/)		すべてのタイプと結合可能	Same type {real, double, complex}(OP){real, double, complex}	Same type real (OP) {double, complex}
	巾乗 (**)		all type ** integer {integer, real, double} ** {real, double}	alltype ** integer {real, double} ** {real, double}	HARP と同様
	関係式		{integer, real, double}(ŌP){integer, real, double}	integer (ŌP) integer {real, double}(ŌP){real, double}	HARP と同様

項　　目	NEAC シリーズ 2200 FORTRAN K	HITAC 5020 HARP	IBM 7090/7094 FORTRAN IV
2) Assignmentステートメント関係			
算　術 Assignment ステートメント	すべてのタイプ可能	{integer, real, double} ← {integer, real, double} complex ← {real, complex}	{integer, real, double} ← {integer, real, double} complex ← {real, complex}
3) コントロール・ステートメント関係			
DO コントロール・パラメタとして I±C	不　可	可	不　可
RETURN i ステートメント　＊3	不　可	不　可	可
4) 入出力ステートメント関係			
List directed I/O　＊4	可	不　可	不　可
Data・directed I/O (NAMELIST)　5＊	不　可	不　可	可
エラー・EOF・チェック	可	不　可	不　可
フォーマット　Gw. d	F, E, D, I, L, に適用	不　可	不　可
フォーマット　mQ　＊6	不　可（昭和44年3月より改訂の予定）	可	不　可
フォーマット　To　＊7	可	不　可	不　可
フォーマット　IH＋	不　可（昭和43年11月より改訂の予定）	可	可
5) 宣言ステートメント関係			
配　列	3次元	3次元	7次元
EQUIVALENCE での添字付変数の添字の数	1～n 次　n≦3	1　次	1～n 次　n≦3
DATA イニシャリゼーション	可	不　可	可
WORD LENGTH n BITS　＊8	不　可	可	不　可

項　　目	NEAC シリーズ 2200 FORTRAN K	HITAC 5020 HARP	IBM 7090/7094 FORTRAN IV
6) サブプログラム関係			
intrinsic 関数	付表1 参 照	FORTRANK と比較して DABS, DMAX1, DMIN1, DSIGN が無く，IDFIX が有る	FORTRAN K と同様
basic external 関数	付表2 参 照	FORTRAN K に加えて LOG10, TAN, COSH SINH, CBRT, DEXP10 DTAN, DTANH, DCOSH, DSINH, DCBRT, DCABS, DCEXP, DCLOG, DCSIN, DCCOS, DCSQRT	FORTRAN K と同様
ENTRY ステートメント ＊9	不 可	不 可	可
サブルーチン CALL の引数として，文字型定数	可	不 可	可

ハード・ウェア・チェック サブルーチン 及 び ステートメント	OVERFL (j)
	DVCHK (j)
	SLITE (n)
	SLITET (n, j)
	SSWTCH (n, j)
		FORTRAN K に加えて
		IF ACCUMULATOR OVERFLOW n_1, n_2
		IF QUOTIENT OVERFLOW n_1, n_2
		IF DIVIDE CHECK n_1, n_2
		IF (SENSE LIGHT i) n_1, n_2
		IF (SENSE SWITCH i) n_1, n_2
			FORTRAN K と同様

デバッギング サブルーチン 及 び ステートメント	サブルーチン DUMP, PDUMP, MDUMP ステートメント BEGIN TRACE, END TRACE, BEGIN FLOW, END FLOW	ステートメント DEBUG DUMP DEBUG VALUE DEBUG FLOW DEBUG TERMINATE	サブルーチン DUMP PDUMP
そ の 他	サブルーチン CHAIN (I), REREAD, CLOCK (I), EXIT, RENAME	CLOCK (I) DPON OFSON (I) DPOFF OFSOFF (I)	な し
7) そ の 他			
コンティニュエーション・ライン	19 行	4行 (32K以下) 9行 (32K以上)	19 行
ドラム関係 ステートメント	不 可	可	不 可

付 表 1

Intrinsic 関数

		引数	関数	
1	ABS	R	R	*
2	IABS	I	I	
3	DABS	D	D	*
4	AINT	R	R	*
5	INT	R	I	*
6	IDINT	D	I	*
7	AMŌD	R	R	*
8	MŌD	I	I	*
9	AMIN 0	$I_1 \cdots I_n$	R	
10	AMIN 1	$R_1 \cdots R_n$	R	*
11	MIN 0	$I_1 \cdots I_n$	I	*
12	MIN 1	$R_1 \cdots R_n$	I	
13	DMIN 1	$D_1 \cdots D_n$	D	*
14	AMAX 0	$I_1 \cdots I_n$	R	
15	AMAX 1	$R_1 \cdots R_n$	R	*
16	MAX 0	$I_1 \cdots I_n$	I	*
17	MAX 1	$R_1 \cdots R_n$	I	
18	DMAX 1	$D_1 \cdots D_n$	D	*
19	FLŌAT	I	R	
20	DFLŌAT	I	D	
21	IFIX	R	I	*
22	SIGN	R_1, R_2	R	*
23	ISIGN	I_1, I_2	I	*
24	DSIGN	D_1, D_2	D	*
25	DIM	R_1, R_2	R	*
26	IDIM	I_1, I_2	I	
27	SNGL	D	R	
28	REAL	C	R	*
29	AIMAG	C	R	*
30	DBLE	R	D	*
31	CMPLX	R_1, R_2	C	*
32	CŌNJG	C	C	

付 表 2

Basic external 関数

		引数	関数	
1	EXP	R	R	*
2	DEXP	D	D	
3	CEXP	C	C	
4	ALŌG	R	R	*
5	DLŌG	D	D	
6	CLŌG	C	C	
7	ALŌG10	R	R	*
8	DLŌG10	D	D	
9	SIN	R	R	*
10	DSIN	D	D	
11	CSIN	C	C	
12	CŌS	R	R	*
13	DCŌS	D	D	
14	CCŌS	C	C	
15	TANH	R	R	*
16	CABS	C	R	
17	SQRT	R	R	*
18	DSQRT	D	D	
19	CSQRT	C	C	
20	ATAN	R	R	*
21	DATAN	D	D	
22	ATAN 2	R_1, R_2	R	*
23	DATAN 2	D_1, D_2	D	
24	DMŌD	D_1, D_2	D	
25	IAND	I	I	
26	IŌR	I	I	
27	ICŌMPL	I	I	
28	IEXCLR	I	I	

注 1) R；実　　数
D；倍精度実数
C；複 素 数
I；整　　数

注 2) ＊；HARPの方が，引数の型の種類が多いか，あるいはHARPの関数の型と異なる。

注 3) FŌRTRANKではIntrinsic関数とBasic external関数の区別はない。ここではJISに従って分けてある。

＊1.　文字型定数nHX……Xのかわりに：……X：と：で文字列を囲ったもの。

（コーディング・シートに@とかくと，リストには：と印刷される。）

＊2.　2^nと10進数の関係。

n	10　進　数
23	8388608
27	134217728
31	2147483648
35	34359738368
54	18014398509481984
55	36028797018963968
63	9223372036854775808
70	1180591620717411303424
110	1298074214633706907132624082305024

＊3.　CALLの際指定したｉの値によって左からｉ番目のステートメントレベルで示されたステートメントへもどる。

＊4.　FORMAT 指定なしで，リストに書かれた順序に従ってデータの I/O を行なう。

＊5.　データをデータ名と共に出したり，外部媒体上のデータ名で指定の変数にデータを入れる。

＊6.　Field descripter に対するデータに対してのみ使用する。

＊7.　１レコード内で，次のデータが，トップから数えて何桁目から始まるかを示す。

＊8.　実数型定数の１語の長さを指定する。　n＝32，64，128

＊9.　マルチプル・エントリーがサブルーチンの中で可能となる。

4. HITAC 5020 HARP で作られたデックを NEAC にかけるための変換注意事項

ここで述べるものは HITAC 5020 HARP のために作成されたカード・デックを，NEAC のシステムにかける場合に書換を要する点，並びに結果に影響を及ぼす相違点を列挙したものである。しかしこれらの項目は，主に双方のマニュアルの比較により拾い出したものであり，全ての相違点を網羅しているとはいい難い。従ってここには挙がっていない文法上の違い等のために，エラーが生じたり異った結果が出ることは往々にしてあり得る事なので，得られた結果については十分検討する必要がある。(NEAC FŌRTRAN については昭和43年9月現在の FŌRTRAN K に準拠する。)

1) カード・デック

- カードは IBM の H コードでパンチしたものに限られる。
- 全部空白からなるカードは取り除かなければならない。(昭和43年11月より改訂の予定)
- 1桁目に $ マークのあるカードは全て取り除き，改めて NEAC FŌRTRAN の制御カードを挿入しなければならない。(制御カードの付け方については25頁参照。)
- サブプログラムを含むデック編成においては，カードデックの順序として，サブプログラムのデックよりもそれを呼ぶステートメントを含むデックが先に現われなければならない。HARPのようにサブプログラムを前に出す事は許されない。(27頁参照)

2) 宣言文，DATA 文

- 型宣言文で以下のものは許されない。

  WŌRD LENGTH n BITS

  DŌUBLE LENGTH INTEGER

  QUADRUPLE PRECISIŌN

  DŌUBLE PRECISIŌN CŌMPLEX

  NEAC FŌRTRAN ではこれらのデータ型は定義されていないので，定数並びに FUNCTIŌN 文の型としても以上のものは許されない。
- 同じデータ型でも精度は異る。(37頁参照)
- 同一変数名に異った型の宣言文を2重に書くとエラーになる。
- DATA 文により倍精度実数型変数並びに複素数型変数に，文字定数或いは8進数を格納する事はできない。
- DATA 文で論理型変数に8進数を格納すると，左端の2桁のみが格納される。
- DATA 文に8進定数を書く場合，Ō の前に桁数の表示が要る。

  $n\bar{O}d_1d_2\cdots\cdots d_n$　　但し，$1 \leqq n \leqq 16$

  尚，変数に格納できる桁数は

  整数型で　8桁まで

  実数型で　16桁まで

  である。

○ DATA 文により変数に文字型データを格納する場合，
変数に格納できる文字数は

整数型で　4文字まで

実数型で　8文字まで

である。

○ DATA 文中に論理定数を書く場合

T　又は　.T.

F　又は　.F.

の表現を用いてはならない。必ず

. TRUE.

. FALSE.

と書かなければならない。

3) 定　　数

○ 整数型定数は 8388607 ($2^{23}-1$) を越えてはならない。

○ 複素数型定数の括弧の中に倍精度実定数を書いてはいけない。

○ 倍精度実定数はDで始まる指数部のついたものに限られる。プログラムの中にDで始まる指数部がなくて精度が11桁以上ある定数が書いてあると，コンパイラーが自動的に下位の桁を打ち切って1倍精度にする。(WARNING メッセージが出る。)

○ 文字型定数 $nHx_1x_2\cdots\cdots x_n$ で n が8より大きいものは許されない。変数に格納する場合は

整数型で　4文字まで

実数型で　8文字まで

である。

4) 式

○ 取扱われる数値の範囲と精度が異るから実行後の結果は必ずしも同一とはならない。

○ 1倍精度同志の四則演算又はベキ乗算の結果は2倍精度にならない。

○ 文字定数を関係式の中に書いてはいけない。一度変数に代入した上で比較する事。(項目5代入文参照)

○ 一般に，演算結果，代入文でのデータ型の変換，入出力でのデータ型の変換の際にオーバーフローが生じた時の処置は同じではない。

{ エラー・メッセージの有無
{ オーバーフロー後の値

○ 除数が0の除算を行なってもメッセージは出ない。除数が1であるものとみなして除算を行わずに計算を続行する。(整数型でも実数型でも同じ。)

○ a**b で a=0, b≦0 の時の結果は以下のようになる。

a＝0，b＝0ならば　1

a＝0，b〈0ならば　オーバーフロー

（実数型の場合も，これに準ずる。）

何れの場合もメッセージが出る。

5) 代　入　文

○ 代入文の右辺に文字定数を書いた時，左辺の変数に格納できる文字数は

整数型なら　4文字まで

実数型なら　8文字まで

である。

6) DŌ 文

○ 一　般　形

DŌ n i=m_1, m_2, m_3

において m_1, m_2, m_3 が j±c の形であることは許されない。整数型の定数又は添字なし変数だけが許される。

○ DŌ の端末文として，以下のものを含む論理 IF 文は許されない。

GŌ TŌ 文

算術 IF 文

DŌ 文

RETURN 文

STŌP 文

7) ハードウェア IF ステートメント

○ 以下に述べる FŌRTRAN II のステートメントは何れも使用できない。

IF (SENSE LIGHT i) n_1, n_2

IF (SENSE SWITCH i) n_1, n_2

IF ACCUMULATŌR ŌVERFLŌW n_1, n_2

IF QUŌTIENT ŌVERFLŌW n_1, n_2

IF DIVIDE CHECK n_1, n_2

SENSE LIGHT i

8) 入　出　力　文

○ 入出力装置指定のユニット番号が異なる。CALL RENAME 文のカードの挿入により，変更しなければならない。（28頁参照）

○ 入出力リストで DŌ 型くり返しの重なりは3重までしか許されない。

○ DŌ 型並びのくり返し指定

$i=m_1, m_2, m_3$

においてm_1, m_2, m_3には整数型の定数又は添字なし変数のみが許され，j±cの形は書けない。

○ HARP では出力リストのない2進形式の WRITE 文によって一定量の0からなるレコードが作成されるが，NEAC FŌRTRAN の場合，出力リストのない2進形式の WRITE 文によって作成されるレコードには，何が書き込まれるかわからない。従って0からなるレコードを作成する場合にも，出力リストは必ず書かなければならない。

9) FŌRMAT 文

○ 改行指定の制御用文字として

1H+

を用いてはならない。

（昭和43年11月に改訂の予定）

○ nQ の形のベキ因子は使えない。（昭和44年3月に改訂の予定）

○ 99より大きい反復数は許されない。（昭和44年3月より133まで拡張の予定）

○ くり返しの括弧レベルは2まで。但し，一番外側の括弧は含めない。

（昭和44年3月より，レベル3まで拡張の予定）

○ 不定回くり返しの場合，最後の右括弧に出会った場合にのみ前にもどる。この時，書式仕様の内部に括弧があれば，最後から2番目の右括弧に対応する左括弧（反復数も含めて）にもどる。書式仕様の内部に括弧がなければ一番外側の左括弧にもどる。

○ 入力リストが

倍精度実数型

複素数型

論理型

の時，A 欄記述子又は Ō 欄記述子により文字型データ或いは8進数を読み込む事はできない。

○ A 欄記述子により文字型データを変数に入力する場合，入力できる文字数は

整数型で　4文字まで

実数型で　8文字まで

である。

○ Ō 欄記述子により8進数を変数に入力する場合，入力できる桁数は

整数型で　8桁まで

実数型で　16桁まで

である。

○ A 又は Ō 以外の欄記述子による出力で，出力フィールドより出力データの方が長い時は出力フィールドの全体に "*" がプリントされる。

○ Ō 欄記述子 による出力の場合，出力データの長さは常に16桁で

出力フィールド＞16

ならばデータは出力フィールドの左詰にプリントされ

出力フィールド＜16

ならばデータの左端からプリントして右に余ったものは捨てられる。

○ FŌRMAT を配列内に定義する場合，1つの配列要素の格納文字数は

整数型配列の場合　4文字

実数型配列の場合　8文字

である。この点に注意して，必要があれば配列への FŌRMAT 文の格納方法を変えなければならない。

10) 組み込み関数，基本外部関数，システム・サブルーチン

○ 組み込み関数は各々について関数の型と引数の型が一義的に定められている。これについて TRAN はマニュアルにより確かめる必要がある。（40頁参照）

○ HARPの基本外部関数では引数が1倍精度でも関数の型は2倍精度になるが，NEAC FŌRTRAN では引数が1倍精度なら関数型も1倍精度である。

○ 実数型と倍精度実数型で名前の違う組み込み関数或いは基本外部関数は区別して使わなければならない。

ABS	DABS
AMŌD	DMŌD
SIGN	DSIGN
AMAX 1	DMAX 1
AMIN 1	DMIN 1

○ 複素数を実引数とする SNGL 関数は存在しない。特に複素数型の式を実引数として用いる場合，SNGL 関数を用いないように注意する事。

○ 次に挙げる基本外部関数は使えない。

EXP10, TAN, CŌSH, SINH, CBRT

DEXP10, DTAN, DTANH, DCŌSH, DSINH, DCBRT

（2語長整数型に関するものとして）

IDFIX

（倍精度複素数型に関するものとして）

DCABS, DCEXP, DCLŌG, DCSIN, DCCŌS, DCSQRT

（4倍精度実数型に関するものとして）

QEXP, QLŌG, QLŌG10, QATAN, QATAN 2, QSIN, QCŌS, QSQRT

（そ　の　他）

IBTŌD

FŌRTRAN II の関数名

○ 以下のシステム・サブルーチンは使えない。

ŌFSŌN, OFSŌFF, DPŌN, DPŌFF

○ CALL CLŌCK (i)

の結果，i には milli-secōnd 単位の時間が与えられる。

○ CALL CHAIN (R, T)

の形は許されない。

11) サブプログラムの引数に関する注意

○ CALL 文又は関数引用の実引数に文関数名は許されない。

○ 実引数が文字定数の時，対応する仮引数は実数型でなければならない。

○ 仮引数の名前が CŌMMŌN 文の並びに現われてはいけない。この場合は CŌMMŌN 領域を通じてのみ情報の受け渡しをする事。

○ DIMENSIŌN 文に現われた配列名が仮の引数であって，その大きさを整数型変数で指定する場合，それらの整数型変数は仮引数の並びに入っていなければならない。

○ 配列名が仮引数の時，その配列の各次元の長さは仮引数名に対する DIMENSIŌN 文に従い，実引数の配列の次元の長さには関係しない。

○ 実引数が配列要素名で仮引数が1次元の配列名の時，HARP では仮引数配列の第1要素が実引数の1つ後ろの配列要素に対応する。しかし NEAC FŌRTRAN では実引数の配列要素そのものと仮引数配列の第1要素とが対応する。

今，実引数 A(i)，仮引数Xとすると各要素の対応は次のようになる。

HARP	A(i+k) ←——→ X (k)
NEAC FŌRTRAN	A(i+k−1) ←——→ X (k)

k=1, 2, ……

12) BLŌCK DATA SUBPRŌGRAM

○ BLŌCK DATA SUBPRŌGRAM でブランク CŌMMŌN (無名 CŌMMŌN) に初期値を与えてはいけない。

(NEAC FŌRTRAN ではブランク CŌMMŌN に初期値を与える方法は用意されていない。)

○ BLŌCK DATA SUBPRŌGRAM を含むカードデックの編成は30頁に従って作成しなければならない。

13) DRUM に関するステートメント

○ DRUM に関するステートメントは全て使用できない。

READ DRUM v, k

WRITE DRUM v, k

READ DRUM PRŌCEED v, k

WRITE DRUM PRŌCEED v, k

CALL DMTEST (i)

DRUM DIMENSIŌN

DRUM CŌMMŌN

14) デバッグ・ステートメント

○ デバッグ用の文は全て異るので以下のものは使えない。

DEBUG VALUES n_1, n_2, IF (e) a_1, a_2, ……

DEBUG VALUES n_1, n_2, a_1, a_2, ……

DEBUG FLŌW n_1, n_2

DEBUG DUMP n, IF (e) a_1, a_2, ……

DEBUG DUMP n, a_1, a_2, ……

DEBUG TERMINATE n, IF (e)

DEBUG TERMINATE n

(但し NEAC FŌRTRAN には BEGIN TRACE, END TRACE, BEGIN FLŌW, END FLŌW がある。)

15) チェイン・ジョブ

○ NEAC FŌRTRAN でのチェイン・ジョブは HARP におけるサブプログラム単位の overlay の手法とは異り，各チェインごとにメインプログラムを持ち，オブジェクト・プログラムは各チェインごとに完全に入換えられる。但しブランク CŌMMŌN の領域だけはチェイン間で保存されるので，この領域を通じてチェイン間の情報の受け渡しができる。

詳細は28頁参照の事。

参 考 資 料

(1) プログラム・マニュアル HITAC 5020, 5020E/F FŌRTRAN (HARP) (5020—3—005—02) 日立製作所

(2) プログラム書換えの手引 (1) 〔FŌRTRAN-K ←—→ HARP—5020〕(第1版) 東北大学大型計算機センター設置準備本部

(3) NEAC- シリーズ 2200 FŌRTRAN コンパイラK説明書 (EŌI-39807) 日本電気

5. HARP から NEAC FŌRTRAN への変換のためのチェック・リスト

〔これは変換注意事項の要約である。〕

① • カードは IBM の H コード下でパンチしてあるか。

• ~~全部空白のカードは取り除いたか。（昭和43年11月より改訂の予定）~~

• 制御カードは全て差換えたか。

• デック編成の違いを考慮したか。

② • データ型，精度の指定で以下のものを使っていないか。

WŌRD LENGTH n BITS

DŌUBLE LENGTH INTEGER

QUADRUPLE PRECISIŌN

DŌUBLE PRECISIŌN CŌMPLEX

③ • 倍精度実数型，複素数型変数に文字型データ又は8進数を格納しようとしていないか。

• 文字定数，8進数を変数に格納する時，データ型と格納文字数（又は桁数）の対応を考慮したか。

• 8進定数で Ō の前に桁数の表示をつけたか。

• 論理定数にT, F, . T., . F. の表現を用いていないか。

• 倍精度実定数にD指数部を付けたか。

• 整数型定数で 8388607 より大きいものを書いていないか。

• 複素数型定数の括弧の中に，D指数部のついた定数を書き込んでいないか。

• 文字定数を関係式の中に書いていないか。

④ • DŌ 文又は入出力リストの DŌ 型くり返しのパラメータ $i = m_1, m_2, m_3$ において，m_1, m_2, m_3 に $j \pm c$ の形を使っていないか。

• 以下の文を含む論理 IF 文を，DŌ 文の端末文として使っていないか。

GŌ TŌ 文

算術 IF 文

DŌ 文

RETURN 文

STŌP 文

⑤ • CALL RENAME 文の挿入により，入出力装置のユニット指定番号を変えたか。

• 入出力リストの DŌ 型くり返しの重なりが4重以上になっていないか。

• 2進形式の全ての WRITE 文に出力リストが書いてあるか。

⑥ ~~• FŌRMAT 文で改行指定文字に 1H+ を使っていないか。（昭和43年11月より改訂の予定）~~

• nQ の形のベキ因子を使っていないか。（昭和44年3月より改訂の予定）

• 100以上の反復数を使っていないか。（（昭和44年3月より，133まで使用可能になる）

• くり返しの括弧のレベルが3以上になっていないか。（昭和44年3月より，3重まで使用可能になる）

- 不定回くり返しの違いを考慮したか。
- 配列内に FŌRMAT を定義する時，配列内に格納できる文字数の違いを考慮したか。

⑦ • 組み込み関数並びに基本外部関数で関数名と引数の型，関数の型の対応を確かめたか。
- 次に挙げる基本外部関数を使っていないか。

  EXP10, TAN, CŌSH, SINH, CBRT

  DEXP10, DTAN, DTANH, DCŌSH, DSINH

  DCBRT, IDFIX

  引数が倍精度複素数型のもの

  引数が4倍精度実数型のもの

  IBTŌD

  FŌRTRAN II の関数名
- SNGL 関数の引数に複素数を使っていないか。
- 以下のシステム・サブルーチンを使っていないか。

  ŌFSŌN, ŌFSŌFF, DPŌN, DPŌFF
- CALL CLŌCK (i) の結果 i には milli-second 単位の時間が与えられることを考慮したか。

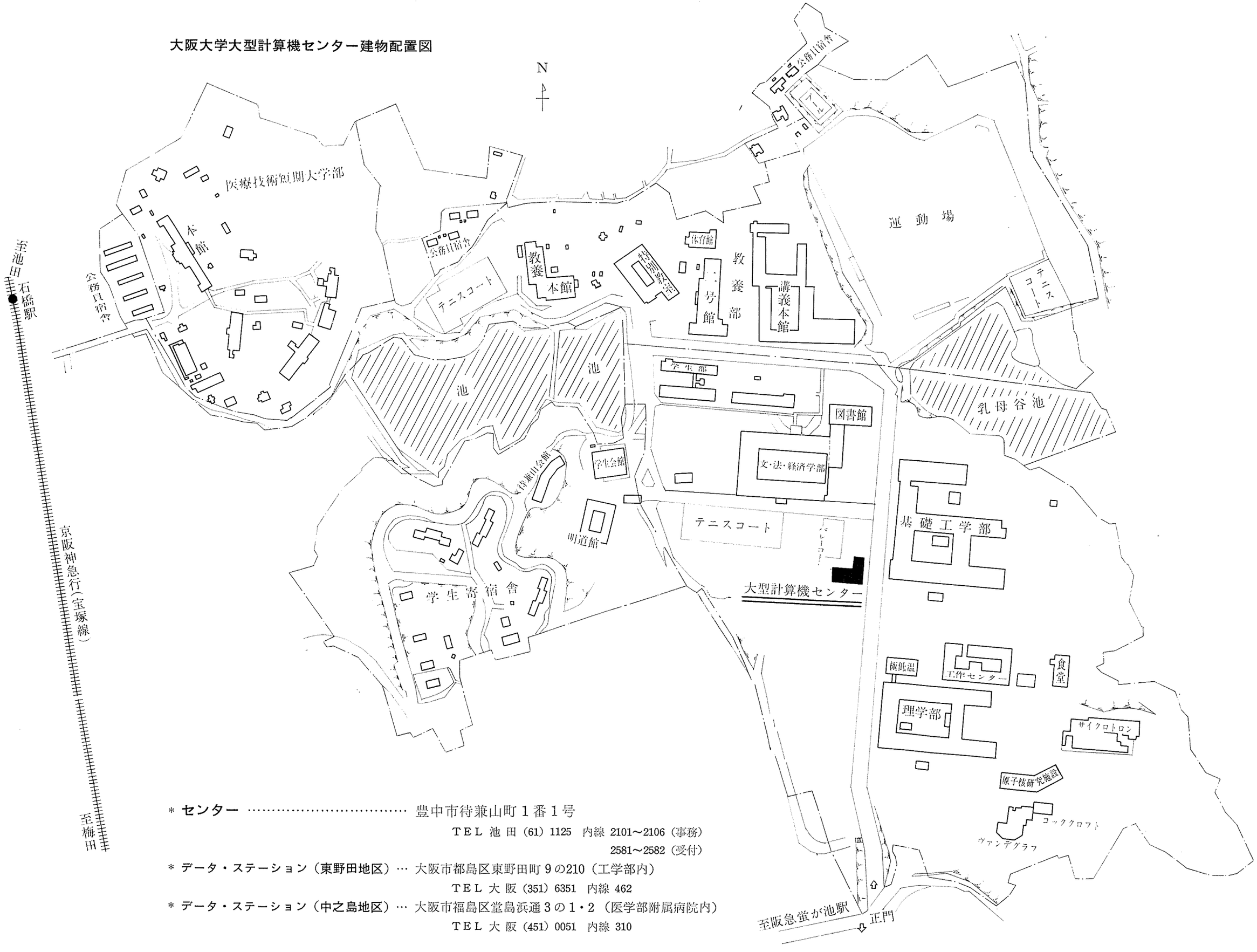
⑧ • 文関数名を実引数に書いていないか。
- 実引数が文字定数の時，対応する仮引数は実数型になっているか。
- 仮引数が CŌMMŌN 文の中に現われていないか。
- 仮引数配列の大きさを指定する整数型変数が，仮引数の並びの中に入っているか。
- 仮引数配列の構造は仮引数配列名に対する DIMENSIŌN 文によることを考慮したか。
- 実引数が配列要素名で仮引数が配列名の時，要素の対応の違いを考慮したか。

⑨ • BLŌCK DATA SUBPRŌGRAM でブランク CŌMMŌN に初期値を与えていないか。

⑩ • FŌRTRAN II のハードウェア IF ステートメントを使っていないか。
- DRUM に関するステートメントを使っていないか。
- HARP のデバッグ・ステートメントは全て除いたか。

⑪ • チェイン・ジョブの方法を変えたか。

大阪大学大型計算機センター建物配置図



＊ **センター** …… 豊中市待兼山町１番１号

TEL 池田 (61) 1125　内線 2101～2106（事務）
2581～2582（受付）

＊ **データ・ステーション（東野田地区）** … 大阪市都島区東野田町９の210（工学部内）

TEL 大阪 (351) 6351　内線 462

＊ **データ・ステーション（中之島地区）** … 大阪市福島区堂島浜通３の１・２（医学部附属病院内）

TEL 大阪 (451) 0051　内線 310