

5. 技術解説

ちょっとおもしろい言語 (APL)

医学部原研遺伝学部門 吉川 勲

私、毎朝〇時〇〇分、西彼杵郡長与町高田郷 2360 番地 3 を車でスタート、道尾駅裏經由医学部キャンパスまで通勤しています。最近は信号機がかなり増え、まず百合野出口・マツダの坂・……如己堂・神学校・赤レンガの天主堂前と 17ヶ所の信号機を赤で止まったり、青でうまく通り抜けたり、1日1回、年間通して約 330 回くり返しています。

さて問題。17ヶ所それぞれの信号は全くデタラメに点滅していて、赤と青の時は半々の確率であるとして。もっとも最近の信号は複数の信号機を 1つの信号群として制御されているようですから停止と発進を頻々とくり返さないようにされています。それはさておき、さきの前提で考えると医学部キャンパスに到着するまで赤信号にひっかかり停止する回数は、日によって運良く 0 回（確率としては $(\frac{1}{2})^{17}$ ですが）からいらいらの上昇を招く 17 回まであります。そこで年間 330 回通勤するとして、信号で停止する回数に対応した運転日数の分布状態を知るシミュレーション実験プログラムを作れというのが問題です。これはコンピュータの中に一様乱数を発生させて行う簡単な Mante Carlo 法といったものです。

話の主題は上記の問題を FORTRAN 言語と APL 言語で記述し、APL 言語を少々、我田引水的に紹介しようというわけです。FORTRAN 言語でのプログラムを優秀な薬学部学生 H 君と、プログラミングのプロ Mrs. 情報処理センター S さんの 2 人に、APL 言語を不消私めがプログラムし比較してみます。

FORTRAN 言語で書かれた H 君のプログラムと実行結果

```

C      PROGRAM=1
      INTEGER A(400),C(20)
      REAL RAN
      IX=0
      DO 10 I=1,330
10     A(I)=0
      DO 20 I=1,330
      DO 25 J=1,17
      IX=IX+I+J
      CALL RANU2(IX,RAN,1,ICON)
      IF(RAN .GE. 0.5) A(I)=A(I)+1
25     CONTINUE
20     CONTINUE
      DO 30 J=1,17
30     C(J)=0
      DO 40 I=1,330
      DO 45 J=1,17
      IF(A(I) .EQ. J) C(J)=C(J)+1
45     CONTINUE

```

技術解説

```
40 CONTINUE
   WRITE(6,100) (J,C(J),J=1,17)
100 FORMAT(1H , 'TIMES ** DAYS' //,17(1H ,14,6X,14//))
STOP
END
```

TIMES	**	DAYS
1		0
2		0
3		1
4		2
5		23
6		27
7		43
8		59
9		62
10		60
11		24
12		20
13		3
14		5
15		1
16		0
17		0

FORTRAN 言語で書かれた S さんのプログラムと実行結果

```
00010      DIMENSION IDAY(18),A(1)
00040      L=0
00050      DO 1 I=1,330
00060      ICOUNT=0
00070      DO 10 J=1,17
00071      CALL RANU2(L,A,1,ILL)
00080      IRAM=A(1)*100.0
00090      IF(MOD(IRAM,2).EQ.0) ICOUNT=ICOUNT+1
00100      L=L+1
00110 10    CONTINUE
00120      IDAY(ICOUNT+1)=IDAY(ICOUNT+1)+1
00130 1    CONTINUE
00131      WRITE(6,11)
00132 11    FORMAT(1H , '   カイウ   ヒコチ')
00140      DO 100 K=1,18
00150      K1=K-1
00160      WRITE(6,101)K1,IDAY(K)
00170 101   FORMAT(1H ,2I6)
00180 100   CONTINUE
00190      STOP
00200      END
```

カインウ	ヒニチ
0	0
1	0
2	0
3	2
4	7
5	15
6	30
7	58
8	60
9	54
10	58
11	28
12	10
13	5
14	3
15	0
16	0
17	0

APL言語で書かれたプログラムと実行結果

プログラム : 10 0⍎K,[1.5]÷/(K←~1+(18)∘.=+/~1+?330 17ρ2

実行結果 :	0	0
	1	0
	2	0
	3	3
	4	6
	5	18
	6	24
	7	49
	8	61
	9	71
	10	44
	11	33
	12	12
	13	8
	14	1
	15	0
	16	0
	17	0

信号停止回数 日数

では、APL言語になじみのない方々のため、プログラムについて少々解説します。APL言語で書かれた式は右から左という順で関数が実行されていきます。ただしカッコの内側でも右から左という順に実行されますが、カッコの外側の表現よりも先に実行されます。

330 17ρ2

ρは変形関数です。各要素2のマトリックス、330行17列を作ります。

技術解説

実行結果： 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
)
 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2
 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2

?330 17ρ2

?は乱数関数です。一般に? Bの結果は1からBまでの間の任意に選ばれた整数を発生します。

 ?300
 137
 ?300
 50
 ?5 7 9
 4 1 3

各要素2のマトリックス全ての要素に対して1から2までの乱数、すなわち1か2の乱数を発生させます。

実行結果： 1 2 1 2 1 1 2 2 2 1 2 2 1 1 2 2 1
 1 1 1 2 2 2 2 2 1 2 1 2 2 2 1 1 2
 1 2 2 2 1 1 2 2 2 2 1 2 2 1 1 2 1
 1 1 1 1 1 2 2 1 2 2 2 1 2 1 1 1 2
 1 2 1 1 2 2 1 2 1 2 2 2 1 1 2 2 2
)
 2 2 2 1 2 1 2 1 2 2 1 1 1 1 1 2 1
 1 1 2 2 2 1 2 2 2 1 1 1 2 1 1 2 2
 2 1 1 2 2 2 2 1 1 1 1 2 2 2 2 2 1

-1+?330 17ρ2

+は加法関数です。1か2の乱数要素マトリックスの全要素に-1をプラスします。

0 1 0 1 0 0 1 1 1 0 1 1 0 0 1 1 0
 0 0 0 1 1 1 1 1 0 1 0 1 1 1 0 0 1
 0 1 1 1 0 0 1 1 1 1 0 1 1 0 0 1 0
 0 0 0 0 0 1 1 0 1 1 1 0 1 0 0 0 1
 0 1 0 0 1 1 0 1 0 1 1 1 0 0 1 1 1
)
 1 1 1 0 1 0 1 0 1 1 0 0 0 0 0 1 0
 0 0 1 1 1 0 1 1 1 0 0 0 1 0 0 1 1
 1 0 0 1 1 1 1 0 0 0 0 1 1 1 1 1 0

技術解説

$$\begin{array}{r}
 A \leftarrow 2 \quad 3 \\
 B \leftarrow 4 \quad 5 \\
 A \circlearrowleft \cdot \times B \\
 \left. \begin{array}{r} 8 \quad 10 \\ 12 \quad 15 \end{array} \right\} \begin{array}{l} \text{右の計算を行った} \\ \text{ものと同じこと。} \end{array}
 \end{array}
 \left[\begin{array}{r} 8 \\ 10 \\ 12 \\ 15 \end{array} \right] \begin{array}{l} 2 \times 4 \\ 2 \times 5 \\ 3 \times 4 \\ 3 \times 5 \end{array}$$

=は等しい関数といわれるものです。A=BとしてAとBが等しいときは1を、AとBが等しくないときは0を示します。

$$\begin{array}{r}
 0 = 5 \\
 0 \\
 1 \\
 A \leftarrow 8 \quad 6 \quad 5 \quad 3 \\
 B \leftarrow 4 \quad 6 \quad 2 \quad 1 \\
 A = B \\
 0 \quad 1 \quad 0 \quad 0
 \end{array}$$

。、と=の組合せは次のようになります。

$$\begin{array}{r}
 A \leftarrow 3 \\
 B \leftarrow 1 \quad 2 \quad 3 \quad 4 \\
 A \circlearrowleft \cdot = B \\
 0 \quad 0 \quad 1 \quad 0 \\
 A \leftarrow 1 \quad 3 \\
 B \leftarrow 1 \quad 2 \quad 3 \quad 4 \\
 A \circlearrowleft \cdot = B \\
 1 \quad 0 \quad 0 \quad 0 \\
 0 \quad 0 \quad 1 \quad 0
 \end{array}$$

したがって (K ← -1 + 218)。、 = + / - 1 + ? 330 17 p 2はAの各要素(停止回数: 18通り)と各日(330)との間で0と1(論理数)のマトリックス(18行 330列)を作ります。

		(列)			
		第 1 日	第 2 日	第 3 日 第 330 日
停止回数	0	0	0	0 0
	1	0	0	0
	2	0	0	0
(行)	:	:	:	:	:
	8	0	0	0
	:	:	:	:	:
	10	1	0	0
	:	:	:	:	:
	17	0	1	1

+/(K←-1+∖18)∗. =+/-1+?330 17ρ2

+/-は前に述べたように18行 330列のマトリックスを各行で合計することになりますから、各停止回数に応じた日数が計算できるということです。

結果は次の通りです。

0 0 0 3 6 18 24 49 61 71 44 33 12 8 1 0 0 0

10 0?K, [1.5]+/(K←-1+∖18)∗. =+/-1+?330 17ρ2

, はベクトル化, 連結, 層連結を実行する関数です。

```

      A←1 2 3 4
      B←5 6 7 8
      A,B
1 2 3 4 5 6 7 8
      A,[0.5]B
      1 2 3 4
      5 6 7 8
      A,[1.5]B
      1 5
      2 6
      3 7
      4 8
    
```

[] 中の数字 (実数) は, (コンマ) と組合せて層連結関数として働きます。指標 (数字) が0と1の間, 1~2の間で上記例に示したように連結されます。

, は右側の変数 (マトリックス) をベクトル化する働きがあります。

```

      C←A,[0.5]B
      C
      1 2 3 4
      5 6 7 8
      D←A,[1.5]B
      D
      1 5
      2 6
      3 7
      4 8
      ,C
      1 2 3 4 5 6 7 8
      ,D
      1 5 2 6 3 7 4 8
    
```

φ は書式関数です。φ の左側の第1番目 (例では10) の数字は総桁数を決めます。第2番目の数字 (例では0又は5) は使用する精度を決めます。

```

      A←5 15.8 3 4.5
      10 0?A
           5           16           3           5
      10 5?A
      5.00000 15.80000 3.00000 4.50000
    
```

技術解説

さて、最終結果は下記の通りです。

信号停止回数	日数
0	0
1	0
2	0
3	3
4	6
5	18
6	24
7	49
8	61
9	71
10	44
11	33
12	12
13	8
14	1
15	0
16	0
17	0

最初の計算は乱数の初期値を 16807 で実行しましたが、次に乱数初期値を 5555 として実行した結果を示します。

□RL+5555

```
10 07K,[1.5]+/(K+1+(18)*.=+/-1+?330 17p2
```

0	0
1	0
2	0
3	0
4	4
5	15
6	43
7	46
8	69
9	56
10	53
11	21
12	9
13	7
14	6
15	1
16	0
17	0

FORTRANとAPLで作成したプログラムを比較してどうでしたか。FORTRANよりもAPLの方が優れているとか、あるいはAPLがより高級な言語であるとかなんてこと、さらさら申し述べるつもりはありません。それ程私、両言語に精通しているわけではありませんので。とにかく若い学生

諸君，FORTRANにこり固まらないで（失礼の段重々…）APLを少し試してみたらいかがですか。

センターのマシンにはAPL言語プロセッサがすでに導入済みとのこと，APLで処理できるように周辺機器・その他を整備することはセンターの使命ですぞ!!

