

# JによるMS-DOS N88BASICのプログラムの自動修正

## The Automatic Correction of N88BASIC Programs by J

慶応義塾大学理工学部  
竹内寿一郎

### 1. はじめに

身の回りに過去の遺産として、昔のMS-DOS N88BASICがPC-9801もしくはEPSON-286シリーズで稼動していたものが多く存在している。近年のWindowsの普及により、これらのプログラムやハードウェアは使われなくなって、倉庫に眠っているのが殆どである。最近になって依然として活躍をつづけているシステムがあり、それがいろいろなところで障害が生じていて、その対策を相談されたのでここに取り上げてみた。そのマシンはかなり古く、さすがにフロッピィとかコネクタに異常が生じ、ハードウェアの痛みもひどかった。置き換え費用に十分なお金が懸けられないとのことで、中古のPC-9821(Windows95)を用いることになった。このマシンではN88BASICはそのままで使用可能であるが(ただし、いくつかの命令は仕様の変更がある)、マシンによってはプログラムに書かれたドライブ記号を「A:¥」から「C:¥」または「B:¥」に全て変更しなければならなくなった。なお、現在のWindows XPや2000上で稼動するBASIC98などで使用するには、プログラムの中のドライブ名は「C:¥」に変えなければならない。いずれにせよ従来のN88BASICでは「A:¥」がデフォルトであったため、全てのプログラムでドライブ名を修正することになったが、このシステムではおよそ60個の独立したプログラムがあり、それが相互に呼び合っていて動いているので、することは簡単ではあるが処理時間が多く取られるのが予想された。そのため、Jの関数を使用して自動変換することを試みてみた。

### 2. 問題点

N88BASICを使っていた方はよく知っていると思うが、MS-DOS N88BASICプログラムは内部コードで書かれた場合と、ASCII(テキストファイル)で書かれた場合の2通りの保存方法があり、いずれも全く同じように動く。プログラムの読み出し(LOAD)・書き出し(SAVE)の時間が異なり、また記録容量も当然異なっているし、内部コードで書かれている方が圧倒的に早い。特に長いプログラムでは顕著になる。例えばASCIIコードで保存するとPRINTは5バイトで表されるが、PRINTなどの予約語は内部コードでは1バイトで表現されるから少ない文字数で済む。現在保存されているプログラムは、殆どすべてが内部コード表現で書かれているので、それをテキストファイルに変えてから置換作業をする(WindowsのEditorで置換する)ため、非常に手間がかかる(次ページ図1、2に内部コードとASCIIコードで保存した場合のリストがあるのでそれらを参照のこと)。

### 3. JによるAMEND(修正)

まず、オーソドックスな変換作業について検討してみた。「A:¥」はプログラムの何処に、何箇所あるか分からないので、次の手順で変換作業を行われねばならない。

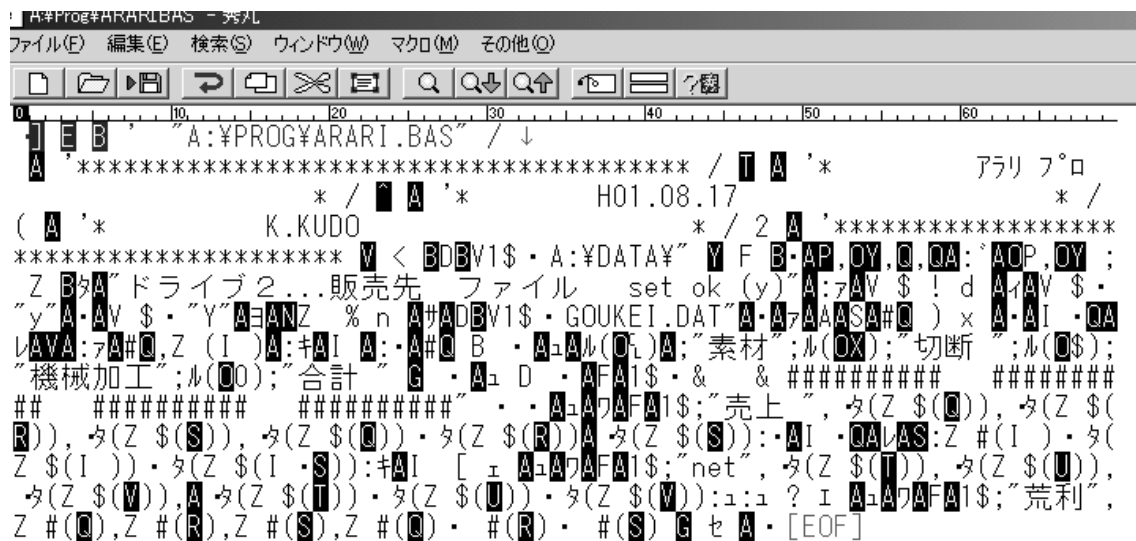


図1 . 内部コードを使用したファイルを編集ソフトで開いた例  
 この中にテキストの文字列がいくつか見られる



図2 . ASCIIで保存した同じファイルを編集ソフトで開いた例  
 当然であるがずっと大きいファイルになっている

- ( 1 ) N88BASIC の起動
- ( 2 ) 修正すべきプログラムのロード
- ( 3 ) ロードしたプログラムを ASCII 形式で保存する
- ( 4 ) 2 ~ 3 を繰り返し 60 回行い終了
- ( 5 ) Windows でエディターを起動
- ( 6 ) 修正すべきプログラムを選んでクリック
- ( 7 ) 全ての「A:¥」を「B:¥」に置き換える
- ( 8 ) 選んだプログラムを保存
- ( 9 ) 6 ~ 8 を繰り返し 60 回行って終了

このとき、全てのプログラムがテキストファイルになってから、5 ~ 9 の代わりに J の関数で処理をしても良い。

試みにプログラムが内部コード表現のまま、J の関数で置き換え (J では Amend) が出来るかどうかの実験を試みた。

```

dir=: 'A:\prog\'          NB. ディレクトリの指定
a=: fread dir, 'arari.bas' NB. ファイルの読み込み
$a
994                      NB. ファイルの大きさは 994 バイトである
d=: ('A:\' E. a)#i.#a     NB. A:\ の先頭指標を全て見つける
d
7 262                   NB. 先頭の指標は 7 と 262 である
id=: d+"0 1 i.3          NB. 全ての "先頭指標+0 1 2" をつくる
id
7 8 9                  NB. 7 8 9 と 262 263 264 を直せばよい
262 263 264
b=: 'B:\' id }a         NB. AMEND({}) で「B:¥」を各指標 id へ代入
7 8 9 262 263 264{b
B:\B:\                  NB. 確かに置き換わっている
7 8 9 262 263 264{a
A:\A:\
b fwrite <dir, 'arari_a.bas' NB. 出来上がった b を名前を変えて保存

```

何と内部コードのままのファイルで変換できることに感動してしまった。

#### 4 . J による自動置き換えプログラム

まず、変換すべきファイルのファイル名すべてを含むファイル、FILE.TXT を作成する。J で MS-DOS コマンドが使えるれば簡単であるが、Windows のコマンドは J 上では使えないので、MS-DOS でファイルを作成する。MS-DOS コマンドを入力する画面から、

```
A:\prog\> DIR > FILE.TXT
```

として、全ての BASIC プログラムが入ったディレクトリの一覧を表にしたファイル FILE.TXT を作成する。

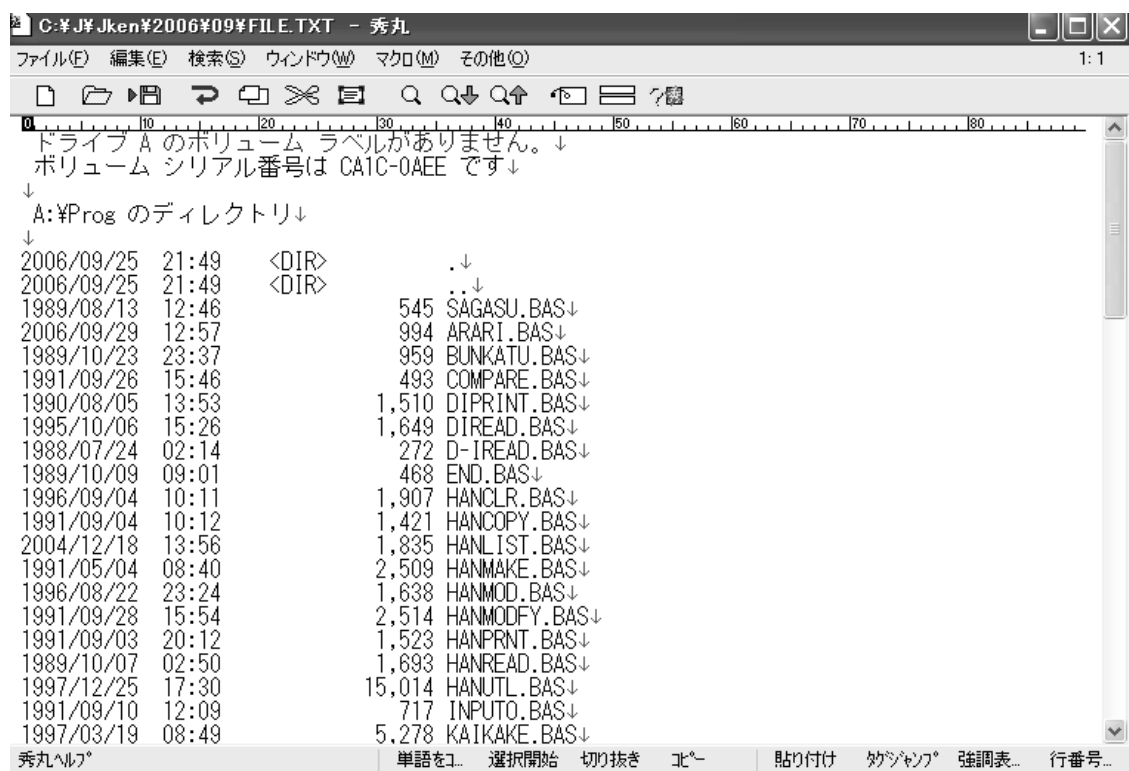


図 3 . 変換すべき BASIC プログラムの一覧、DIR で記述されたファイル日付、ファイルサイズ等余分な情報が入っている

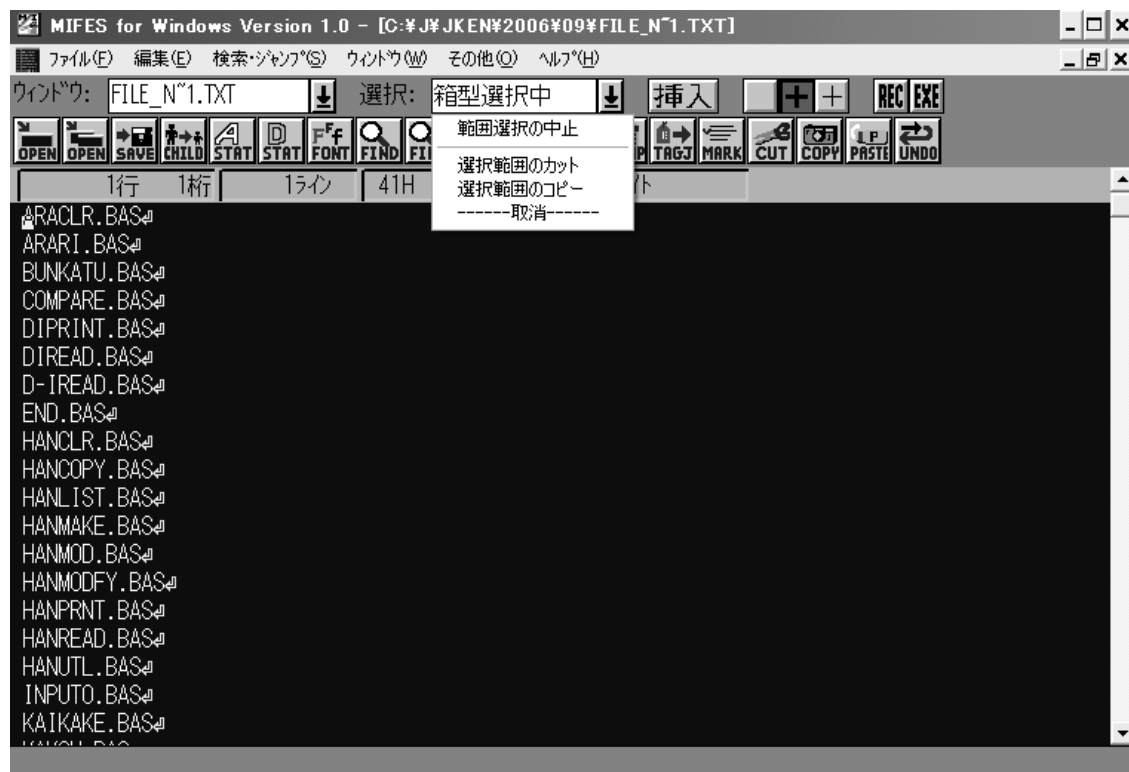


図 4 . 変換すべき BASIC プログラムの一覧、箱型編集で余分な情報をカットアルファベット順にしたファイル名だけにした

ディレクトリ情報から箱型編集が出来るエディタ、Mifes-Windows 版を使用して、ファイル名以外の情報をカットして FILE\_NAME.TXT というファイルを作成した。これを使用して J による自動修正関数によりすべての BASIC プログラムの「A:¥」を「B:¥」に置き換えた。

```

0!:0<'c:\j503\system\main\files.ijs'  NB. 事前準備
readchar=:3 : '>each cutopen(1!:1 y.)-.CR'  NB. 文字列読み込み
F=:readchar<dir,'FILE_NAME.TXT'
Take=:3 : 0
i=.0
label_L1.
if. (#F)<i=.>:i do. goto_owari. end.
NB. if. 5<i=.>:i do. goto_owari. end.
a=.fread fl=.dir,Fn=(>(i-1){F)
d=( 'A:\' E. a)#i.#a  NB.A:\の場所を探す
id=.d+"0 1 i.3
b=. 'B:\' id }a  NB.A:\のところを B:\に置き換えている
b fwrite <'B:\Prog\'',Fn
print fl
goto_L1.
label_owari.
)

```

#### 使用法

```

Take ''
ARACLR.BAS
ARARI.BAS
BUNKATU.BAS
COMPARE.BAS
DIPRINT.BAS
DIREAD.BAS
D-IREAD.BAS
END.BAS
HANCLR.BAS
HANCOPY.BAS
HANLIST.BAS
HANMAKE.BAS
.
.
.

```

この関数を使うことによって 60 数個の BASIC プログラム内の「A:¥」を「B:¥」に置き換える作業を約 30 秒で処理することが出来た。