

## スマホの上の J-qt と J-console —プログラム実行の操作とシステム構造の比較—

西川 利男

スマホ（スマートフォン）の上でJが動かせるので、昨年暮れからいろいろやってきた。鳥邊鍊太郎氏、志村正人氏によると J on スマホの実行は J801-qt でとすすめられた。なるほど、Jの機能レベルについては、かなり違う。しかし、私自身、J-console で始めてしまい、いろいろ格闘したので、J-console にも捨てがたいところがある。

両者についてプログラムの実行の操作とシステム構造について、私なりに比較してみた。

### 1. 実行操作の点からの比較

#### • J701-console の場合

原則として、コマンド入力で実行操作する。くわしくはすでに、以下に報告済みであるので、省略する。

「スマホー私にとっての哲学とその上でJを動かす奮闘記」 JAPLA 研究会資料 2014/12/6

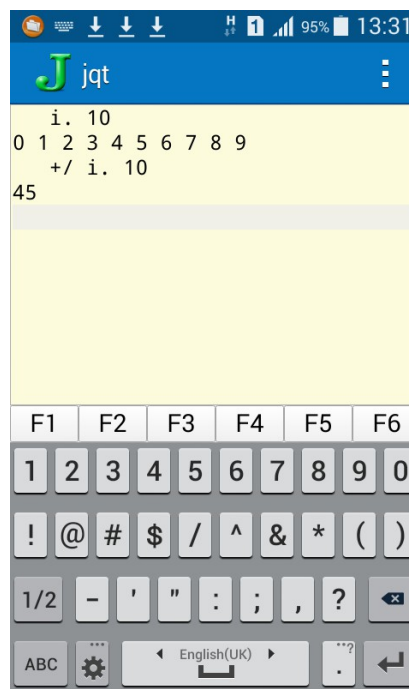
「Jによるスマホのシステム処理ーディレトリ構造をさぐる」 JAPLA 研究会資料 2015/1/17

一言、付け加えると、J701-console をスマホでなく、パソコンの上で起動すると、昔なつかしい黒い DOS 画面上にJの実行が表示される。

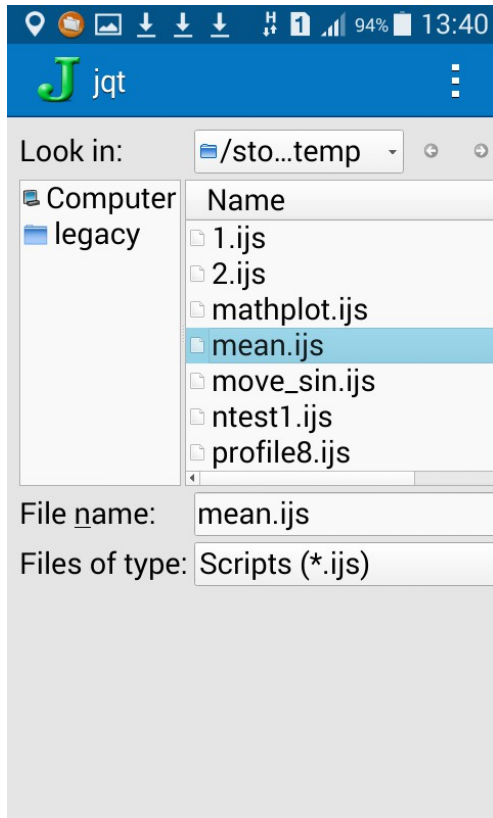
#### • J801-qt の場合

実行(term)、システム操作、編集(edit)と3種類の画面を切り替えて操作する。

まず、起動した画面では、図のように対話的にJが実行される。



次に、すでに作成したプログラムはJ-qt画面の右上のメニューから Open Temp File として、現れる左下のリストからたとえば mean.ijs のように選んで、2回タップすると、右下のような編集画面に切り替わりプログラムのリスティングが現れる。



編集画面の右上のメニューから、[run] - [All lines]を選ぶと、画面ははじめにJを起動したときの実行画面に戻り、ロードしたプログラムのリスティングも表示される。

さらに、ここでは、定義した動詞 mean が実行できる。

ここで、最初の

```
mean i. 101
```

50

は、プログラムをロードしたときの実行であり、

次の

```
mean i. 10
```

4.5

実行画面で、手による入力での実行である。

## 2. J801-qt のファイル構造

J801-qt ではユーザ作成のプログラムファイルは、J701-console の場合と同様に

```
J801-user
```

のフォルダーのもとに格納される。

しかし、多機能を支える非常に多くの J の Add-on プログラムは別のフォルダーのもとに格納されている。

```
本体/Android/data/com.jssoftware.android.qtide/files/
```

```
|—addon
```

```
|—bin
```

```
|—system
```

として、そのサブフォルダーとして多数のプログラムがあり、複雑である。

```
addon
```

```
|—tables —demos —control—
```

```
|—graphics —labs —core
```

```
|      |      | system
```

```
|      |      | math
```

```
|      |      | examples
```

```
|      |      | graphics
```

```
|      |— treemap
```

```
|      |— viewmat
```

```
|      |— tgsj
```

```
|      |— plot
```

```
|      |— print
```

```
|      |— grid
```

```
|      |— gl2
```

```
|—types
```

```
|—stats
```

```
|—xml
```

```
|—games
```

```
|—debug
```

### 3. J グラフィックス電卓としての利用

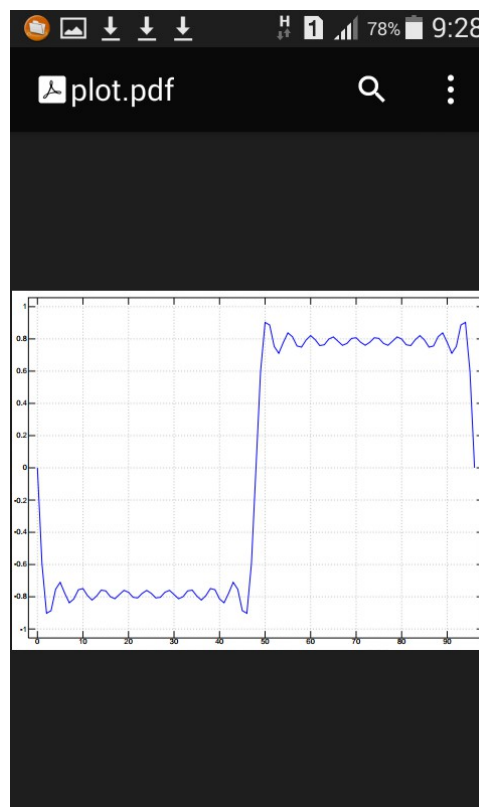
J on スマホの数学計算とグラフィックスへの利用として、矩形波関数のフーリエ変換による近似の例を示す。

fourier 10

とすると、右の図が表示される。ここでは J701-console で行った。

間宮真佐人、西川利男「化学計測のためのフーリエ変換法入門」 p.202, 共立出版(1983).

```
J Console
load 'fourier.txt'
four
3 : 0
fn =. y
fnodd =. 1 + 2 * fn
(1 % fnodd) * (sin fnodd * 2p1 *
1r96 * (i: 48))
)
fourier
3 : 0
n =. y
i =. 1
ft =. four 0
while. i < n do.
NB. rd 1
ft =. ft + four i
i =. i + 1
end.
plot ft
)
```



#### 4. 使い勝手の上からの両者の比較

私としては、現在のところ、まだスマホの上のJを使いこなしてはいるとは言えないが次のような感想を持った。

- Jプリミティブの実行であれば、コマンドレベルであるので、J-consoleで十分である。かつ安全である。手軽であることはもちろんである。
- J-qtのプログラムの作成、実行などは、複雑で分かりにくい。

Jを高性能な電卓としてなら、十分であるが、長年パソコンの上で行ってきたグラフィックスを含めた高性能のJの機能を実現するのはまだ先のことになろう。とくに、プログラミングの開発環境としては、満足できない。