

まあ、お茶でも飲みながらI : 初めてのファイル転送

肥田木, 直子
九州大学大型計算機センターシステム管理掛

<https://doi.org/10.15017/1470274>

出版情報 : 九州大学大型計算機センター広報. 28 (3), pp.232-243, 1995-09. 九州大学大型計算機センター
バージョン :
権利関係 :



まあ、お茶でも飲みながら I

～ 初めてのファイル転送 ～

肥田木 直子 *

プロローグ

これは【必要に迫られて、ワークステーションから仕方なく MSP を使わなくてはならないが、実はよくわからん。】というアナタのために書きました。基本的にパーソナルコンピュータでも動作は同じです。これはあくまでも入門編ですので、こだわりたい方は関係資料や参考文献をじっくり読み、思う存分究めていただきたいと思います。

ひととおりにやってみましょう

仮にアナタが、研究室のボスから『計算結果を明日までにだせ！』と言われたとします。しかし、その計算はデカ過ぎて、研究室にあるワークステーションでは、明らかに実行不可能です。そこでどうするか？

必要なファイルを大型計算機センターに送り込んで、MSP 上で計算をさせれば一応の解決をみるようです。

アナタの愛用ワークステーションに入って、実際にやってみましょう。a79999a 部分はアナタの MSP の登録番号¹で、user% や ftp> はプロンプトを表しています。

```

user% ftp           <--- FTP クライアントの起動
ftp> open 133.5.9.2       <--- MSP に接続要求
Connected to 133.5.9.2
220 Service ready for new user
Name (kyu-msp:a79999a): a79999a  <--- アナタの課題番号
331 User name okay, need password
Password: _____       <--- アナタのパスワード
230 User logged in, proceed
ftp>

```

この時点で【FTP とは何ぞや？】と思った人がいるはず。FTP についての詳しい説明は、当センター発行の『利用の手引 MSP:TCP/IP 編』にバッチリ書いてあります。その中から 3 行程あげておきましょう。

*九州大学大型計算機センター・システム管理掛

¹当センターに登録をしていない場合は、所定の「計算機利用申請書」に必要事項を記入し、所属の連絡所を通じてセンターに提出して下さい。詳細は『利用の手引 基本編』九州大学大型計算機センターを参照下さい。

FTP は LAN(Local Area Network) で接続されたワークステーション間のファイル転送に広く用いられているプロトコルである。MSP の FTP サーバは UXP やワークステーションからのファイル転送要求に対して、MSP データセットとファイル転送機能を提供する。

(原文まま)

ということです。

また、冒頭で入力した open 133.5.9.2 は、当センター MSP システムの IP アドレスです。ちなみにドメイン名は kyu-msp.cc.kyushu-u.ac.jp です²。

では、続けましょう。

```
ftp> pwd ↵ <--- 現在地表示のコマンド
251 ''/A79999A'' is current directory <--- 課題番号をホームディレクトリとみなす
ftp> ls ↵ <--- MSP 側のデータセット一覧を表示する
200 Command okay
150 Open data connection
@AAA.FORT
ABC.FORT/
BBC.TEXT
TEST.DATA/
TESTZ.FORT/ <--- / 印はディレクトリとみなされた区分データセット
226 File transfer complete
100 bytes received in 0.2 seconds (0.35 Kbytes/s)
```

アナタが MSP に送り込みたいファイルは、Fortran のソースプログラム~/boss/kenq.f と、データファイル~/fortdata/kenq1.data だとします。

まず、ディレクトリ~/boss に移動し、その後 KENQ.FORT³という名前⁴で MSP に転送してみましょう。

```
ftp> lcd boss ↵ <--- ホームディレクトリから boss へ
Local directory now /home/usra/a79999a/boss
ftp> !ls -x ↵ <--- boss ディレクトリ下を表示
asobi      kowai      isogi      kenq.f      gomi
ftp> put kenq.f kenq.fort ↵ <--- ついに MSP へ転送
200 Command okay
150 File status okay;about to open data connection
226 File transfer complete
local: kenq.f remote: kenq.fort
721 bytes sent in 0.05 seconds (1.8e Kbytes/s)
```

同じようにワークステーション側のデータファイル~/fortdata/kenq1.data をKENQ1.DATA という名前で MSP に送ります。そのためにディレクトリを移動します。

²詳しくは『利用の手引 MSP:TCP/IP 編』で。

³MSP 側の英字表記は区別のために大文字にしています。MSP 上では基本的に大文字と小文字の区別はありませんので、MSP 上で小文字を打ち込んでも、特別差し支えありません。(MSP 専用端末では、属性により大文字と小文字の区別をつけるものもあります。)

⁴MSP 上では、Fortran プログラムのファイル識別修飾子を FORT にしている方(～.FORT)が、なにかと便利です。

```
ftp> lcd ../fortdata  ␣      <--- ディレクトリ boss から fortdata へ
Local directory now /home/usra/a79999a/fortdata
ftp> !ls -x  ␣      <--- fortdata ディレクトリ下を表示
asia.data      cosmo.data      kenq1.data      kenq2.data
ftp> put kenq1.data  ␣      <--- 送り先が同じ名前なら省略可能
200 Command okay
150 File status okay;about to open data connection
226 File transfer complete
local: kenq1.data remote: kenq1.data
98 bytes sent in 0.0005 seconds (0.8e Kbytes/s)
ftp> bye  ␣      <--- これを知らなきゃ始まらない、いや終らないFTPの終了コマンド
221 Service closing CONTROL connection
user%
```

【は一、これで一段落だ。】と思ったアナタは大間違い。

【わかってるって。MSPでジョブ流せば終りだろ?】などと思ったアナタも、残念ですが違います。なんとワークステーション→MSPのファイル転送処理で、大抵の人がぶちあたる壁が存在するのです。ををを。その壁とは(送り元)ファイルと(送り先)データセットの形式の違いによるものですが、そんなに心配はいりません(と、安心させておく)。さあ勇気を出して、次いってみよう(だみ声)。

プログラムが潰れてる!

うまくファイルが転送できたかどうか、MSPに入って確認してみます。

```
user% telnet 133.5.9.2  ␣      <--- MSPのTELNETサーバへ
Trying 133.5.9.2 ...
Connected to 133.5.9.2.
Escape character is '^]'.

PLEASE KEY IN ''TERMINAL TYPE'' (E:END,%X:XTERM6683,NULL:HELP) --->_  ␣
                                     ↑ 空 return
*** TERMINAL TYPE LIST BEGIN ***
   1. VT#TTY      2. VT#TTYJ    3. VT#9526    4. VT#6650    5. VT#6650S
   6. SUN#9526    7. NEWS9526   8. G15#6650  9. FMR#6650  10. TTY
  11. SUN#6650   12. NEWS6650  13. OPENWIN  14. MSWIN    15. MSWIN#S
  16. MSWIN#J   17. NWC
*** TERMINAL TYPE LIST END ***
PLEASE KEY IN ''TERMINAL TYPE'' (E:END,%X:XTERM6683,NULL:HELP) --->
```

【おいおい。TELNETっていきなり言われてもなあ。しかもTERMINAL TYPEって何だ?】と戸惑っているアナタの姿が見えるので、説明しましょう。

まずTELNETについてですが、これもやはり『利用の手引 MSP:TCP/IP編』にズバツと書いてありますから自分で読んで下さい。という意地悪はやめて、少しですが書いておきます。

MSP の TELNET サーバは、UXP やワークステーションなどから MSP の TSS を利用する機能を提供する。MSP の IP アドレスは 133.5.9.2 でホスト名は kyu-msp.cc.kyushu-u.ac.jp である。TSS 端末には F6650 などの富士通専用端末やパソコン端末があるため、TELNET サーバにはこれらの端末エミュレーション機能も提供している。

(原文まま)

TELNET サーバについてはなんとなくわかったけれど、じゃあ TERMINAL TYPE は？ということになります。これは、「ワークステーションに日本語が使えないもの、キーボード配列が異なるもの、キー押下時に発生するキーシーケンスが異なるものなどがあるために、TELNET サーバが各種ワークステーションの端末属性を定義できるようにしている機能」のことです。ここで各種詳しく説明するのがベストですが、スペースと私のやる気の問題上、勝手ながら省かせて頂きます。いえいえ大丈夫。これも『利用の手引 MSP:TCP/IP 編』にキッチリ説明が書かれています。

ここらへんで、なんとなく霧が晴れたような気になりましたね？
では先を急ぎましょう。

```
PLEASE KEY IN ''TERMINAL TYPE'' (E:END,%X:XTERM6683,NULL:HELP) --->4 
    ↑ほとんどの場合 4 でいいのですが、なかには合わないワークステーションもあります
ENTER USERID -
a79999a           <--- アナタ (MSP) の登録番号
+ PASSWORD ?
_____           <--- アナタ (MSP) のパスワード
KDS40613I USER(A79999A) LAST ACCESS DATE(1995.09.30),TIME(16:42:54)
KEQ56455I A79999A LOGON IN PROGRESS AT 17:25:53 ON AUG 30, 1995
JOB NO = TSU1648 CN(01)
KEQ56951I NO BROADCAST MESSAGES

*****
**              次回のシステム停止時間は以下の通りです.              **
**                                                    **
**              ×月××日 (×)   5 : 0 0 ~ 1 2 : 3 0 まで              **
**                                                    **
*****

READY
PFDE           <--- エディタを立ちあげる
```

【PFDE? またまた見慣れないコマンドだぞ。】と心配しないで下さい。九大大型計算機センター広報 Vol.25 No.2 (1992) に『PFD と PFD/E の使用法』という非常に良い解説記事がありますので、これも「利用の手引」同様、是非手元に置いてほしいものです⁵。

このPFDE コマンドを打つと、画面は以下のようになります。

⁵ 置くだけではダメです。よく読んで実践して下さい。

```

-----< データセット一覧 >-----
コマンド ==>                                     移動量 ==> CUR
ユーザ識別修飾子 : A79999A

                                     ユーザ登録名 -A79999A   時刻 -17:29
***** データセット名 --< コマンド入力フィールド >----- メッセージ -----
',',',',', @AAA.FORT          -
',',',',', ABC.FORT          -
',',',',', BBC.TEXT          -
',',',',', KENQ.FORT          -
',',',',', KENQ1.DATA         -
',',',',', TEST.DATA         -
',',',',', TESTZ.FORT        -

```

ここで、ちゃんとKENQ.FORTとKENQ1.DATAが作成されているのがわかりますね。では、中身を覗いてみましょう。

```

-----< データセット一覧 >-----
コマンド ==>                                     移動量 ==> CUR
ユーザ識別修飾子 : A79999A

                                     ユーザ登録名 -A79999A   時刻 -17:29
***** データセット名 --< コマンド入力フィールド >----- メッセージ -----
',',',',', @AAA.FORT          -
',',',',', ABC.FORT          -
',',',',', BBC.TEXT          -
',',',',', KENQ.FORT          E  ☐      <--- エディットのE
',',',',', KENQ1.DATA         -
',',',',', TEST.DATA         -
',',',',', TESTZ.FORT        -

```

エイヤツとリターンキーを押すと...

```

EDIT --- A79999A.KENQ.FORT ----- 表示欄 001 072
コマンド ==>                                     移動量 ==> CUR
***** ***** データの先頭 *****V10L30*****
000001      PROGRAM KENQ
000002      PARAMETER(MAXN=1049)
000003      REAL  AR(MAXN,MAXN),BR(MAXN,MAXN),CR(MAXN,MAXN),
000004      &      PIR,FR(MAXN),VWR(MAXN)
000005      REAL*8 AD(MAXN,MAXN),BD(MAXN,MAXN),CD(MAXN,MAXN),
000006      &      FD(MAXN),VWD(MAXN),T(8),CPU(4),PID,
000007      &      OP1,OP2,FLOPS(4)
000008      INTEGER N,K,I,J,IS,IP(MAXN),ICON,ITER
000009      WRITE(6,100)

```

一見まともに見えますが、ここに悪魔が潜んでいます。ためにこのプログラムをRUN コマンドで実行してみてください。

```

EDIT --- A79999A.KENQ.FORT ----- 表示欄 001 072
コマンド ==> RUN [ ] 移動量 ==> CUR
***** データの先頭 *****V10L30*****
000001 PROGRAM KENQ
000002 PARAMETER(MAXN=1049)
000003 REAL AR(MAXN,MAXN),BR(MAXN,MAXN),CR(MAXN,MAXN),
000004 & PIR,FR(MAXN),VWR(MAXN)
000005 REAL*8 AD(MAXN,MAXN),BD(MAXN,MAXN),CD(MAXN,MAXN),
000006 & FD(MAXN),VWD(MAXN),T(8),CPU(4),PID,
    
```

するとどうでしょう。

【FORTRAN77 EX 翻訳開始】

FORTRAN77 EX 診断メッセージ：プログラム名 (MAIN),FLAG(I),OPTIMIZE(B)

```

JWD1026I-S END 文がありません。
JWD1043I-I 行識別番号が数字列ではありません。
JWD1044I-I 行識別番号が昇順になっていません。
JWD1105I-S 00000000 変数名, 配列名, 配列要素名又は部分列名の指定しか
許していない場所に, それら以外の式を指定しています。
JWD1110I-S 00000000 英字名であるべきところが英字名ではありません。
JWD1121I-S 00000000 文の後ろに余計な文字があります。
JWD1302I-S 00000000 この文は,FORTRAN の文とはみなせません。
JWD2006I-I I は, 宣言だけされていて, 実行文で引用されていません。
JWD2006I-I J は, 宣言だけされていて, 実行文で引用されていません。
    
```

【翻訳終了】 , 完了コード = 12

ばばーん !! 完了コード = 12 という恐ろしい文字とともにワサワサとエラーメッセージが出現しましたね。

【何故何故? ワークステーションじゃ正常に動作していたのに. 】

とどめに、この画面で行番号を新たにふりなおすREN (RENUMBER) コマンドをうってみましょう。

```

EDIT --- A79999A.KENQ.FORT ----- 表示欄 001 072
コマンド ==> REN [ ] 移動量 ==> CUR
***** データの先頭 *****V10L30*****
000001 PROGRAM KENQ
000002 PARAMETER(MAXN=1049)
000003 REAL AR(MAXN,MAXN),BR(MAXN,MAXN),CR(MAXN,MAXN),
000004 & PIR,FR(MAXN),VWR(MAXN)
000005 REAL*8 AD(MAXN,MAXN),BD(MAXN,MAXN),CD(MAXN,MAXN),
000006 & FD(MAXN),VWD(MAXN),T(8),CPU(4),PID,
000007 & OP1,OP2,FLOPS(4)
    
```

さあ、どうなりますか...

```

EDIT --- A79999A.KENQ.FORT ----- 表示欄 001 072
コマンド ==> 移動量 ==> CUR
***** データの先頭 *****V10L30*****
000100 000001000GRAM KENQ
000200 00000200RAMETER(MAXN=1049)
000300 00000300AL AR(MAXN,MAXN),BR(MAXN,MAXN),CR(MAXN,MAXN),
000400 00000400 PIR,FR(MAXN),VWR(MAXN)
000500 00000500AL*8 AD(MAXN,MAXN),BD(MAXN,MAXN),CD(MAXN,MAXN),
000600 00000600 FD(MAXN),VWD(MAXN),T(8),CPU(4),PID,
000700 00000700 OP1,OP2,FLOPS(4)
000800 00000800TEGER N,K,I,J,IS,IP(MAXN),ICON,ITER
000900 00000900ITE(6,100)
001000 00001000ITE(6,105)
001100 00001100
001200 00001200ER=10
001300 00001300 10 K=2,ITER
001400 000014002**K
001500 000015001 = 2.0D0*DFLOAT(N)**3
001600 000016002 = (2.0D0*DFLOAT(N)**3)/3.0D0 + 2.0D0*DFLOAT(N)**2

```

【ギャアアア！プログラムが潰れたあ！】

そうです。これこそみんながプチ当たる壁なのです。

意地悪な壁

ここからその謎を解き明かしたいと思いますが、【そんな暇は無いんじゃない!!】とお急ぎの方はこの項目を飛ばして次に進んだ方が良いでしょう。

頭に入れよう：その1

ワークステーションやパソコンなどと違って、MSPのエディタやFortranには、行番号という概念があります。そのため、FTPを使ってワークステーションなどからMSP上にファイルを転送した場合、いくら嫌だと言っても強制的に行番号をつけられてしまいます。

頭に入れよう：その2

Fortranの記述形式には「固定（標準）形式」と「自由形式」があるのはご存知でしょう。これらは、見た目には少々文法が違うだけなのですが、ことコンパイル処理になると、形式の差は大きくなります。ちなみに、ワークステーションやパソコンで書かれるプログラムは、通常「固定形式」です。（こういうことを意識している人は、あまりいないかもしれません。）

頭に入れよう：その3

MSPのデータセット（ワークステーションなどで言うファイルのことですね。）は、「固定長レコー

ド形式」と「可変長ブロック化レコード形式」⁶があります。Fortran コンパイラは、このデータセットの属性を見て処理をしています。「固定長レコード形式」ならば「固定形式」で、「可変長ブロック化レコード形式」ならば「自由形式」でコンパイルするのです。たとえ中身がどうであれ、強引にやってしまう困った奴です。

頭に入れよう：その4

FTP を使って、MSP 上にデータセットを新規作成した場合、デフォルトとして「可変長ブロック化レコード形式」ができあがります。これはあくまでデフォルトですので、設定の変更は可能です。

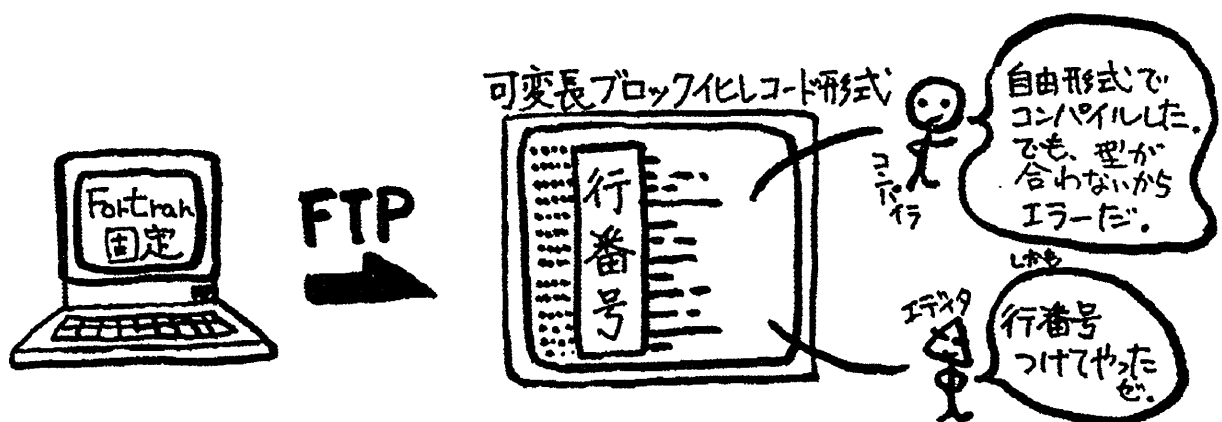
これら4つが激しく絡み合あって、非常に複雑な状況を作り出します。では、その激しい絡み合いを分析してみましょう。

考えよう：その1

ワークステーションにある「固定形式」の Fortran プログラムを、MSP に（データセット新規作成で）送ります。これは先ほどアナタが絶望した状況と同じです。この場合《頭に入れよう：その4》で触れたように「可変長ブロック化レコード形式」のデータセットができあがりますが、中身は「固定形式」です。中身と器の型が合っていませんね。

さて、Fortran コンパイルは、どのような動きをするのでしょうか。そうです。データセットの型を優先させ、「自由形式」で強引にコンパイルしてしまいます。この時点で当然不整合があらわれますが、悪夢はここだけにとどまりません。MSP のエディタ⁷もまた、データセットの属性を見て、中身のプログラムの形式を判断しています。この場合データセットは「可変長ブロック化レコード形式」ですから、エディタも『じゃあ、中身は「自由形式」なのね。「自由形式」ならば、1カラム目から8カラム目まで行番号付けなきゃ。』と、これまた強引に行番号を上書きしてしまうのです。転送された何も知らないウブなプログラムは、忠実に左詰めを行なっていますので、エディタの働きで数カラム分プログラムが潰れてしまうのです。あーあ。

結局この状況（Fortran 「固定形式」→新規作成「可変長ブロック化レコード形式」）では、コンパイラやエディタが目の前に立ちほだかり、エラーメッセージがワサワサと出て失敗するはめになるのです。おお恐い。



考えよう：その2

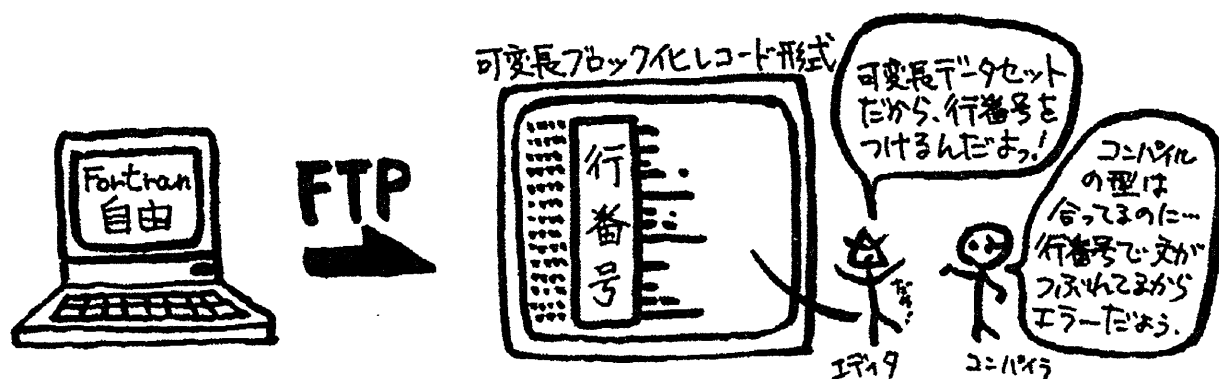
ワークステーションにある「自由形式」の Fortran プログラムを、MSP に（データセット新規作成で）送ります。

⁶他にも属性はありますが、ここでは重要な2つだけを例にあげています。属性内容については『利用の手引 TSS 編』に詳しく説明してありますので参照下さい。

⁷さっき PFDE と打ちましたね。あれです。

【ふふふ、これなら中身と器の組合せがバッチリだから、うまくいくだろう。】なんて、残念ですが大間違いで、エディタがまた勝手なことをしてくれています。

《考えよう：その1》で触れたように、データセットの属性「可変長ブロック化レコード形式」を見て、1カラム目から8カラム目まで行番号をつけます。しかも上書きです。中身と器、いくら形式が合っても、行番号をドローンと上書きされてはコンパイルもできません。あーあ、これも失敗。まことに残念です。



FTP の設定を変えてみよう

【新規作成で「可変長ブロック化レコード形式」を作らなきゃいいんだろ？それって、どうやるの？】というアナタの声にお答えして、「FTP の中で、データセットのデフォルト属性を変更する」というやり方を紹介しましょう。デフォルト属性を変更するということは、新規作成した場合「固定長レコード形式」のデータセットを作るということです。

```
ftp> quote site attrf set rec(fb) lr(80) bl(23440)  [↵] <--- ここで属性変更
200 Command okay
ftp> put kenq.f kenq.fort  [↵] <--- MSP へ転送
200 Command okay
150 File status okay;about to open data connection
226 File transfer complete
local: kenq.f remote: kenq.fort
721 bytes sent in 0.05 seconds (1.8e Kbytes/s)
```

quote site... の1行を入力するだけで、その後 MSP 上に新規作成したデータセットは、全て「固定長レコード形式」になります。しかし FTP を終了した時点で、その設定は無効になりますから気をつけましょう。

考えよう：その3

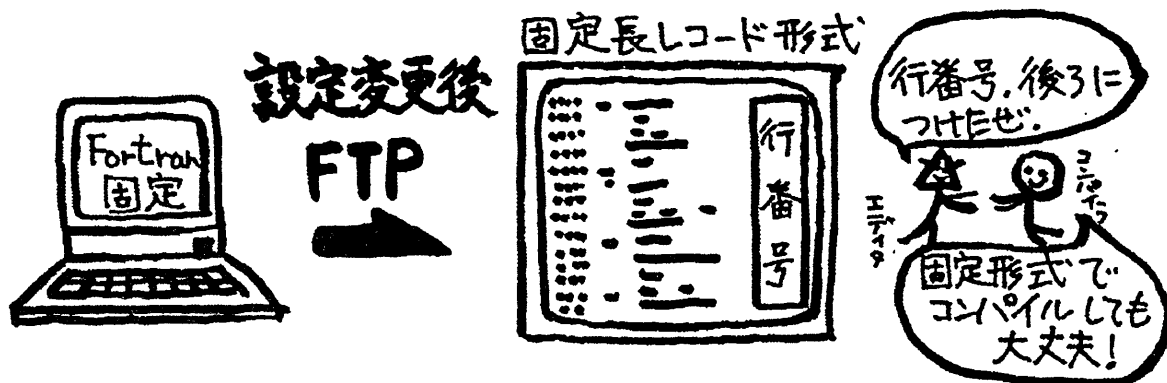
【新規作成データセットの設定変更は分かったし、「固定長レコード形式」なら、もう大丈夫だね。】ブブー!! まだまだ落とし穴があります。

これまでの《考えよう》をよく読んでいたらわかってると思いますが、いくら送り先を「固定長レコード形式」のデータセットにしても、中身が「自由形式」のプログラムだったらいけませんよ。（なんで？と思う人は《頭に入れよう：その1》から読みなおし!）

しかし前出のとおり、通常ワークステーションなどでかかれた Fortran プログラムは「固定形式」です。よってほとんどの場合中身と器の型が一致しますね。それで、エディタの動きですが、「固定形式」の場合は1カラム目から行番号を入れるのではなく、73カラム目から80カラム目まで入れてしまいます⁸。

⁸ 「固定形式」は一行80カラムと決っています。

その「固定形式」に則ったプログラムなら、エディタだって悪さのしようがないわけです。
おおおお、よかった。ここにきてやっと救われた感じがしますね。



【で、ずっと気になってたんだけど、一番左端についてる6桁の数字は行番号じゃないの？】
これはエディタの管理番号で、データセットの中身とは直接関係はありません⁹。

そもそも何故 Fortran には、固定と自由という2種類の形式がまぎらわしくも存在するのでしょうか。これはプログラムの一行分を一枚のカードとして穴をあけ、それを読み込ませていたカードリーダー（若いユーザーなら知らない。もちろん私も知らない）時代の余韻が残っているからです。そのカードリーダーの流れをくむのが固定形式（標準形式とも言いますね）で、TSS 時代¹⁰になって生まれたのが自由形式というわけです¹¹。この二つは今のところ淘汰されることもなく、現役で頑張っているのです。ううむ。

考えよう：その4

【これまでは FTP 上で新規作成した時の話をしたけど、先に MSP に入って器だけ作っておくというのはどうかな？】

おーっと！いい意見ですね。先に器だけ作っておく¹²というのはグッドです。送り元のプログラムが「固定形式」で、MSP 上のデータセットを「固定長レコード形式」にしておけば問題ないです。しかし、中身が「自由形式」であつたら結局《考えよう：その2》の時と同じで、プログラムが潰れてしまいます。

送った先でデータセットを変身させる

【面倒なんで、転送後に形式を書き換える方法ない？】

ありますよ。一つめは、アナタが Fortran 「固定形式」のプログラムを送ったところ「可変長ブロック化レコード形式」のデータセット KENQ.FORT が作成され、これではまずいと思って「固定長レコード形式」のデータセット NEW.FORT にする例です。

TELNET で MSP に入ってください。

⁹行番号と管理番号が一致してることもあります。離して考えて下さい。

¹⁰TSS (Time Sharing System)。多数の利用者が待ち時間を意識せず、同時に計算機システムを利用できる。若い人にとっては当たり前のシステムですが、TSS 登場時はそれなりの感動があったらしいです。（当時の様子をよく知りたい場合は、広報 Vol.27 No.4(1994) 掲載の『苦むした相談員の思い出、その他』がオススメです。）

¹¹データセットの属性などは『利用の手引 TSS 編』を参照下さい。

¹²MSP に ALLOCATE というコマンドがあります。便利ですので『利用の手引 MSP コマンド編』をよく読んで、有効に使いましょう。

READY

COPY KENQ.FORT NEW.FORT NON RECFM(FB) LRECL(80) BLKSIZE(23440) 

↑ MSP 上の可変長形式のデータセット KENQ.FORT を、固定長形式の NEW.FORT にコピー

READY

次は Fortran 「自由形式」のプログラムを送った「可変長ブロック化レコード形式」のデータセット KENQ2.FORT が、このままだと行番号を割り当てられて 8 桁潰れてしまうので、それを防ぐために NEW2.FORT へコピーする例です。

同じく TELNET で MSP に入ってください。

READY

COPY KENQ2.FORT NEW2.FORT NON NUM2(1 8) RECFM(VB) LRECL(255) BLKSIZE(23440) 

↑ MSP 上の KENQ2.FORT を、可変長形式行番号付きの NEW2.FORT にコピー

READY

どちらの場合もコピーをただけなので、コピー元のデータセットは残っています。このコピー元は役に立たないでしょうから、消してしまいましょう¹³。

もっと簡単に実行したい

【オレは一回だけ、実行できりゃいいんだけどなあ？】

ここまでは、プログラム転送後もそのデータセットを何度か使用する、ということを念頭に説明しました。しかし、もっともっと簡単に殺那的な応急処置を紹介しておきます。

これは中身が Fortran 「固定形式」で、器が「可変長ブロック化レコード形式」の場合です。FIXED と書かれた部分に注目。

EDIT --- A79999A.KENQ.FORT ----- 表示欄 001 072

コマンド ==> FORT FIRSTCOL(5) FIXED  移動量 ==> CUR

***** データの先頭 *****V10L30*****

```
000001      PROGRAM KENQ
000002      PARAMETER(MAXN=1049)
000003      REAL  AR(MAXN,MAXN),BR(MAXN,MAXN),CR(MAXN,MAXN),
000004      &      PIR,FR(MAXN),VWR(MAXN)
000005      REAL*8 AD(MAXN,MAXN),BD(MAXN,MAXN),CD(MAXN,MAXN),
000006      &      FD(MAXN),VWD(MAXN),T(8),CPU(4),PID,
000007      &      OP1,OP2,FLOPS(4)
```

同じように中身が Fortran 「自由形式」、器が「可変長ブロック化レコード形式」で組合せ的話には問題なさそうな場合。この場合は何度も言いますが、行番号分のカラムの移動が必要になりますので、FIXED のところを FREE にすればよいわけです。

¹³ 消去コマンドなどは『利用の手引 MSP コマンド編』を参照下さい。

エピローグ

といったところで、明日までボスに仕事を提出しないといけないアナタの苦痛が、少しは和らいでしょうか？

いやあ、本当に時代は進んでしまいましたね。パソコンやワークステーションを持っていると、線を“ちょちょっと”つなぐだけで、遠く離れたところにあるテキストファイルや画像データやらが手元に届くようになるんですから¹⁴。もっと便利さを追求したいアナタや、もっとしっかりと勉強したい私自身のために、次回は画像データの出力について解説しようと思います。（ををを！言い切ったな。）では、またお目にかかりましょう。

参考文献一覧

全編で参考にした文献を載せておきます。

- 『利用の手引 基本編』 九州大学大型計算機センター， 1995.
- 『利用の手引 TSS 編（第3版）』 九州大学大型計算機センター， 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， 1993.
- 『利用の手引 MSP:TCP/IP 編』 九州大学大型計算機センター， 1994.
- 『PFD と PFD/E の使用法』 九州大学大型計算機センター広報， Vol.25， No.2， pp.119-165， 1992.

¹⁴ こういう仕事に携わっている私が一番驚いている。