

## 第4部：『魔方陣の最新研究』： 摂田 寛二

### 第4章： 解説『魔方陣研究の諸技法』II

## 第5節： 再び「原初七方陣」と「ひな形変換法」の研究

### 0. 「完全オイラー七方陣」の新しい研究成果を反映させる

前節の研究で、汎七方陣と連立型の対称汎七方陣の全容がようやく明らかになった（「完全オイラー型七方陣」に限ってのことではあるが……）。

この成果をただちに「原初七方陣」と「ひな形変換法」の研究に反映させよう。

今までの研究では、例えば連立型の対称汎七方陣には、2つのひな形と暫定的に作った原初七方陣によって、少なくとも  $576 \times 2$  個の解があると予想された。

しかし、その後の調査で、対称汎七方陣の解は全部で 3456 個あることがわかった。

どうも我々は、最良のひな形をまだ発見していないらしい。また 3456 個もの解を持つ原初七方陣の集合をまだ作っていないらしい。

はたして別個に最良のひな形を発見できるものだろうか。またそれに合った原初七方陣を作れるものだろうか。そして、「完全オイラー型」の目的方陣の解の全数 3456 個を再現・構成することができるものであろうか。

もう一度最初から調べなおさなければならない。

### 1. 最良のひな形をさがす

ここは、例によって「再帰的な検査法」を実施するしかあるまい。解の1つ1つについて自分自身をひな形にして自己変換を試みるのである。そうして元の原初方陣に戻すことができれば、そのひな形には「可逆性」があることがわかる。可逆性がある解には、良いひな形になるものが多いことが既にわかっている。

//

```
/** Program for 'Reversibility' Test *
```

```
void revtest(){
```

```
  short n;
```

```
  for(n=1;n<50;n++){
```

```
    new1sol[n]=sol[sol[n]];
```

```
    new2sol[n]=new1sol[sol[n]];
```

```
    new3sol[n]=new2sol[sol[n]];
```

```
    ...
```

```
  }
```

```
}
```

```
//
```

```
** 再帰的な変換を三回繰り返した結果から可逆性のあるものを検索 **
```

```
* 主対角線の1本が {1, 9, 17, 25, 33, 41, 49} で構成されているものを調べた *
```

```
184/T0
```

```
/T1
```

```
/T2
```

```
/T3
```

```
 1 21 34 47 11 24 37  1 27 46 16 42 12 31  1 2 3 4 5 6 7  1 21 34 47 11 24 37
45 9 22 42 6 19 32 39 9 35 5 24 43 20  8 9 10 11 12 13 14 45 9 22 42 6 19 32
40 4 17 30 43 14 27 28 47 17 36 13 32 2 15 16 17 18 19 20 21 40 4 17 30 43 14 27
35 48 12 25 38 2 15 10 29 6 25 44 21 40 22 23 24 25 26 27 28 35 48 12 25 38 2 15
23 36 7 20 33 46 10 48 18 37 14 33 3 22 29 30 31 32 33 34 35 23 36 7 20 33 46 10
18 31 44 8 28 41 5 30 7 26 45 15 41 11 36 37 38 39 40 41 42 18 31 44 8 28 41 5
13 26 39 3 16 29 49 19 38 8 34 4 23 49 43 44 45 46 47 48 49 13 26 39 3 16 29 49
```

```
256/T0
```

```
/T1
```

```
/T2
```

```
/T3
```

```
 1 24 47 21 37 11 34  1 2 3 4 5 6 7  1 24 47 21 37 11 34  1 2 3 4 5 6 7
42 9 32 6 22 45 19  8 9 10 11 12 13 14 42 9 32 6 22 45 19  8 9 10 11 12 13 14
27 43 17 40 14 30 4 15 16 17 18 19 20 21 27 43 17 40 14 30 4 15 16 17 18 19 20 21
12 35 2 25 48 15 38 22 23 24 25 26 27 28 12 35 2 25 48 15 38 22 23 24 25 26 27 28
46 20 36 10 33 7 23 29 30 31 32 33 34 35 46 20 36 10 33 7 23 29 30 31 32 33 34 35
31 5 28 44 18 41 8 36 37 38 39 40 41 42 31 5 28 44 18 41 8 36 37 38 39 40 41 42
16 39 13 29 3 26 49 43 44 45 46 47 48 49 16 39 13 29 3 26 49 43 44 45 46 47 48 49
```

<b>315/T0</b>	<b>/T1</b>	<b>/T2</b>	<b>/T3</b>
1 27 46 16 42 12 31	1 21 34 47 11 24 37	1 2 3 4 5 6 7	1 27 46 16 42 12 31
39 9 35 5 24 43 20	45 9 22 42 6 19 32	8 9 10 11 12 13 14	39 9 35 5 24 43 20
28 47 17 36 13 32 2	40 4 17 30 43 14 27	15 16 17 18 19 20 21	28 47 17 36 13 32 2
10 29 6 25 44 21 40	35 48 12 25 38 2 15	22 23 24 25 26 27 28	10 29 6 25 44 21 40
48 18 37 14 33 3 22	23 36 7 20 33 46 10	29 30 31 32 33 34 35	48 18 37 14 33 3 22
30 7 26 45 15 41 11	18 31 44 8 28 41 5	36 37 38 39 40 41 42	30 7 26 45 15 41 11
19 38 8 34 4 23 49	13 26 39 3 16 29 49	43 44 45 46 47 48 49	19 38 8 34 4 23 49
<b>491/T0</b>	<b>/T1</b>	<b>/T2</b>	<b>/T3</b>
1 38 26 14 44 32 20	1 35 13 40 18 45 23	1 2 3 4 5 6 7	1 38 26 14 44 32 20
28 9 46 34 15 3 40	31 9 36 21 48 26 4	8 9 10 11 12 13 14	28 9 46 34 15 3 40
48 29 17 5 42 23 11	12 39 17 44 22 7 34	15 16 17 18 19 20 21	48 29 17 5 42 23 11
19 7 37 25 13 43 31	42 20 47 25 3 30 8	22 23 24 25 26 27 28	19 7 37 25 13 43 31
39 27 8 45 33 21 2	16 43 28 6 33 11 38	29 30 31 32 33 34 35	39 27 8 45 33 21 2
10 47 35 16 4 41 22	46 24 2 29 14 41 19	36 37 38 39 40 41 42	10 47 35 16 4 41 22
30 18 6 36 24 12 49	27 5 32 10 37 15 49	43 44 45 46 47 48 49	30 18 6 36 24 12 49
<b>510/T0</b>	<b>/T1</b>	<b>/T2</b>	<b>/T3</b>
1 39 28 10 48 30 19	1 21 34 47 11 24 37	1 8 15 22 29 36 43	1 27 46 16 42 12 31
27 9 47 29 18 7 38	45 9 22 42 6 19 32	2 9 16 23 30 37 44	39 9 35 5 24 43 20
46 35 17 6 37 26 8	40 4 17 30 43 14 27	3 10 17 24 31 38 45	28 47 17 36 13 32 2
16 5 36 25 14 45 34	35 48 12 25 38 2 15	4 11 18 25 32 39 46	10 29 6 25 44 21 40
42 24 13 44 33 15 4	23 36 7 20 33 46 10	5 12 19 26 33 40 47	48 18 37 14 33 3 22
12 43 32 21 3 41 23	18 31 44 8 28 41 5	6 13 20 27 34 41 48	30 7 26 45 15 41 11
31 20 2 40 22 11 49	13 26 39 3 16 29 49	7 14 21 28 35 42 49	19 38 8 34 4 23 49
<b>624/T0</b>	<b>/T1</b>	<b>/T2</b>	<b>/T3</b>
1 45 40 35 23 18 13	1 27 46 16 42 12 31	1 8 15 22 29 36 43	1 21 34 47 11 24 37
21 9 4 48 36 31 26	39 9 35 5 24 43 20	2 9 16 23 30 37 44	45 9 22 42 6 19 32
34 22 17 12 7 44 39	28 47 17 36 13 32 2	3 10 17 24 31 38 45	40 4 17 30 43 14 27
47 42 30 25 20 8 3	10 29 6 25 44 21 40	4 11 18 25 32 39 46	35 48 12 25 38 2 15
11 6 43 38 33 28 16	48 18 37 14 33 3 22	5 12 19 26 33 40 47	23 36 7 20 33 46 10
24 19 14 2 46 41 29	30 7 26 45 15 41 11	6 13 20 27 34 41 48	18 31 44 8 28 41 5
37 32 27 15 10 5 49	19 38 8 34 4 23 49	7 14 21 28 35 42 49	13 26 39 3 16 29 49
<b>672/T0</b>	<b>/T1</b>	<b>/T2</b>	<b>/T3</b>
1 46 42 31 27 16 12	1 19 30 48 10 28 39	1 2 3 4 5 6 7	1 46 42 31 27 16 12
20 9 5 43 39 35 24	47 9 27 38 7 18 29	8 9 10 11 12 13 14	20 9 5 43 39 35 24
32 28 17 13 2 47 36	37 6 17 35 46 8 26	15 16 17 18 19 20 21	32 28 17 13 2 47 36
44 40 29 25 21 10 6	34 45 14 25 36 5 16	22 23 24 25 26 27 28	44 40 29 25 21 10 6
14 3 48 37 33 22 18	24 42 4 15 33 44 13	29 30 31 32 33 34 35	14 3 48 37 33 22 18
26 15 11 7 45 41 30	21 32 43 12 23 41 3	36 37 38 39 40 41 42	26 15 11 7 45 41 30
38 34 23 19 8 4 49	11 22 40 2 20 31 49	43 44 45 46 47 48 49	38 34 23 19 8 4 49
<b>747/T0</b>	<b>/T1</b>	<b>/T2</b>	<b>/T3</b>
1 48 39 30 28 19 10	1 2 3 4 5 6 7	1 48 39 30 28 19 10	1 2 3 4 5 6 7
18 9 7 47 38 29 27	8 9 10 11 12 13 14	18 9 7 47 38 29 27	8 9 10 11 12 13 14
35 26 17 8 6 46 37	15 16 17 18 19 20 21	35 26 17 8 6 46 37	15 16 17 18 19 20 21
45 36 34 25 16 14 5	22 23 24 25 26 27 28	45 36 34 25 16 14 5	22 23 24 25 26 27 28
13 4 44 42 33 24 15	29 30 31 32 33 34 35	13 4 44 42 33 24 15	29 30 31 32 33 34 35
23 21 12 3 43 41 32	36 37 38 39 40 41 42	23 21 12 3 43 41 32	36 37 38 39 40 41 42
40 31 22 20 11 2 49	43 44 45 46 47 48 49	40 31 22 20 11 2 49	43 44 45 46 47 48 49

今までは **No. 510**, **No. 624** の 2 個のひな形しかわかっていなかったが、今回何と幾つもの別のひな形候補が見つかった。

とりわけ **No. 256**, **No. 747** の 2 個が見事だ。1 回の操作で原初方陣に戻り、2 回目でその原初方陣を変換して、自己自身を再現している。ただし、普通のリスト出力では、この 2 個は両立しない。整理に用いる不等式に抵触するからだ ( $n7 < n43$  もしくは  $n2 > n8$ )。

今回は、「最良のひな形」として最後の候補 **No. 747** を選定する。そして「ひな形変換の規則」を次のように記述する。

```
//
// * Do-it-After-the-Model Transformation *
void damt7sml(){
  d[1]=c[1]; d[2]=c[48]; d[3]=c[39]; d[4]=c[30]; d[5]=c[28]; d[6]=c[19]; d[7]=c[10];
  d[8]=c[18]; d[9]=c[9]; d[10]=c[7]; d[11]=c[47]; d[12]=c[38]; d[13]=c[29]; d[14]=c[27];
  d[15]=c[35]; d[16]=c[26]; d[17]=c[17]; d[18]=c[8]; d[19]=c[6]; d[20]=c[46]; d[21]=c[37];
  d[22]=c[45]; d[23]=c[36]; d[24]=c[34]; d[25]=c[25]; d[26]=c[16]; d[27]=c[14]; d[28]=c[5];
  d[29]=c[13]; d[30]=c[4]; d[31]=c[44]; d[32]=c[42]; d[33]=c[33]; d[34]=c[24]; d[35]=c[15];
  d[36]=c[23]; d[37]=c[21]; d[38]=c[12]; d[39]=c[3]; d[40]=c[43]; d[41]=c[41]; d[42]=c[32];
  d[43]=c[40]; d[44]=c[31]; d[45]=c[22]; d[46]=c[20]; d[47]=c[11]; d[48]=c[2]; d[49]=c[49];
}
//
```

## 2. 原初七方陣の概要

このひな形変換法の還元力は、大変に強い。他の解まで全部1回で「原初方陣」に戻ってしまう。したがって、原初方陣の概要を予め知るには、このひな形で全ての解を一度変換してみると良い。大切なヒントを、目的方陣の側からもらえる。

そして、あらためて原初七方陣の解集合を最初から構成する方法・条件を考えれば良い。

答を予め知っていてそれを利用するのは、アンフェアな同語反復だと思われるかも知れない。が、その心配は要らない。われわれが知りたいのは、解集合の間にある一対一対応の関係だ。それを確かめるのが、我々の目的だからだ。そのために対象方陣を予め良く知っておくことは、何らやましいことではない。むしろ、完璧を期するために、勧められるべき準備・予習の1つだと言える。

下が、その結果のリストだ。これは、再帰的な検査に良く似ているが、再帰するのは、最初の解(ひな形)だけで、他の例は、上の“damt7sml();”を繰り返し適用しているのみである。6例のみを掲げる。

**\*\* 対称汎七方陣の解に「ひな形変換法」を施して原初七方陣に戻す実験 \*\***  
**\* 主対角線の1本が {1, 9, 17, 25, 33, 41, 49} で構成されているものに試した \***

1/S /D7i		1/P /D7i	
1 48 39 30 28 19 10	0654321 0531642	1 2 3 4 5 6 7	0000000 0123456
18 9 7 47 38 29 27	2106543 3164205	8 9 10 11 12 13 14	1111111 0123456
35 26 17 8 6 46 37	4321065 6420531	15 16 17 18 19 20 21	2222222 0123456
45 36 34 25 16 14 5	6543210 2053164	22 23 24 25 26 27 28	3333333 0123456
13 4 44 42 33 24 15	1065432 5316420	29 30 31 32 33 34 35	4444444 0123456
23 21 12 3 43 41 32	3210654 1642053	36 37 38 39 40 41 42	5555555 0123456
40 31 22 20 11 2 49	5432106 4205316	43 44 45 46 47 48 49	6666666 0123456
2/S /D7i		2/P /D7i	
1 46 42 31 27 16 12	0654321 0362514	1 4 7 3 6 2 5	0000000 0362514
20 9 5 43 39 35 24	2106543 5140362	13 9 12 8 11 14 10	1111111 5140362
32 28 17 13 2 47 36	4321065 3625140	18 21 17 20 16 19 15	2222222 3625140
44 40 29 25 21 10 6	6543210 1403625	23 26 22 25 28 24 27	3333333 1403625
14 3 48 37 33 22 18	1065432 6251403	35 31 34 30 33 29 32	4444444 6251403
26 15 11 7 45 41 30	3210654 4036251	40 36 39 42 38 41 37	5555555 4036251
38 34 23 19 8 4 49	5432106 2514036	45 48 44 47 43 46 49	6666666 2514036
3/S /D7i		3/P /D7i	
1 45 40 35 23 18 13	0654321 0246135	1 5 2 6 3 7 4	0000000 0415263
21 9 4 48 36 31 26	2106543 6135024	12 9 13 10 14 11 8	1111111 4152630
34 22 17 12 7 44 39	4321065 5024613	16 20 17 21 18 15 19	2222222 1526304
47 42 30 25 20 8 3	6543210 4613502	27 24 28 25 22 26 23	3333333 5263041
11 6 43 38 33 28 16	1065432 3502461	31 35 32 29 33 30 34	4444444 2630415
24 19 14 2 46 41 29	3210654 2461350	42 39 36 40 37 41 38	5555555 6304152
37 32 27 15 10 5 49	5432106 1350246	46 43 47 44 48 45 49	6666666 3041526

4/S /D7i	4/P /D7i
1 42 27 12 46 31 16 0531642 0654321	1 8 15 22 29 36 43 0123456 0000000
24 9 43 35 20 5 39 3164205 2106543	2 9 16 23 30 37 44 0123456 1111111
47 32 17 2 36 28 13 6420531 4321065	3 10 17 24 31 38 45 0123456 2222222
21 6 40 25 10 44 29 2053164 6543210	4 11 18 25 32 39 46 0123456 3333333
37 22 14 48 33 18 3 5316420 1065432	5 12 19 26 33 40 47 0123456 4444444
11 45 30 15 7 41 26 1642053 3210654	6 13 20 27 34 41 48 0123456 5555555
34 19 4 38 23 8 49 4205316 5432106	7 14 21 28 35 42 49 0123456 6666666
5/S /D7i	5/P /D7i
1 39 28 10 48 30 19 0531642 0362514	1 11 21 24 34 37 47 0123456 0362514
27 9 47 29 18 7 38 3164205 5140362	6 9 19 22 32 42 45 0123456 5140362
46 35 17 6 37 26 8 6420531 3625140	4 14 17 27 30 40 43 0123456 3625140
16 5 36 25 14 45 34 2053164 1403625	2 12 15 25 35 38 48 0123456 1403625
42 24 13 44 33 15 4 5316420 6251403	7 10 20 23 33 36 46 0123456 6251403
12 43 32 21 3 41 23 1642053 4036251	5 8 18 28 31 41 44 0123456 4036251
31 20 2 40 22 11 49 4205316 2514036	3 13 16 26 29 39 49 0123456 2514036
6/S /D7i	6/P /D7i
1 38 26 14 44 32 20 0531642 0246135	1 12 16 27 31 42 46 0123456 0415263
28 9 46 34 15 3 40 3164205 6135024	5 9 20 24 35 39 43 0123456 4152630
48 29 17 5 42 23 11 6420531 5024613	2 13 17 28 32 36 47 0123456 1526304
19 7 37 25 13 43 31 2053164 4613502	6 10 21 25 29 40 44 0123456 5263041
39 27 8 45 33 21 2 5316420 3502461	3 14 18 22 33 37 48 0123456 2630415
10 47 35 16 4 41 22 1642053 2461350	7 11 15 26 30 41 45 0123456 6304152
30 18 6 36 24 12 49 4205316 1350246	4 8 19 23 34 38 49 0123456 3041526

上のリストでは、右側の方陣群が「原初七方陣」に相当する。我々が目指すのは、右側を最初に作って、それを“damt7sml();”で変換して、左側の目的方陣を作ることだ。それが普通にできれば、原初方陣と目的方陣の間に一対一対応の関係があることを示せる。

### 3. 原初七方陣の構成法

では、原初七方陣構成のための定義式・条件式は、どのように立てれば良いだろうか。

上のリストをざっと見ても、例えば、「クロスサムの等和」や「内項の和と外項の和の等和」の条件が、部分的にならともかく、全ての場合で成り立つわけではないことに気付く。汎対角線七方陣の1タイプなのに必ずしも成り立たない。偶数次とは、様子がだいぶ違う。

**\*\* 原初方陣と目的方陣の第一基本形：基準配列とひな形 \*\***

[原初七方陣]

[対称汎七方陣]

5 6 7	1   2   3   4   5   6   7	1 2 3	28 19 10	1   48   39   30   28   19   10	1 48 39
12 13 14	8   9   10   11   12   13   14	8 9 10	38 29 27	18   9   7   47   38   29   27	18 9 7
19 20 21	15   16   17   18   19   20   21	15 16 17	6 46 37	35   26   17   8   6   46   37	35 26 17
26 27 28	22   23   24   25   26   27   28	22 23 24	16 14 5	45   36   34   25   16   14   5	45 36 34
33 34 35	29   30   31   32   33   34   35	29 30 31	33 24 15	13   4   44   42   33   24   15	13 4 44
40 41 42	36   37   38   39   40   41   42	36 37 38	43 41 32	23   21   12   3   43   41   32	23 21 12
47 48 49	43   44   45   46   47   48   49	43 44 45	11 2 49	40   31   22   20   11   2   49	40 31 22

50の補数の配置は、当然と言えば当然だが、

$$1 + 49 = 2 + 48 = 3 + 47 = 4 + 46 = 5 + 45 = 6 + 44 = 7 + 43 = \dots$$

$$= 18 + 32 = 19 + 31 = 20 + 30 = 21 + 29 = 22 + 28 = 23 + 27 = 24 + 26 = 25 + 25 = 50$$

が、相対的に点对称の位置に置かれている。これらは、全ての場合で成り立つ。

この他にも、全ての場合で成り立つ条件は何かを探し出さなければならない。

これについては、四次や五次などの低次の原初方陣の研究を通じて、対称汎七方陣の原初方陣には、下のような定義式を与えるのが良いと次第にわかってきた。

**\* 行列の定和 \***

$n1+n48+n39+n30+n28+n19+n10=C$	... rw1		$n1+n18+n35+n45+n13+n23+n40=C$	... cl1
$n18+n9+n7+n47+n38+n29+n27=C$	... rw2		$n48+n9+n26+n36+n4+n21+n31=C$	... cl2
$n35+n26+n17+n8+n6+n46+n37=C$	... rw3		$n39+n7+n17+n34+n44+n12+n22=C$	... cl3
$n45+n36+n34+n25+n16+n14+n5=C$	... rw4		$n30+n47+n8+n25+n42+n3+n20=C$	... cl4
$n13+n4+n44+n42+n33+n24+n15=C$	... rw5		$n28+n38+n6+n16+n33+n43+n11=C$	... cl5
$n23+n21+n12+n3+n43+n41+n32=C$	... rw6		$n19+n29+n46+n14+n24+n41+n2=C$	... cl6
$n40+n31+n22+n20+n11+n2+n49=C$	... rw7		$n10+n27+n37+n5+n15+n32+n49=C$	... cl7

**\* 補数の定和と対称配置 \***

$$\begin{aligned}
 &n1+n49=n2+n48=n3+n47=n4+n46=n5+n45=n6+n44=n7+n43 \\
 &=n8+n42=n9+n41=n10+n40=n11+n39=n12+n38=n13+n37=n14+n36 \\
 &=n15+n35=n16+n34=n17+n33=n18+n32=n19+n31=n20+n30=n21+n29 \\
 &=n22+n28=n23+n27=n24+n26=n25+n25=CC \quad \dots \text{ scc}
 \end{aligned}$$

**\* 汎対角線の定和 \***

$n1+n9+n17+n25+n33+n41+n49=C$	... pd1		$n1+n27+n46+n16+n42+n12+n31=C$	... pq1
$n48+n7+n8+n16+n24+n32+n40=C$	... pd2		$n48+n18+n37+n14+n33+n3+n22=C$	... pq2
$n39+n47+n6+n14+n15+n23+n31=C$	... pd3		$n39+n9+n35+n5+n24+n43+n20=C$	... pq3
$n30+n38+n46+n5+n13+n21+n22=C$	... pd4		$n30+n7+n26+n45+n15+n41+n11=C$	... pq4
$n28+n29+n37+n45+n4+n12+n20=C$	... pd5		$n28+n47+n17+n36+n13+n32+n2=C$	... pq5
$n19+n27+n35+n36+n44+n3+n11=C$	... pd6		$n19+n38+n8+n34+n4+n23+n49=C$	... pq6
$n10+n18+n26+n34+n42+n43+n2=C$	... pd7		$n10+n29+n6+n25+n44+n21+n40=C$	... pq7

だが、困ったことに、これだけでは遅い。どうにも非効率なのである。上の条件式だけで普通に原初方陣を構成しても、解が中々出て来ない。いくら待ってもラチがあかない。「遅くて悪名高い七方陣だから仕方無い」と言っても、とても我慢のしようが無い。

「クロスサム等和」や「内項と外項の等和」条件を定義式に加えれば、高速にはなる。だが、それではごく少数の原初方陣しか構成できない。この2つは、今回は使用しない。

**\* リスト1：原初七方陣とひな形変換法によって再構成した対称汎七方陣のリスト（部分） \***

1/P /S1							2/P /S2																				
1	2	3	4	5	6	7	1	48	39	30	28	19	10	1	2	3	4	5	6	7	1	48	39	16	28	33	10
8	9	10	11	12	13	14	18	9	7	47	38	29	27	8	9	10	11	12	13	14	32	9	7	47	38	15	27
15	16	17	18	19	20	21	35	26	17	8	6	46	37	29	30	31	32	33	34	35	21	26	31	8	6	46	37
22	23	24	25	26	27	28	45	36	34	25	16	14	5	22	23	24	25	26	27	28	45	36	20	25	30	14	5
29	30	31	32	33	34	35	13	4	44	42	33	24	15	15	16	17	18	19	20	21	13	4	44	42	19	24	29
36	37	38	39	40	41	42	23	21	12	3	43	41	32	36	37	38	39	40	41	42	23	35	12	3	43	41	18
43	44	45	46	47	48	49	40	31	22	20	11	2	49	43	44	45	46	47	48	49	40	17	22	34	11	2	49
3/P /S1							4/P /S2																				
1	2	3	4	5	6	7	1	48	32	37	28	12	17	1	2	3	4	5	6	7	1	48	32	9	28	40	17
15	16	17	18	19	20	21	11	16	7	47	31	36	27	15	16	17	18	19	20	21	39	16	7	47	31	8	27
8	9	10	11	12	13	14	42	26	10	15	6	46	30	36	37	38	39	40	41	42	14	26	38	15	6	46	30
22	23	24	25	26	27	28	45	29	41	25	9	21	5	22	23	24	25	26	27	28	45	29	13	25	37	21	5
36	37	38	39	40	41	42	20	4	44	35	40	24	8	8	9	10	11	12	13	14	20	4	44	35	12	24	36
29	30	31	32	33	34	35	23	14	19	3	43	34	39	29	30	31	32	33	34	35	23	42	19	3	43	34	11
43	44	45	46	47	48	49	33	38	22	13	18	2	49	43	44	45	46	47	48	49	33	10	22	41	18	2	49
5/P /S1							6/P /S2																				
1	2	3	4	5	6	7	1	48	18	37	28	12	31	1	2	3	4	5	6	7	1	48	18	9	28	40	31
29	30	31	32	33	34	35	11	30	7	47	17	36	27	29	30	31	32	33	34	35	39	30	7	47	17	8	27
8	9	10	11	12	13	14	42	26	10	29	6	46	16	36	37	38	39	40	41	42	14	26	38	29	6	46	16
22	23	24	25	26	27	28	45	15	41	25	9	35	5	22	23	24	25	26	27	28	45	15	13	25	37	35	5
36	37	38	39	40	41	42	34	4	44	21	40	24	8	8	9	10	11	12	13	14	34	4	44	21	12	24	36
15	16	17	18	19	20	21	23	14	33	3	43	20	39	15	16	17	18	19	20	21	23	42	33	3	43	20	11
43	44	45	46	47	48	49	19	38	22	13	32	2	49	43	44	45	46	47	48	49	19	10	22	41	32	2	49

.....

例えば、上のリスト。これを出力するのに何十分もかかる。しかも、この後が遅い。

綺麗な出力を諦めるしかない。高速版を作って別に試したが、やはり、1万個数えるのに数時間もかかった。全部で何千時間かかるのか、見当も付かない。

だが、考えてみれば、無理もない。元々の目的方陣をストレートに作っても、時間がかかるのである。何時間もかけて何万個数えても、全然先が見えて来なかった経験がある。

他に何か良い工夫は無いものだろうか。何年もの間いろいろ実験し試行錯誤・悪戦苦闘したあげく、最後の最後にひらめきの恩恵があった。

対象を「完全オイラー方陣」に絞ることにして、いっそのこと原初方陣を「新オイラー法」で作ってみてはどうか。完全オイラー方陣なら速いのではないか。

新オイラー法では、上位・下位2面の七進法分解図を使う。最初から分解図を設計・構成する。単位面を幾つか作って、任意の2面を組み合わせる、計算で方陣を合成する。

そして、いつもの定和条件の代わりに {0, 1, 2, 3, 4, 5, 6} を各1度ずつ使うという例のラテン条件を持たせる。「オイラー方陣」として設計・構成するのである。

[原初七方陣]	[七進法分解図]
47 48 49 43 44 45 46 47 48 49 43 44 45	6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 6 4 5 6 0 1 2 3 4 5 6 0 1 2
5 6 7   1   2   3   4   5   6   7   1 2 3 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 4 5 6   0 1 2 3 4 5 6 0 1 2
12 13 14   8   9   10   11   12   13   14   8 9 10 1 1 1 1	1 1 1 1 1 1 1 1 1 1 1 4 5 6   0 1 2 3 4 5 6 0 1 2
19 20 21   15   16   17   18   19   20   21   15 16 17 2 2 2 2	2 2 2 2 2 2 2 2 2 2 2 4 5 6   0 1 2 3 4 5 6 0 1 2
26 27 28   22   23   24   25   26   27   28   22 23 24 3 3 3 3	3 3 3 3 3 3 3 3 3 3 3 4 5 6   0 1 2 3 4 5 6 0 1 2
33 34 35   29   30   31   32   33   34   35   29 30 31 4 4 4 4	4 4 4 4 4 4 4 4 4 4 4 4 5 6   0 1 2 3 4 5 6 0 1 2
40 41 42   36   37   38   39   40   41   42   36 37 38 5 5 5 5	5 5 5 5 5 5 5 5 5 5 5 4 5 6   0 1 2 3 4 5 6 0 1 2
47 48 49   43   44   45   46   47   48   49   43 44 45 6 6 6 6	6 6 6 6 6 6 6 6 6 6 6 4 5 6   0 1 2 3 4 5 6 0 1 2
5 6 7 1 2 3 4 5 6 7 1 2 3 0 0 0 0	0 0 0 0 0 0 0 0 0 0 0 0 4 5 6 0 1 2 3 4 5 6 0 1 2
	[上位] [下位]

[ひな形]	[七進法分解図]
11 2 49 40 31 22 20 11 2 49 40 31 22	1 0 6 5 4 3 2 1 0 6 5 4 3 3 1 6 4 2 0 5 3 1 6 4 2 0
28 19 10   1   48   39   30   28   19   10   1 48 39 3 2 1 0	0 6 5 4 3 2 1 0 0 6 5 6 4 2   0 5 3 1 6 4 2 0 5 3
38 29 27   18   9   7   47   38   29   27   18 9 7 5 4 3 2	2 1 0 6 5 4 3 2 2 1 0 2 0 5   3 1 6 4 2 0 5 3 1 6
6 46 37   35   26   17   8   6   46   37   35 26 17 0 6 5 4	4 3 2 1 0 6 5 4 4 3 2 5 3 1   6 4 2 0 5 3 1 6 4 2
16 14 5   45   36   34   25   16   14   5   45 36 34 2 1 0 6	5 4 3 2 1 0 6 5 6 5 4 1 6 4   2 0 5 3 1 6 4 2 0 5
33 24 15   13   4   44   42   33   24   15   13 4 44 4 3 2 1	1 0 6 5 4 3 2 1 1 0 6 4 2 0   5 3 1 6 4 2 0 5 3 1
43 41 32   23   21   12   3   43   41   32   23 21 12 6 5 4 3	3 2 1 0 6 5 4 3 3 2 1 0 5 3   1 6 4 2 0 5 3 1 6 4
11 2 49   40   31   22   20   11   2   49   40 31 22 1 0 6 5	5 4 3 2 1 0 6 5 5 4 3 3 1 6   4 2 0 5 3 1 6 4 2 0
28 19 10 1 48 39 30 28 19 10 1 48 39 3 2 1 0	6 5 4 3 2 1 0 6 5 6 4 2 0 5 3 1 6 4 2 0 5 3
	[上位] [下位]

\* 行列の定和 \*

n1+n48+n39+n30+n28+n19+n10=21 ... rw1		n1+n18+n35+n45+n13+n23+n40=21 ... cl1
n18+n9+n7+n47+n38+n29+n27=21 ... rw2		n48+n9+n26+n36+n4+n21+n31=21 ... cl2
n35+n26+n17+n8+n6+n46+n37=21 ... rw3		n39+n7+n17+n34+n44+n12+n22=21 ... cl3
n45+n36+n34+n25+n16+n14+n5=21 ... rw4		n30+n47+n8+n25+n42+n3+n20=21 ... cl4
n13+n4+n44+n42+n33+n24+n15=21 ... rw5		n28+n38+n6+n16+n33+n43+n11=21 ... cl5
n23+n21+n12+n3+n43+n41+n32=21 ... rw6		n19+n29+n46+n14+n24+n41+n2=21 ... cl6
n40+n31+n22+n20+n11+n2+n49=21 ... rw7		n10+n27+n37+n5+n15+n32+n49=21 ... cl7

**\* 補数の定和と対称配置 \***

$n1+n49=n2+n48=n3+n47=n4+n46=n5+n45=n6+n44=n7+n43$   
 $=n8+n42=n9+n41=n10+n40=n11+n39=n12+n38=n13+n37=n14+n36$   
 $=n15+n35=n16+n34=n17+n33=n18+n32=n19+n31=n20+n30=n21+n29$   
 $=n22+n28=n23+n27=n24+n26=n25+n25=6 \quad \dots \text{ scc}$

**\* 汎対角線の定和 \***

$n1+n9+n17+n25+n33+n41+n49=21$	...	pd1		$n1+n27+n46+n16+n42+n12+n31=21$	...	pq1
$n48+n7+n8+n16+n24+n32+n40=21$	...	pd2		$n48+n18+n37+n14+n33+n3+n22=21$	...	pq2
$n39+n47+n6+n14+n15+n23+n31=21$	...	pd3		$n39+n9+n35+n5+n24+n43+n20=21$	...	pq3
$n30+n38+n46+n5+n13+n21+n22=21$	...	pd4		$n30+n7+n26+n45+n15+n41+n11=21$	...	pq4
$n28+n29+n37+n45+n4+n12+n20=21$	...	pd5		$n28+n47+n17+n36+n13+n32+n2=21$	...	pq5
$n19+n27+n35+n36+n44+n3+n11=21$	...	pd6		$n19+n38+n8+n34+n4+n23+n49=21$	...	pq6
$n10+n18+n26+n34+n42+n43+n2=21$	...	pd7		$n10+n29+n6+n25+n44+n21+n40=21$	...	pq7

問題は、原初方阵をこの方法で実際に作れるかどうかだ。原初方阵は、最初から行列の定和を仮定しない。つまり、厳密な意味では魔方陣ではない。「ラテン方阵」でもない。はたして「オイラー方阵」でないものを、「新オイラー法」で作れるものか。

しかし、ひな形変換した後では、必ず完全オイラー方阵になる。一対一対応するのである。だから、原初方阵を「新オイラー法」で作るのは、理にかなっている。自然である。

ここは、実際にやってみるしか無いと、思った。

**4. 「新オイラー法」で、原初方阵を作る**

乾坤一擲の思いで組んだ小生の実際のプログラムを以下に見ていただこう。

内部表現は、古典的表記と違い、各数値から 1 を引き、0~6 の 7 個の数字を使う。

行列・汎対角線の七項定和をチェックするのに、{0, 1, 2, 3, 4, 5, 6} を各 1 度ずつ使うというラテン方阵の条件を検査する。上位面でも下位面でもそれが成り立てば、 $(0+1+2+3+4+5+6) \times 7 + (0+1+2+3+4+5+6) = 21 \times 7 + 21 = 168 (+)$  で定和になるからである。

この七項定和の 168 は、古典的表記法では魔方陣定数の 175 に相当する。

どの変数も、 $rw[i][j]$ ,  $cl[i][j]$ ,  $pd[i][j]$ ,  $pb[i][j]$  の 4 方向から定義する。

標準解のリスト出力用不等式は、ただ  $n1 < n49$ ,  $n1 < n10$ ,  $n1 < n40$  and  $n18 < n48$ ; とした。

```

/** 'Prototype Squares' of Order 7 made by the 'New Euler's Method' **
/** and Simultaneous MS77: Both Self-complementary and Pan-diagonal; **
/** Reconstructed by the 'DAM Transformation' **
/** File: 'CESPT7Sml.c' Dictated by Kanji Setsuda on **
/** July 19, 2005 and Oct. 1, 2006; Recompiled on **
/** Feb.18, 2015 with MacOSX 10.10.2 & Xcode 6.1.1; **
//
#include <stdio.h>
//
short int cnt, bcnt;
short lcnt, hcnt, cnt2, cnt3;
short CC, LSM;
short u1,u2;
short nm[50], flg[50];
short c[50], d[50];
short anm[5][53];
short pt[3457][53];
char thlu[25][50], tllu[193][50];
char mtc[25][193];
//
short rw[8][7], cl[8][7];
short pd[8][7], pb[8][7];
//
void stp01(void), stp02(void), stp03(void), stp04(void);
void stp05(void), stp06(void), stp07(void), stp08(void);
void stp09(void), stp10(void), stp11(void), stp12(void);
void stp13(void), stp14(void), stp15(void), stp16(void);

```

```

void stp17(void), stp18(void), stp19(void), stp20(void);
void stp21(void), stp22(void), stp23(void), stp24(void);
void ansrecord(void);
void prlunit(void);
void cmbcmp(void);
void prans(void), pr2ans(void);
void damt7(void);
void ptsort(short x), excpt(short x);
//
/* Main Program *
int main(){
short m,n;
printf("\n");
printf("** Compose 'Prototype Squares' of Order 7 by the 'New Euler's Method', **\n");
printf("** Transform them into Simultaneous MS77: Self-complementary and Pan-diagonal **\n");
printf("** by the 'DAM Transformation' and List out the Standard Solutions(Part) **\n");
printf("\n");
for(n=0;n<50;n++){nm[n]=-1;}
for(m=1;m<8;m++){
for(n=0;n<7;n++){rw[m][n]=0; cl[m][n]=0; pd[m][n]=0; pb[m][n]=0;}
}
CC=6; LSM=21; hcnt=0; cnt=0; bcnt=0; cnt3=0;
nm[25]=3;
rw[4][3]=1; cl[4][3]=1; pd[1][3]=1; pb[7][3]=1;
stp01();
lcnt=cnt;
prlunit();
printf("\n");
printf(" [Compact List of Solutions]\n");
printf("** 'Prototype Squares' of Order 7 and 'Simultaneous' MS77: Both SC and PD **\n");
cnt=0; cnt3=0;
cmbcmp();
ptsort(cnt);
printf(" [Count = %d] OK!\n",cnt);
printf("** Recomplied and Listed by Kanji Setsuda on\n");
printf(" Feb.18, 2015 with MacOSX 10.10.2 & Xcode 6.1.1 **\n");
printf("\n");
return 0;
}
//
/* Level #1: *
/* Set n1{1,1,1,1} & n49{7,7,1,6} *
void stp01(){
short p1=1, p2=1, p3=1, p4=1;
short q1=7, q2=7, q3=1, q4=6;
short a,b;
for(a=0;a<7;a++){b=CC-a;
if((rw[p1][a]==0)&&(cl[p2][a]==0)&&(pd[p3][a]==0)&&(pb[p4][a]==0)){
if((rw[q1][b]==0)&&(cl[q2][b]==0)&&(pd[q3][b]==0)&&(pb[q4][b]==0)){
nm[1]=a; nm[49]=b;
rw[p1][a]=1; cl[p2][a]=1; pd[p3][a]=1; pb[p4][a]=1;
rw[q1][b]=1; cl[q2][b]=1; pd[q3][b]=1; pb[q4][b]=1;
stp02();
rw[p1][a]=0; cl[p2][a]=0; pd[p3][a]=0; pb[p4][a]=0;
rw[q1][b]=0; cl[q2][b]=0; pd[q3][b]=0; pb[q4][b]=0;
}}}}
}
/* Set n48{1,2,2,2} & n2{7,6,7,5} *
void stp02(){
short p1=1, p2=2, p3=2, p4=2;
short q1=7, q2=6, q3=7, q4=5;
short a,b;
for(a=6;a>=0;a--){b=CC-a;
if((rw[p1][a]==0)&&(cl[p2][a]==0)&&(pd[p3][a]==0)&&(pb[p4][a]==0)){
if((rw[q1][b]==0)&&(cl[q2][b]==0)&&(pd[q3][b]==0)&&(pb[q4][b]==0)){

```



```

        nm[48]=a; nm[2]=b;
        rw[p1][a]=1; cl[p2][a]=1; pd[p3][a]=1; pb[p4][a]=1;
        rw[q1][b]=1; cl[q2][b]=1; pd[q3][b]=1; pb[q4][b]=1;
        stp03();
        rw[p1][a]=0; cl[p2][a]=0; pd[p3][a]=0; pb[p4][a]=0;
        rw[q1][b]=0; cl[q2][b]=0; pd[q3][b]=0; pb[q4][b]=0;
    }}}
}
/* Set n39{1,3,3,3} & n11{7,5,6,4} *
void stp03(){
    short p1=1, p2=3, p3=3, p4=3;
    short q1=7, q2=5, q3=6, q4=4;
    short a,b;
    for(a=6;a>=0;a--){b=CC-a;
        if((rw[p1][a]==0)&&(cl[p2][a]==0)&&(pd[p3][a]==0)&&(pb[p4][a]==0)){
            if((rw[q1][b]==0)&&(cl[q2][b]==0)&&(pd[q3][b]==0)&&(pb[q4][b]==0)){
                nm[39]=a; nm[11]=b;
                rw[p1][a]=1; cl[p2][a]=1; pd[p3][a]=1; pb[p4][a]=1;
                rw[q1][b]=1; cl[q2][b]=1; pd[q3][b]=1; pb[q4][b]=1;
                stp04();
                rw[p1][a]=0; cl[p2][a]=0; pd[p3][a]=0; pb[p4][a]=0;
                rw[q1][b]=0; cl[q2][b]=0; pd[q3][b]=0; pb[q4][b]=0;
            }}}
}
/* Set n30{1,4,4,4} & n20{7,4,5,3} *
void stp04(){
    short p1=1, p2=4, p3=4, p4=4;
    short q1=7, q2=4, q3=5, q4=3;
    short a,b;
    for(a=6;a>=0;a--){b=CC-a;
        if((rw[p1][a]==0)&&(cl[p2][a]==0)&&(pd[p3][a]==0)&&(pb[p4][a]==0)){
            if((rw[q1][b]==0)&&(cl[q2][b]==0)&&(pd[q3][b]==0)&&(pb[q4][b]==0)){
                nm[30]=a; nm[20]=b;
                rw[p1][a]=1; cl[p2][a]=1; pd[p3][a]=1; pb[p4][a]=1;
                rw[q1][b]=1; cl[q2][b]=1; pd[q3][b]=1; pb[q4][b]=1;
                stp05();
                rw[p1][a]=0; cl[p2][a]=0; pd[p3][a]=0; pb[p4][a]=0;
                rw[q1][b]=0; cl[q2][b]=0; pd[q3][b]=0; pb[q4][b]=0;
            }}}
}
/* Set n28{1,5,5,5} & n22{7,3,4,2} *
void stp05(){
    short p1=1, p2=5, p3=5, p4=5;
    short q1=7, q2=3, q3=4, q4=2;
    short a,b;
    for(a=6;a>=0;a--){b=CC-a;
        if((rw[p1][a]==0)&&(cl[p2][a]==0)&&(pd[p3][a]==0)&&(pb[p4][a]==0)){
            if((rw[q1][b]==0)&&(cl[q2][b]==0)&&(pd[q3][b]==0)&&(pb[q4][b]==0)){
                nm[28]=a; nm[22]=b;
                rw[p1][a]=1; cl[p2][a]=1; pd[p3][a]=1; pb[p4][a]=1;
                rw[q1][b]=1; cl[q2][b]=1; pd[q3][b]=1; pb[q4][b]=1;
                stp06();
                rw[p1][a]=0; cl[p2][a]=0; pd[p3][a]=0; pb[p4][a]=0;
                rw[q1][b]=0; cl[q2][b]=0; pd[q3][b]=0; pb[q4][b]=0;
            }}}
}
/* Set n19{1,6,6,6} & n31{7,2,3,1} *
void stp06(){
    short p1=1, p2=6, p3=6, p4=6;
    short q1=7, q2=2, q3=3, q4=1;
    short a,b;
    for(a=6;a>=0;a--){b=CC-a;
        if((rw[p1][a]==0)&&(cl[p2][a]==0)&&(pd[p3][a]==0)&&(pb[p4][a]==0)){
            if((rw[q1][b]==0)&&(cl[q2][b]==0)&&(pd[q3][b]==0)&&(pb[q4][b]==0)){
                nm[19]=a; nm[31]=b;

```

```

        rw[p1][a]=1; cl[p2][a]=1; pd[p3][a]=1; pb[p4][a]=1;
        rw[q1][b]=1; cl[q2][b]=1; pd[q3][b]=1; pb[q4][b]=1;
        stp07();
        rw[p1][a]=0; cl[p2][a]=0; pd[p3][a]=0; pb[p4][a]=0;
        rw[q1][b]=0; cl[q2][b]=0; pd[q3][b]=0; pb[q4][b]=0;
    }}}
}
/** Set n10{1,7,7,7} & n40{7,1,2,7} *
void stp07(){
    short p1=1, p2=7, p3=7, p4=7;
    short q1=7, q2=1, q3=2, q4=7;
    short a,b;
    for(a=0;a<7;a++){b=CC-a;
        if((rw[p1][a]==0)&&(cl[p2][a]==0)&&(pd[p3][a]==0)&&(pb[p4][a]==0)){
            if((rw[q1][b]==0)&&(cl[q2][b]==0)&&(pd[q3][b]==0)&&(pb[q4][b]==0)){
                nm[10]=a; nm[40]=b;
                rw[p1][a]=1; cl[p2][a]=1; pd[p3][a]=1; pb[p4][a]=1;
                rw[q1][b]=1; cl[q2][b]=1; pd[q3][b]=1; pb[q4][b]=1;
                stp08();
                rw[p1][a]=0; cl[p2][a]=0; pd[p3][a]=0; pb[p4][a]=0;
                rw[q1][b]=0; cl[q2][b]=0; pd[q3][b]=0; pb[q4][b]=0;
            }}}
    }
/** Level #2: *
/** Set n18{2,1,7,2} & n32{6,7,2,5} *
void stp08(){
    short p1=2, p2=1, p3=7, p4=2;
    short q1=6, q2=7, q3=2, q4=5;
    short a,b;
    for(a=0;a<7;a++){b=CC-a;
        if((rw[p1][a]==0)&&(cl[p2][a]==0)&&(pd[p3][a]==0)&&(pb[p4][a]==0)){
            if((rw[q1][b]==0)&&(cl[q2][b]==0)&&(pd[q3][b]==0)&&(pb[q4][b]==0)){
                nm[18]=a; nm[32]=b;
                rw[p1][a]=1; cl[p2][a]=1; pd[p3][a]=1; pb[p4][a]=1;
                rw[q1][b]=1; cl[q2][b]=1; pd[q3][b]=1; pb[q4][b]=1;
                stp09();
                rw[p1][a]=0; cl[p2][a]=0; pd[p3][a]=0; pb[p4][a]=0;
                rw[q1][b]=0; cl[q2][b]=0; pd[q3][b]=0; pb[q4][b]=0;
            }}}
    }
/** Set n9{2,2,1,3} & n41{6,6,1,4} *
void stp09(){
    short p1=2, p2=2, p3=1, p4=3;
    short q1=6, q2=6, q3=1, q4=4;
    short a,b;
    for(a=0;a<7;a++){b=CC-a;
        if((rw[p1][a]==0)&&(cl[p2][a]==0)&&(pd[p3][a]==0)&&(pb[p4][a]==0)){
            if((rw[q1][b]==0)&&(cl[q2][b]==0)&&(pd[q3][b]==0)&&(pb[q4][b]==0)){
                nm[9]=a; nm[41]=b;
                rw[p1][a]=1; cl[p2][a]=1; pd[p3][a]=1; pb[p4][a]=1;
                rw[q1][b]=1; cl[q2][b]=1; pd[q3][b]=1; pb[q4][b]=1;
                stp10();
                rw[p1][a]=0; cl[p2][a]=0; pd[p3][a]=0; pb[p4][a]=0;
                rw[q1][b]=0; cl[q2][b]=0; pd[q3][b]=0; pb[q4][b]=0;
            }}}
    }
/** Set n7{2,3,2,4} & n43{6,5,7,3} *
void stp10(){
    short p1=2, p2=3, p3=2, p4=4;
    short q1=6, q2=5, q3=7, q4=3;
    short a,b;
    for(a=0;a<7;a++){b=CC-a;
        if((rw[p1][a]==0)&&(cl[p2][a]==0)&&(pd[p3][a]==0)&&(pb[p4][a]==0)){
            if((rw[q1][b]==0)&&(cl[q2][b]==0)&&(pd[q3][b]==0)&&(pb[q4][b]==0)){
                nm[7]=a; nm[43]=b;

```

```

        rw[p1][a]=1; cl[p2][a]=1; pd[p3][a]=1; pb[p4][a]=1;
        rw[q1][b]=1; cl[q2][b]=1; pd[q3][b]=1; pb[q4][b]=1;
        stp11();
        rw[p1][a]=0; cl[p2][a]=0; pd[p3][a]=0; pb[p4][a]=0;
        rw[q1][b]=0; cl[q2][b]=0; pd[q3][b]=0; pb[q4][b]=0;
    }}}
}
/* Set n47{2,4,3,5} & n3{6,4,6,2} *
void stp11(){
    short p1=2, p2=4, p3=3, p4=5;
    short q1=6, q2=4, q3=6, q4=2;
    short a,b;
    for(a=6;a>=0;a--){b=CC-a;
        if((rw[p1][a]==0)&&(cl[p2][a]==0)&&(pd[p3][a]==0)&&(pb[p4][a]==0)){
            if((rw[q1][b]==0)&&(cl[q2][b]==0)&&(pd[q3][b]==0)&&(pb[q4][b]==0)){
                nm[47]=a; nm[3]=b;
                rw[p1][a]=1; cl[p2][a]=1; pd[p3][a]=1; pb[p4][a]=1;
                rw[q1][b]=1; cl[q2][b]=1; pd[q3][b]=1; pb[q4][b]=1;
                stp12();
                rw[p1][a]=0; cl[p2][a]=0; pd[p3][a]=0; pb[p4][a]=0;
                rw[q1][b]=0; cl[q2][b]=0; pd[q3][b]=0; pb[q4][b]=0;
            }}}
}
/* Set n38{2,5,4,6} & n12{6,3,5,1} *
void stp12(){
    short p1=2, p2=5, p3=4, p4=6;
    short q1=6, q2=3, q3=5, q4=1;
    short a,b;
    for(a=6;a>=0;a--){b=CC-a;
        if((rw[p1][a]==0)&&(cl[p2][a]==0)&&(pd[p3][a]==0)&&(pb[p4][a]==0)){
            if((rw[q1][b]==0)&&(cl[q2][b]==0)&&(pd[q3][b]==0)&&(pb[q4][b]==0)){
                nm[38]=a; nm[12]=b;
                rw[p1][a]=1; cl[p2][a]=1; pd[p3][a]=1; pb[p4][a]=1;
                rw[q1][b]=1; cl[q2][b]=1; pd[q3][b]=1; pb[q4][b]=1;
                stp13();
                rw[p1][a]=0; cl[p2][a]=0; pd[p3][a]=0; pb[p4][a]=0;
                rw[q1][b]=0; cl[q2][b]=0; pd[q3][b]=0; pb[q4][b]=0;
            }}}
}
/* Set n29{2,6,5,7} & n21{6,2,4,7} *
void stp13(){
    short p1=2, p2=6, p3=5, p4=7;
    short q1=6, q2=2, q3=4, q4=7;
    short a,b;
    for(a=6;a>=0;a--){b=CC-a;
        if((rw[p1][a]==0)&&(cl[p2][a]==0)&&(pd[p3][a]==0)&&(pb[p4][a]==0)){
            if((rw[q1][b]==0)&&(cl[q2][b]==0)&&(pd[q3][b]==0)&&(pb[q4][b]==0)){
                nm[29]=a; nm[21]=b;
                rw[p1][a]=1; cl[p2][a]=1; pd[p3][a]=1; pb[p4][a]=1;
                rw[q1][b]=1; cl[q2][b]=1; pd[q3][b]=1; pb[q4][b]=1;
                stp14();
                rw[p1][a]=0; cl[p2][a]=0; pd[p3][a]=0; pb[p4][a]=0;
                rw[q1][b]=0; cl[q2][b]=0; pd[q3][b]=0; pb[q4][b]=0;
            }}}
}
/* Set n27{2,7,6,1} & n23{6,1,3,6} *
void stp14(){
    short p1=2, p2=7, p3=6, p4=1;
    short q1=6, q2=1, q3=3, q4=6;
    short a,b;
    for(a=6;a>=0;a--){b=CC-a;
        if((rw[p1][a]==0)&&(cl[p2][a]==0)&&(pd[p3][a]==0)&&(pb[p4][a]==0)){
            if((rw[q1][b]==0)&&(cl[q2][b]==0)&&(pd[q3][b]==0)&&(pb[q4][b]==0)){
                nm[27]=a; nm[23]=b;
                rw[p1][a]=1; cl[p2][a]=1; pd[p3][a]=1; pb[p4][a]=1;

```

```

        rw[q1][b]=1; cl[q2][b]=1; pd[q3][b]=1; pb[q4][b]=1;
        stp15();
        rw[p1][a]=0; cl[p2][a]=0; pd[p3][a]=0; pb[p4][a]=0;
        rw[q1][b]=0; cl[q2][b]=0; pd[q3][b]=0; pb[q4][b]=0;
    }}}
}
/* Level #3: *
/* Set n35{3,1,6,3} & n15{5,7,3,4} *
void stp15(){
    short p1=3, p2=1, p3=6, p4=3;
    short q1=5, q2=7, q3=3, q4=4;
    short a,b;
    for(a=6;a>=0;a--){b=CC-a;
        if((rw[p1][a]==0)&&(cl[p2][a]==0)&&(pd[p3][a]==0)&&(pb[p4][a]==0)){
            if((rw[q1][b]==0)&&(cl[q2][b]==0)&&(pd[q3][b]==0)&&(pb[q4][b]==0)){
                nm[35]=a; nm[15]=b;
                rw[p1][a]=1; cl[p2][a]=1; pd[p3][a]=1; pb[p4][a]=1;
                rw[q1][b]=1; cl[q2][b]=1; pd[q3][b]=1; pb[q4][b]=1;
                stp16();
                rw[p1][a]=0; cl[p2][a]=0; pd[p3][a]=0; pb[p4][a]=0;
                rw[q1][b]=0; cl[q2][b]=0; pd[q3][b]=0; pb[q4][b]=0;
            }}}
}
/* Set n26{3,2,7,4} & n24{5,6,2,3} *
void stp16(){
    short p1=3, p2=2, p3=7, p4=4;
    short q1=5, q2=6, q3=2, q4=3;
    short a,b;
    for(a=6;a>=0;a--){b=CC-a;
        if((rw[p1][a]==0)&&(cl[p2][a]==0)&&(pd[p3][a]==0)&&(pb[p4][a]==0)){
            if((rw[q1][b]==0)&&(cl[q2][b]==0)&&(pd[q3][b]==0)&&(pb[q4][b]==0)){
                nm[26]=a; nm[24]=b;
                rw[p1][a]=1; cl[p2][a]=1; pd[p3][a]=1; pb[p4][a]=1;
                rw[q1][b]=1; cl[q2][b]=1; pd[q3][b]=1; pb[q4][b]=1;
                stp17();
                rw[p1][a]=0; cl[p2][a]=0; pd[p3][a]=0; pb[p4][a]=0;
                rw[q1][b]=0; cl[q2][b]=0; pd[q3][b]=0; pb[q4][b]=0;
            }}}
}
/* Set n17{3,3,1,5} & n33{5,5,1,2} *
void stp17(){
    short p1=3, p2=3, p3=1, p4=5;
    short q1=5, q2=5, q3=1, q4=2;
    short a,b;
    for(a=0;a<7;a++){b=CC-a;
        if((rw[p1][a]==0)&&(cl[p2][a]==0)&&(pd[p3][a]==0)&&(pb[p4][a]==0)){
            if((rw[q1][b]==0)&&(cl[q2][b]==0)&&(pd[q3][b]==0)&&(pb[q4][b]==0)){
                nm[17]=a; nm[33]=b;
                rw[p1][a]=1; cl[p2][a]=1; pd[p3][a]=1; pb[p4][a]=1;
                rw[q1][b]=1; cl[q2][b]=1; pd[q3][b]=1; pb[q4][b]=1;
                stp18();
                rw[p1][a]=0; cl[p2][a]=0; pd[p3][a]=0; pb[p4][a]=0;
                rw[q1][b]=0; cl[q2][b]=0; pd[q3][b]=0; pb[q4][b]=0;
            }}}
}
/* Set n8{3,4,2,6} & n42{5,4,7,1} *
void stp18(){
    short p1=3, p2=4, p3=2, p4=6;
    short q1=5, q2=4, q3=7, q4=1;
    short a,b;
    for(a=0;a<7;a++){b=CC-a;
        if((rw[p1][a]==0)&&(cl[p2][a]==0)&&(pd[p3][a]==0)&&(pb[p4][a]==0)){
            if((rw[q1][b]==0)&&(cl[q2][b]==0)&&(pd[q3][b]==0)&&(pb[q4][b]==0)){
                nm[8]=a; nm[42]=b;
                rw[p1][a]=1; cl[p2][a]=1; pd[p3][a]=1; pb[p4][a]=1;

```

```

        rw[q1][b]=1; cl[q2][b]=1; pd[q3][b]=1; pb[q4][b]=1;
        stp19();
        rw[p1][a]=0; cl[p2][a]=0; pd[p3][a]=0; pb[p4][a]=0;
        rw[q1][b]=0; cl[q2][b]=0; pd[q3][b]=0; pb[q4][b]=0;
    }}}
}
/* Set n6{3,5,3,7} & n44{5,3,6,7} *
void stp19(){
    short p1=3, p2=5, p3=3, p4=7;
    short q1=5, q2=3, q3=6, q4=7;
    short a,b;
    for(a=0;a<7;a++){b=CC-a;
        if((rw[p1][a]==0)&&(cl[p2][a]==0)&&(pd[p3][a]==0)&&(pb[p4][a]==0)){
            if((rw[q1][b]==0)&&(cl[q2][b]==0)&&(pd[q3][b]==0)&&(pb[q4][b]==0)){
                nm[6]=a; nm[44]=b;
                rw[p1][a]=1; cl[p2][a]=1; pd[p3][a]=1; pb[p4][a]=1;
                rw[q1][b]=1; cl[q2][b]=1; pd[q3][b]=1; pb[q4][b]=1;
                stp20();
                rw[p1][a]=0; cl[p2][a]=0; pd[p3][a]=0; pb[p4][a]=0;
                rw[q1][b]=0; cl[q2][b]=0; pd[q3][b]=0; pb[q4][b]=0;
            }}}
}
/* Set n46{3,6,4,1} & n4{5,2,5,6} *
void stp20(){
    short p1=3, p2=6, p3=4, p4=1;
    short q1=5, q2=2, q3=5, q4=6;
    short a,b;
    for(a=6;a>=0;a--){b=CC-a;
        if((rw[p1][a]==0)&&(cl[p2][a]==0)&&(pd[p3][a]==0)&&(pb[p4][a]==0)){
            if((rw[q1][b]==0)&&(cl[q2][b]==0)&&(pd[q3][b]==0)&&(pb[q4][b]==0)){
                nm[46]=a; nm[4]=b;
                rw[p1][a]=1; cl[p2][a]=1; pd[p3][a]=1; pb[p4][a]=1;
                rw[q1][b]=1; cl[q2][b]=1; pd[q3][b]=1; pb[q4][b]=1;
                stp21();
                rw[p1][a]=0; cl[p2][a]=0; pd[p3][a]=0; pb[p4][a]=0;
                rw[q1][b]=0; cl[q2][b]=0; pd[q3][b]=0; pb[q4][b]=0;
            }}}
}
/* Set n37{3,7,5,2} & n13{5,1,4,5} *
void stp21(){
    short p1=3, p2=7, p3=5, p4=2;
    short q1=5, q2=1, q3=4, q4=5;
    short a,b;
    for(a=6;a>=0;a--){b=CC-a;
        if((rw[p1][a]==0)&&(cl[p2][a]==0)&&(pd[p3][a]==0)&&(pb[p4][a]==0)){
            if((rw[q1][b]==0)&&(cl[q2][b]==0)&&(pd[q3][b]==0)&&(pb[q4][b]==0)){
                nm[37]=a; nm[13]=b;
                rw[p1][a]=1; cl[p2][a]=1; pd[p3][a]=1; pb[p4][a]=1;
                rw[q1][b]=1; cl[q2][b]=1; pd[q3][b]=1; pb[q4][b]=1;
                stp22();
                rw[p1][a]=0; cl[p2][a]=0; pd[p3][a]=0; pb[p4][a]=0;
                rw[q1][b]=0; cl[q2][b]=0; pd[q3][b]=0; pb[q4][b]=0;
            }}}
}
/* Level #4: *
/* Set n45{4,1,5,4} & n5{4,7,4,3} *
void stp22(){
    short p1=4, p2=1, p3=5, p4=4;
    short q1=4, q2=7, q3=4, q4=3;
    short a,b;
    for(a=6;a>=0;a--){b=CC-a;
        if((rw[p1][a]==0)&&(cl[p2][a]==0)&&(pd[p3][a]==0)&&(pb[p4][a]==0)){
            if((rw[q1][b]==0)&&(cl[q2][b]==0)&&(pd[q3][b]==0)&&(pb[q4][b]==0)){
                nm[45]=a; nm[5]=b;
                rw[p1][a]=1; cl[p2][a]=1; pd[p3][a]=1; pb[p4][a]=1;

```

```

        rw[q1][b]=1; cl[q2][b]=1; pd[q3][b]=1; pb[q4][b]=1;
        stp23();
        rw[p1][a]=0; cl[p2][a]=0; pd[p3][a]=0; pb[p4][a]=0;
        rw[q1][b]=0; cl[q2][b]=0; pd[q3][b]=0; pb[q4][b]=0;
    }}}
}
/* Set n36{4,2,6,5} & n14{4,6,3,2} *
void stp23(){
    short p1=4, p2=2, p3=6, p4=5;
    short q1=4, q2=6, q3=3, q4=2;
    short a,b;
    for(a=6;a>=0;a--){b=CC-a;
        if((rw[p1][a]==0)&&(cl[p2][a]==0)&&(pd[p3][a]==0)&&(pb[p4][a]==0)){
            if((rw[q1][b]==0)&&(cl[q2][b]==0)&&(pd[q3][b]==0)&&(pb[q4][b]==0)){
                nm[36]=a; nm[14]=b;
                rw[p1][a]=1; cl[p2][a]=1; pd[p3][a]=1; pb[p4][a]=1;
                rw[q1][b]=1; cl[q2][b]=1; pd[q3][b]=1; pb[q4][b]=1;
                stp24();
                rw[p1][a]=0; cl[p2][a]=0; pd[p3][a]=0; pb[p4][a]=0;
                rw[q1][b]=0; cl[q2][b]=0; pd[q3][b]=0; pb[q4][b]=0;
            }}}
}
/* Set n34{4,3,7,6} & n16{4,5,2,1} *
void stp24(){
    short p1=4, p2=3, p3=7, p4=6;
    short q1=4, q2=5, q3=2, q4=1;
    short a,b;
    for(a=6;a>=0;a--){b=CC-a;
        if((rw[p1][a]==0)&&(cl[p2][a]==0)&&(pd[p3][a]==0)&&(pb[p4][a]==0)){
            if((rw[q1][b]==0)&&(cl[q2][b]==0)&&(pd[q3][b]==0)&&(pb[q4][b]==0)){
                nm[34]=a; nm[16]=b;
                rw[p1][a]=1; cl[p2][a]=1; pd[p3][a]=1; pb[p4][a]=1;
                rw[q1][b]=1; cl[q2][b]=1; pd[q3][b]=1; pb[q4][b]=1;
                ansrecord();
                rw[p1][a]=0; cl[p2][a]=0; pd[p3][a]=0; pb[p4][a]=0;
                rw[q1][b]=0; cl[q2][b]=0; pd[q3][b]=0; pb[q4][b]=0;
            }}}
}
}
//
/* Record the Answers *
void ansrecord(){
    short n;
    cnt++;
    tllu[cnt-1][0]=cnt;
    for(n=1;n<50;n++){tllu[cnt-1][n]=nm[n];}
    if((nm[1]<nm[49])&&(nm[1]<nm[10])&&(nm[1]<nm[40])&&(nm[48]>nm[18])){
        hcnt++;
        thlu[hcnt-1][0]=hcnt;
        for(n=1;n<50;n++){thlu[hcnt-1][n]=nm[n];}
    }
}
//
/* Classify the Latin Units *
void prlunit(){
    short t,l,m,n;
    printf(" [Counts of Latin Units = %dH/%dL]\n",hcnt,cnt);
    for(m=0;m<hcnt;m++){
        for(n=0;n<cnt;n++){
            t=0;
            for(l=1;l<50;l++){if(thlu[m][l]==tllu[n][l]){t++;}}
            mtc[m][n]=t;
            t=0;
            for(l=1;l<50;l++){if(thlu[m][l]+tllu[n][l]==6){t++;}}
            if(mtc[m][n]<t){mtc[m][n]=t;}
            if(mtc[m][n]==7){bcnt++;}
        }
    }
}

```

```

    }
}
}
//
/** Combine and Compose *
void cmbcmp(){
short n,d,fc;
for(u1=0;u1<hcnt;u1++){cnt2=0;
for(u2=0;u2<lcnt;u2++){
if(mtc[u1][u2]==7){
for(n=1;n<50;n++){flg[n]=0;}
for(n=1;n<50;n++){
d=thlu[u1][n]*7+tllu[u2][n]+1;
nm[n]=d; flg[d]++;
}
fc=0;
for(n=1;n<50;n++){if(flg[n]==1){fc++;}else{break;}}
if(fc==49){
if((nm[1]<nm[49])&&(nm[1]<nm[10])&&(nm[1]<nm[40])&&(nm[48]>nm[18])){
prans();}
}
}
}
}
}
//
/** Print the Answers *
void prans(){
short n;
pt[cnt][0]=cnt+1;
for(n=1;n<50;n++){pt[cnt][n]=nm[n];}
pt[cnt][51]=u1+1; pt[cnt][52]=u2+1;
cnt++; cnt2++;
if(cnt2==1){
anm[cnt3][0]=cnt;
for(n=1;n<50;n++){anm[cnt3][n]=nm[n];}
anm[cnt3][51]=u1+1; anm[cnt3][52]=u2+1;
cnt3++;
if(cnt3==2){pr2ans(); cnt3=0;}
}
}
//
/** Print 2 Answers *
void pr2ans(){
short l,l7,m,n;
for(m=0;m<2;m++){
for(n=1;n<50;n++){c[n]=anm[m][n];}
damt7();
for(n=1;n<50;n++){anm[m+2][n]=d[n];}
printf(" PT:%7d/H%6d/L Sol:%15d#",anm[m][51],anm[m][52],anm[m][0]);
if(m<1){printf(" ");}
}
printf("\n");
for(l=0;l<7;l++){l7=l*7;
for(m=0;m<2;m++){
for(n=1;n<8;n++){printf("%3d",anm[m][l7+n]);}
printf(" ");
for(n=1;n<8;n++){printf("%3d",anm[m+2][l7+n]);}
if(m<1){printf(" ");}
}
printf("\n");
}
printf("\n");
}
//

```

```

/** Do-it-After-the-Model Transformation *
void damt7(){
  d[1]=c[1];  d[2]=c[48];  d[3]=c[39];  d[4]=c[30];  d[5]=c[28];  d[6]=c[19];  d[7]=c[10];
  d[8]=c[18];  d[9]=c[9];  d[10]=c[7];  d[11]=c[47];  d[12]=c[38];  d[13]=c[29];  d[14]=c[27];
  d[15]=c[35];  d[16]=c[26];  d[17]=c[17];  d[18]=c[8];  d[19]=c[6];  d[20]=c[46];  d[21]=c[37];
  d[22]=c[45];  d[23]=c[36];  d[24]=c[34];  d[25]=c[25];  d[26]=c[16];  d[27]=c[14];  d[28]=c[5];
  d[29]=c[13];  d[30]=c[4];  d[31]=c[44];  d[32]=c[42];  d[33]=c[33];  d[34]=c[24];  d[35]=c[15];
  d[36]=c[23];  d[37]=c[21];  d[38]=c[12];  d[39]=c[3];  d[40]=c[43];  d[41]=c[41];  d[42]=c[32];
  d[43]=c[40];  d[44]=c[31];  d[45]=c[22];  d[46]=c[20];  d[47]=c[11];  d[48]=c[2];  d[49]=c[49];
}
//
/** Sort the Prototype Squares *
void ptsort(short x){
  short mx,n,f;
  int d1,d2,d3,d4;
  mx=x-1;
  do{f=0;
  for(n=0;n<mx;n++){
    d1=(((pt[n][1]*50+pt[n][2])*50+pt[n][3])*50+pt[n][4])*50+pt[n][5];
    d2=(((pt[n+1][1]*50+pt[n+1][2])*50+pt[n+1][3])*50+pt[n+1][4])*50+pt[n+1][5];
    d3=(((pt[n][8]*50+pt[n][15])*50+pt[n][22])*50+pt[n][29])*50+pt[n][36];
    d4=(((pt[n+1][8]*50+pt[n+1][15])*50+pt[n+1][22])*50+pt[n+1][29])*50+pt[n+1][36];
    if(d1>d2){excpt(n); f=1;}
    else if((d1==d2)&&(d3>d4)){excpt(n); f=1;}
  }
  mx--;
  }while(f>0);
}
//
void excpt(short x){
  short n,d;
  for(n=0;n<53;n++){d=pt[x][n]; pt[x][n]=pt[x+1][n]; pt[x+1][n]=d;}
}
//

```

やってみて、損は無かった。あつという間に原初七方陣の解を全数作り、ひな形変換法を施して連立型の対称汎七方陣（完全オイラー方陣）の解 3456個を見事に再構成した。

**\* 「新オイラー法」によって構成した七進法分解図の単位面 [上位] \***

1/	2/	3/	4/	5/	6/	7/	8/
0000000	0000000	0000000	0000000	0000000	0000000	0000000	0000000
1111111	1111111	2222222	2222222	4444444	4444444	5555555	5555555
2222222	4444444	1111111	5555555	1111111	5555555	2222222	4444444
3333333	3333333	3333333	3333333	3333333	3333333	3333333	3333333
4444444	2222222	5555555	1111111	5555555	1111111	4444444	2222222
5555555	5555555	4444444	4444444	2222222	2222222	1111111	1111111
6666666	6666666	6666666	6666666	6666666	6666666	6666666	6666666
9/	10/	11/	12/	13/	14/	15/	16/
0123456	0143256	0213546	0253146	0365241	0364152	0362154	0361245
0123456	0143256	0213546	0253146	2410365	1520364	1540362	2450361
0123456	0143256	0213546	0253146	3652410	3641520	3621540	3612450
0123456	0143256	0213546	0253146	4103652	5203641	5403621	4503612
0123456	0143256	0213546	0253146	6524103	6415203	6215403	6124503
0123456	0143256	0213546	0253146	1036524	2036415	4036215	5036124
0123456	0143256	0213546	0253146	5241036	4152036	2154036	1245036
17/	18/	19/	20/	21/	22/	23/	24/
1023465	1043265	1042653	1024653	1111111	1111111	1354062	1352064
1023465	1043265	0426531	0246531	2222222	4444444	0621354	0641352
1023465	1043265	4265310	2465310	0000000	0000000	3540621	3520641
1023465	1043265	2653104	4653102	3333333	3333333	6213540	6413520
1023465	1043265	6531042	6531024	6666666	6666666	5406213	5206413
1023465	1043265	5310426	5310246	4444444	2222222	2135406	4135206
1023465	1043265	3104265	3102465	5555555	5555555	4062135	2064135



\* 「新オイラー法」によって構成した七進法分解図の単位面 [下位] \*

1/	2/	3/	4/	5/	6/	7/	8/
000000	000000	000000	000000	000000	000000	000000	000000
111111	111111	222222	222222	444444	444444	555555	555555
222222	444444	111111	555555	111111	555555	222222	444444
333333	333333	333333	333333	333333	333333	333333	333333
444444	222222	555555	111111	555555	111111	444444	222222
555555	555555	444444	444444	222222	222222	111111	111111
666666	666666	666666	666666	666666	666666	666666	666666
33/	34/	35/	36/	37/	38/	39/	40/
1023465	1043265	1042653	1024653	1111111	1111111	1111111	1111111
1023465	1043265	0426531	0246531	0000000	0000000	2222222	2222222
1023465	1043265	4265310	2465310	2222222	4444444	0000000	6666666
1023465	1043265	2653104	4653102	3333333	3333333	3333333	3333333
1023465	1043265	6531042	6531024	4444444	2222222	6666666	0000000
1023465	1043265	5310426	5310246	6666666	6666666	4444444	4444444
1023465	1043265	3104265	3102465	5555555	5555555	5555555	5555555
65/	66/	67/	68/	69/	70/	71/	72/
2013564	2053164	2051643	2015643	2103654	2163054	2160543	2106543
2013564	2053164	0516432	0156432	2103654	2163054	1605432	1065432
2013564	2053164	5164320	1564320	2103654	2163054	6054321	0654321
2013564	2053164	1643205	5643201	2103654	2163054	0543216	6543210
2013564	2053164	6432051	6432015	2103654	2163054	5432160	5432106
2013564	2053164	4320516	4320156	2103654	2163054	4321605	4321065
2013564	2053164	3205164	3201564	2103654	2163054	3216054	3210654
97/	98/	99/	100/	101/	102/	103/	104/
4013562	4053162	4051623	4015623	4103652	4163052	4160523	4106523
4013562	4053162	0516234	0156234	4103652	4163052	1605234	1065234
4013562	4053162	5162340	1562340	4103652	4163052	6052341	0652341
4013562	4053162	1623405	5623401	4103652	4163052	0523416	6523410
4013562	4053162	6234051	6234015	4103652	4163052	5234160	5234106
4013562	4053162	2340516	2340156	4103652	4163052	2341605	2341065
4013562	4053162	3405162	3401562	4103652	4163052	3416052	3410652
129/	130/	131/	132/	133/	134/	135/	136/
5023461	5043261	5042613	5024613	5203641	5263041	5260413	5206413
5023461	5043261	0426135	0246135	5203641	5263041	2604135	2064135
5023461	5043261	4261350	2461350	5203641	5263041	6041352	0641352
5023461	5043261	2613504	4613502	5203641	5263041	0413526	6413520
5023461	5043261	6135024	6135024	5203641	5263041	4135260	4135206
5023461	5043261	1350426	1350246	5203641	5263041	1352604	1352064
5023461	5043261	3504261	3502461	5203641	5263041	3526041	3520641
161/	162/	163/	164/	165/	166/	167/	168/
6123450	6143250	6142503	6124503	6213540	6253140	6251403	6215403
6123450	6143250	1425036	1245036	6213540	6253140	2514036	2154036
6123450	6143250	4250361	2450361	6213540	6253140	5140362	1540362
6123450	6143250	2503614	4503612	6213540	6253140	1403625	5403621
6123450	6143250	5036142	5036124	6213540	6253140	4036251	4036215
6123450	6143250	0361425	0361245	6213540	6253140	0362514	0362154
6123450	6143250	3614250	3612450	6213540	6253140	3625140	3621540

[上位/下位 = 24H/192L]

\* 最良の組合せを得るための参照表 \*

l:h	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
1	49	35	21	21	21	21	35	35	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
2	35	49	21	21	21	21	35	35	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
3	21	21	49	35	35	35	21	21	7	7	7	7	7	7	7	7	7	7	7	21	21	7	7	7
4	21	21	35	49	35	35	21	21	7	7	7	7	7	7	7	7	7	7	7	21	21	7	7	7
5	21	21	35	35	49	35	21	21	7	7	7	7	7	7	7	7	7	7	7	21	21	7	7	7
6	21	21	35	35	35	49	21	21	7	7	7	7	7	7	7	7	7	7	7	21	21	7	7	7
7	35	35	21	21	21	21	49	35	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
8	35	35	21	21	21	21	35	49	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
9	7	7	7	7	7	7	7	7	49	35	21	21	7	7	7	7	21	21	7	7	7	7	7	7

10| 7 7 7 7 7 7 7 7 7 35 49 21 21 7 7 7 7 21 21 7 7 7 7 7 7  
 11| 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 21 21 7 7 7 7 7 7  
 12| 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 21 21 7 7 7 7 7 7  
 13| 7 7 7 7 7 7 7 7 7 21 21 49 35 7 7 7 7 7 7 7 7 7 7 7 7  
 14| 7 7 7 7 7 7 7 7 7 21 21 35 49 7 7 7 7 7 7 7 7 7 7 7 7  
 15| 7  
 16| 7  
 17| 7 7 7 7 7 7 7 7 7 7 7 7 7 35 21 21 35 7 7 7 7 7 7 7 7  
 18| 7 7 7 7 7 7 7 7 7 7 7 7 7 49 21 21 35 7 7 7 7 7 7 7 7  
 19| 7 7 7 7 7 7 7 7 7 7 7 7 7 21 35 35 21 7 7 7 7 7 7 21 21  
 20| 7 7 7 7 7 7 7 7 7 7 7 7 7 21 49 35 21 7 7 7 7 7 7 21 21  
 21| 7 7 7 7 7 7 7 7 7 7 7 7 7 21 35 35 21 7 7 7 7 7 7 21 21  
 22| 7 7 7 7 7 7 7 7 7 7 7 7 7 21 35 49 21 7 7 7 7 7 7 21 21  
 23| 7 7 7 7 7 7 7 7 7 7 7 7 7 35 21 21 35 7 7 7 7 7 7 7 7  
 24| 7 7 7 7 7 7 7 7 7 7 7 7 7 35 21 21 49 7 7 7 7 7 7 7 7

178| 7  
 179| 7 7 7 7 7 7 7 7 7 21 21 35 49 7 7 7 7 7 7 7 7 7 7 7 7  
 180| 7 7 7 7 7 7 7 7 7 21 21 49 35 7 7 7 7 7 7 7 7 7 7 7 7  
 181| 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 21 21 7 7 7 7 7 7  
 182| 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 21 21 7 7 7 7 7 7  
 183| 7 7 7 7 7 7 7 7 7 35 49 21 21 7 7 7 7 21 21 7 7 7 7 7 7  
 183| 7 7 7 7 7 7 7 7 7 35 49 21 21 7 7 7 7 21 21 7 7 7 7 7 7  
 184| 7 7 7 7 7 7 7 7 7 49 35 21 21 7 7 7 7 21 21 7 7 7 7 7 7  
 185| 35 35 21 21 21 21 35 49 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7  
 186| 35 35 21 21 21 21 49 35 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7  
 187| 21 21 35 35 35 49 21 21 7 7 7 7 7 7 7 7 7 7 7 7 21 21 7 7 7  
 188| 21 21 35 35 49 35 21 21 7 7 7 7 7 7 7 7 7 7 7 7 21 21 7 7 7  
 189| 21 21 35 49 35 35 21 21 7 7 7 7 7 7 7 7 7 7 7 7 21 21 7 7 7  
 190| 21 21 49 35 35 35 21 21 7 7 7 7 7 7 7 7 7 7 7 7 21 21 7 7 7  
 191| 35 49 21 21 21 21 35 35 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7  
 192| 49 35 21 21 21 21 35 35 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7 7

\* 新オイラー法によって構成した原初七方陣と  
 ひな形変換法によって再構成した対称汎七方陣の解のリスト（抜粋）\*

PT:	1/H	9/L	Sol:	1#	PT:	2/H	9/L	Sol:	145#																		
1	2	3	4	5	6	7	1	48	39	30	28	19	10	1	2	3	4	5	6	7	1	48	39	16	28	33	10
8	9	10	11	12	13	14	18	9	7	47	38	29	27	8	9	10	11	12	13	14	32	9	7	47	38	15	27
15	16	17	18	19	20	21	35	26	17	8	6	46	37	29	30	31	32	33	34	35	21	26	31	8	6	46	37
22	23	24	25	26	27	28	45	36	34	25	16	14	5	22	23	24	25	26	27	28	45	36	20	25	30	14	5
29	30	31	32	33	34	35	13	4	44	42	33	24	15	15	16	17	18	19	20	21	13	4	44	42	19	24	29
36	37	38	39	40	41	42	23	21	12	3	43	41	32	36	37	38	39	40	41	42	23	35	12	3	43	41	18
43	44	45	46	47	48	49	40	31	22	20	11	2	49	43	44	45	46	47	48	49	40	17	22	34	11	2	49

PT:	3/H	9/L	Sol:	289#	PT:	4/H	9/L	Sol:	433#																		
1	2	3	4	5	6	7	1	48	32	37	28	12	17	1	2	3	4	5	6	7	1	48	32	9	28	40	17
15	16	17	18	19	20	21	11	16	7	47	31	36	27	15	16	17	18	19	20	21	39	16	7	47	31	8	27
8	9	10	11	12	13	14	42	26	10	15	6	46	30	36	37	38	39	40	41	42	14	26	38	15	6	46	30
22	23	24	25	26	27	28	45	29	41	25	9	21	5	22	23	24	25	26	27	28	45	29	13	25	37	21	5
36	37	38	39	40	41	42	20	4	44	35	40	24	8	8	9	10	11	12	13	14	20	4	44	35	12	24	36
29	30	31	32	33	34	35	23	14	19	3	43	34	39	29	30	31	32	33	34	35	23	42	19	3	43	34	11
43	44	45	46	47	48	49	33	38	22	13	18	2	49	43	44	45	46	47	48	49	33	10	22	41	18	2	49

PT:	5/H	9/L	Sol:	577#	PT:	6/H	9/L	Sol:	721#																		
1	2	3	4	5	6	7	1	48	18	37	28	12	31	1	2	3	4	5	6	7	1	48	18	9	28	40	31
29	30	31	32	33	34	35	11	30	7	47	17	36	27	29	30	31	32	33	34	35	39	30	7	47	17	8	27
8	9	10	11	12	13	14	42	26	10	29	6	46	16	36	37	38	39	40	41	42	14	26	38	29	6	46	16
22	23	24	25	26	27	28	45	15	41	25	9	35	5	22	23	24	25	26	27	28	45	15	13	25	37	35	5
36	37	38	39	40	41	42	34	4	44	21	40	24	8	8	9	10	11	12	13	14	34	4	44	21	12	24	36
15	16	17	18	19	20	21	23	14	33	3	43	20	39	15	16	17	18	19	20	21	23	42	33	3	43	20	11
43	44	45	46	47	48	49	19	38	22	13	32	2	49	43	44	45	46	47	48	49	19	10	22	41	32	2	49

PT: 7/H 9/L	Sol: 865#	PT: 8/H 9/L	Sol: 1009#
1 2 3 4 5 6 7	1 48 11 30 28 19 38	1 2 3 4 5 6 7	1 48 11 16 28 33 38
36 37 38 39 40 41 42	18 37 7 47 10 29 27	36 37 38 39 40 41 42	32 37 7 47 10 15 27
15 16 17 18 19 20 21	35 26 17 36 6 46 9	29 30 31 32 33 34 35	21 26 31 36 6 46 9
22 23 24 25 26 27 28	45 8 34 25 16 42 5	22 23 24 25 26 27 28	45 8 20 25 30 42 5
29 30 31 32 33 34 35	41 4 44 14 33 24 15	15 16 17 18 19 20 21	41 4 44 14 19 24 29
8 9 10 11 12 13 14	23 21 40 3 43 13 32	8 9 10 11 12 13 14	23 35 40 3 43 13 18
43 44 45 46 47 48 49	12 31 22 20 39 2 49	43 44 45 46 47 48 49	12 17 22 34 39 2 49
PT: 9/H 1/L	Sol: 1153#	PT: 10/H 1/L	Sol: 1297#
1 8 15 22 29 36 43	1 42 27 12 46 31 16	1 8 29 22 15 36 43	1 42 27 12 46 17 30
2 9 16 23 30 37 44	24 9 43 35 20 5 39	2 9 30 23 16 37 44	24 9 43 21 34 5 39
3 10 17 24 31 38 45	47 32 17 2 36 28 13	3 10 31 24 17 38 45	47 18 31 2 36 28 13
4 11 18 25 32 39 46	21 6 40 25 10 44 29	4 11 32 25 18 39 46	35 6 40 25 10 44 15
5 12 19 26 33 40 47	37 22 14 48 33 18 3	5 12 33 26 19 40 47	37 22 14 48 19 32 3
6 13 20 27 34 41 48	11 45 30 15 7 41 26	6 13 34 27 20 41 48	11 45 16 29 7 41 26
7 14 21 28 35 42 49	34 19 4 38 23 8 49	7 14 35 28 21 42 49	20 33 4 38 23 8 49
PT: 11/H 1/L	Sol: 1441#	PT: 12/H 1/L	Sol: 1585#
1 15 8 22 36 29 43	1 35 27 19 46 38 9	1 15 36 22 8 29 43	1 35 27 19 46 10 37
2 16 9 23 37 30 44	24 16 43 42 13 5 32	2 16 37 23 9 30 44	24 16 43 14 41 5 32
3 17 10 24 38 31 45	47 39 10 2 29 28 20	3 17 38 24 10 31 45	47 11 38 2 29 28 20
4 18 11 25 39 32 46	14 6 33 25 17 44 36	4 18 39 25 11 32 46	42 6 33 25 17 44 8
5 19 12 26 40 33 47	30 22 21 48 40 11 3	5 19 40 26 12 33 47	30 22 21 48 12 39 3
6 20 13 27 41 34 48	18 45 37 8 7 34 26	6 20 41 27 13 34 48	18 45 9 36 7 34 26
7 21 14 28 42 35 49	41 12 4 31 23 15 49	7 21 42 28 14 35 49	13 40 4 31 23 15 49
PT: 13/H 1/L	Sol: 1729#	PT: 14/H 1/L	Sol: 1873#
1 22 43 36 15 29 8	1 28 48 40 18 31 9	1 22 43 29 8 36 15	1 28 48 33 11 38 16
16 30 9 2 23 44 37	17 30 8 7 27 47 39	9 37 16 2 23 44 30	10 37 15 7 27 47 32
24 45 38 17 31 10 3	26 46 38 16 29 14 6	24 45 31 10 38 17 3	26 46 31 9 36 21 6
32 11 4 25 46 39 18	35 13 5 25 45 37 15	39 18 4 25 46 32 11	42 20 5 25 45 30 8
47 40 19 33 12 5 26	44 36 21 34 12 4 24	47 33 12 40 19 5 26	44 29 14 41 19 4 24
13 6 27 48 41 20 34	11 3 23 43 42 20 33	20 6 27 48 34 13 41	18 3 23 43 35 13 40
42 21 35 14 7 28 49	41 19 32 10 2 22 49	35 14 42 21 7 28 49	34 12 39 17 2 22 49
PT: 15/H 1/L	Sol: 2017#	PT: 16/H 1/L	Sol: 2161#
1 22 43 15 8 36 29	1 28 48 19 11 38 30	1 22 43 8 15 29 36	1 28 48 12 18 31 37
9 37 30 2 23 44 16	10 37 29 7 27 47 18	16 30 37 2 23 44 9	17 30 36 7 27 47 11
24 45 17 10 38 31 3	26 46 17 9 36 35 6	24 45 10 17 31 38 3	26 46 10 16 29 42 6
39 32 4 25 46 18 11	42 34 5 25 45 16 8	32 39 4 25 46 11 18	35 41 5 25 45 9 15
47 19 12 40 33 5 26	44 15 14 41 33 4 24	47 12 19 33 40 5 26	44 8 21 34 40 4 24
34 6 27 48 20 13 41	32 3 23 43 21 13 40	41 6 27 48 13 20 34	39 3 23 43 14 20 33
21 14 42 35 7 28 49	20 12 39 31 2 22 49	14 21 35 42 7 28 49	13 19 32 38 2 22 49
PT: 17/H 1/L	Sol: 2305#	PT: 18/H 1/L	Sol: 2449#
8 1 15 22 29 43 36	8 49 27 5 39 31 16	8 1 29 22 15 43 36	8 49 27 5 39 17 30
9 2 16 23 30 44 37	24 2 36 35 20 12 46	9 2 30 23 16 44 37	24 2 36 21 34 12 46
10 3 17 24 31 45 38	40 32 17 9 43 28 6	10 3 31 24 17 45 38	40 18 31 9 43 28 6
11 4 18 25 32 46 39	21 13 47 25 3 37 29	11 4 32 25 18 46 39	35 13 47 25 3 37 15
12 5 19 26 33 47 40	44 22 7 41 33 18 10	12 5 33 26 19 47 40	44 22 7 41 19 32 10
13 6 20 27 34 48 41	4 38 30 15 14 48 26	13 6 34 27 20 48 41	4 38 16 29 14 48 26
14 7 21 28 35 49 42	34 19 11 45 23 1 42	14 7 35 28 21 49 42	20 33 11 45 23 1 42
PT: 19/H 1/L	Sol: 2593#	PT: 20/H 1/L	Sol: 2737#
8 1 29 15 43 36 22	8 49 6 40 32 24 16	8 1 15 29 43 36 22	8 49 6 40 18 24 30
2 30 16 44 37 23 9	38 30 22 21 13 47 4	2 16 30 44 37 23 9	38 16 22 35 13 47 4
31 17 45 38 24 10 3	19 11 45 2 36 35 27	17 31 45 38 24 10 3	33 11 45 2 36 21 27
18 46 39 25 11 4 32	7 41 33 25 17 9 43	32 46 39 25 11 4 18	7 41 19 25 31 9 43
47 40 26 12 5 33 19	23 15 14 48 5 39 31	47 40 26 12 5 19 33	23 29 14 48 5 39 17
41 27 13 6 34 20 48	46 3 37 29 28 20 12	41 27 13 6 20 34 48	46 3 37 15 28 34 12
28 14 7 35 21 49 42	34 26 18 10 44 1 42	28 14 7 21 35 49 42	20 26 32 10 44 1 42





七方陣の場合は、七進法分解後の上位／下位の各面単位で、分類や変換法を考えても良いのではないかと。例の変換法「半数交換法」も、片面が同じなら、実質的に半数が同じと言えなくもない。

解の新しい分類法、内部変換法の発見に、希望が少なからず膨らむ。

### 6. 汎七方陣の場合

次の問題は、解の総数の多い汎対角線七方陣。やり方はわかっているが、実際の計数プログラムの実行速度が遅くて、結果がなかなか出ない。解の総数が多過ぎると、1個1個の検査・検索・分類・変換もままならない。あれこれ悪戦苦闘してジリジリしながら、2つの解集合の間の一対一対応を長いこと確かめられないままだった。

#### \* \* 原初方陣と目的方陣の第一基本形：基準配列とひな形 \* \*

[原初七方陣]							[汎七方陣]																		
5	6	7	1	2	3	4	5	6	7	1	2	3	28	19	10	1	48	39	30	28	19	10	1	48	39
12	13	14	8	9	10	11	12	13	14	8	9	10	38	29	27	18	9	7	47	38	29	27	18	9	7
19	20	21	15	16	17	18	19	20	21	15	16	17	6	46	37	35	26	17	8	6	46	37	35	26	17
26	27	28	22	23	24	25	26	27	28	22	23	24	16	14	5	45	36	34	25	16	14	5	45	36	34
33	34	35	29	30	31	32	33	34	35	29	30	31	33	24	15	13	4	44	42	33	24	15	13	4	44
40	41	42	36	37	38	39	40	41	42	36	37	38	43	41	32	23	21	12	3	43	41	32	23	21	12
47	48	49	43	44	45	46	47	48	49	43	44	45	11	2	49	40	31	22	20	11	2	49	40	31	22

  

[上位]							[下位]																		
0	0	0	0	0	0	0	0	0	0	0	0	0	4	5	6	0	1	2	3	4	5	6	0	1	2
1	1	1	1	1	1	1	1	1	1	1	1	1	4	5	6	0	1	2	3	4	5	6	0	1	2
2	2	2	2	2	2	2	2	2	2	2	2	2	4	5	6	0	1	2	3	4	5	6	0	1	2
3	3	3	3	3	3	3	3	3	3	3	3	3	4	5	6	0	1	2	3	4	5	6	0	1	2
4	4	4	4	4	4	4	4	4	4	4	4	4	4	5	6	0	1	2	3	4	5	6	0	1	2
5	5	5	5	5	5	5	5	5	5	5	5	5	4	5	6	0	1	2	3	4	5	6	0	1	2
6	6	6	6	6	6	6	6	6	6	6	6	6	4	5	6	0	1	2	3	4	5	6	0	1	2

#### \* 行列の定和 \*

$n1+n48+n39+n30+n28+n19+n10=21$	... rw1		$n1+n18+n35+n45+n13+n23+n40=21$	... cl1
$n18+n9+n7+n47+n38+n29+n27=21$	... rw2		$n48+n9+n26+n36+n4+n21+n31=21$	... cl2
$n35+n26+n17+n8+n6+n46+n37=21$	... rw3		$n39+n7+n17+n34+n44+n12+n22=21$	... cl3
$n45+n36+n34+n25+n16+n14+n5=21$	... rw4		$n30+n47+n8+n25+n42+n3+n20=21$	... cl4
$n13+n4+n44+n42+n33+n24+n15=21$	... rw5		$n28+n38+n6+n16+n33+n43+n11=21$	... cl5
$n23+n21+n12+n3+n43+n41+n32=21$	... rw6		$n19+n29+n46+n14+n24+n41+n2=21$	... cl6
$n40+n31+n22+n20+n11+n2+n49=21$	... rw7		$n10+n27+n37+n5+n15+n32+n49=21$	... cl7

#### \* 汎対角線の定和 \*

$n1+n9+n17+n25+n33+n41+n49=21$	... pd1		$n1+n27+n46+n16+n42+n12+n31=21$	... pq1
$n48+n7+n8+n16+n24+n32+n40=21$	... pd2		$n48+n18+n37+n14+n33+n3+n22=21$	... pq2
$n39+n47+n6+n14+n15+n23+n31=21$	... pd3		$n39+n9+n35+n5+n24+n43+n20=21$	... pq3
$n30+n38+n46+n5+n13+n21+n22=21$	... pd4		$n30+n7+n26+n45+n15+n41+n11=21$	... pq4
$n28+n29+n37+n45+n4+n12+n20=21$	... pd5		$n28+n47+n17+n36+n13+n32+n2=21$	... pq5
$n19+n27+n35+n36+n44+n3+n11=21$	... pd6		$n19+n38+n8+n34+n4+n23+n49=21$	... pq6
$n10+n18+n26+n34+n42+n43+n2=21$	... pd7		$n10+n29+n6+n25+n44+n21+n40=21$	... pq7

原初七方陣を構成する手順は、ほとんど連立型の場合と同じである。

ただし、50の補数ペアの等和と対称配置を初期条件から外す。

そして、各番地につきそれぞれ4本の定和条件を定義する。

「完全オイラー方陣」の構成を目指すので、原初方陣の側でも、七進法分解して上位／下位の各面で「ラテン構造」を追求する。「新オイラー法」でやるしかない。

あれこれ策を講じていろいろと試みたが、高次方陣の計数スピードは、たとえ高速のパソコンでも、気が遠くなるほどに遅い。やはり到底実用にはならない遅さだった。途中で何度も諦めた。だが、思いもかけず、ちょっとした工夫で壁を突破することができた。大方の戦略を変えなくても、ただプログラミングの仕方を戦術的に変えるだけで良かったのだ。

ただし、敵は、何しろ何千万個の解を持つ集合である。それを2個も作って逐一調べ尽くし、比較・検索もしなければならない。ここは、大容量のメモリーを積んだ最新の高性能パソコンをどうしても入手して使う必要があった。幸いハードの進化は速く、願いは叶った。

次に一番新しいプログラムを示そう。スケールは大きいが、何とか全数を数え尽くす。

長いので、プログラムの「キモ」となる部分を示す。その他は、あまり前と変わらない。

```
/** 'Prototype Squares' of Order 7 and Pan-diagonal MS77 **
/** Reconstructed by the 'New Euler's Method' with D5i **
/** File: 'CESPT77PD.c' Dictated by Kanji Setsuda **
/** on July 19, 2005; and Oct. 1, 2006; Recompiled on **
/** Feb.18, 2015 with MacOSX 10.10.2 and Xcode 6.1.1; **
//
#include <stdio.h>
//
long int cnt, bcnt, lcnt;
long cntr[5];
short hcnt, cnt2, cnt3;
short CC, LSM;
short u1,u2;
short nm[50], uflg[50];
short c[50], d[50];
short anm[5][53];
short th[2521][50], tl[20161][50];
char mtc[2521][20161];
//
short rw[8][8], cl[8][8], pd[8][8], pb[8][8];
//
void stp01(void), stp02(void), stp03(void), stp04(void);
void stp05(void), stp06(void), stp07(void), stp08(void);
void stp09(void), stp10(void), stp11(void), stp12(void);
void stp13(void), stp14(void), stp15(void), stp16(void);
void stp17(void), stp18(void), stp19(void), stp20(void);
void stp21(void), stp22(void), stp23(void), stp24(void);
void stp25(void), stp26(void), stp27(void), stp28(void);
void stp29(void), stp30(void), stp31(void), stp32(void);
void stp33(void), stp34(void), stp35(void), stp36(void);
void stp37(void), stp38(void), stp39(void), stp40(void);
void stp41(void), stp42(void), stp43(void), stp44(void);
void stp45(void), stp46(void), stp47(void), stp48(void);
void stp49(void), ansrecord(void);
void prlunit(void);
void cmbcmp(void);
void prans(void), pr2ans(void);
void damt7(void);
void hlsort(short x), excpth(short x);
void llusort(short x), excptl(short x);
//
/** Main Program *
int main(){
short m,n;
printf("\n");
printf("*** Compose 'Prototype Squares' of Order 7 by 'New Euler's Method', ***\n");
```

```

printf("*** and Transform them into Pan-diagonal MS77 by 'DAM Transformation' **\n");
printf("\n");
for(n=0;n<50;n++){nm[n]=-1;}
for(m=0;m<8;m++){
    for(n=0;n<8;n++){
        rw[m][n]=0; cl[m][n]=0; pd[m][n]=0; pb[m][n]=0;}
    }
hcnt=0; cnt=0; bcnt=0; cnt3=0;
stp01();
hlusort(hcnt); llusort(cnt);
prlunit();
printf(" [Compact List of Solutions]\n");
printf("*** 'Prototype Squares' of Order 7 and Pan-diagonal MS77 Reconstructed *\n");
lcnt=cnt; cnt=0; cnt3=0;
for(n=0;n<5;n++){cntr[n]=0;}
cmbcmp();
printf(" [Count = %ld] OK!\n",cnt);
printf("\n");
printf("*** Recompiled and Listed by Kanji Setsuda on\n");
printf(" Feb.18, 2015 with MacOSX 10.10.2 & Xcode 6.1.1 **\n");
printf("\n");
return 0;
}
//
/** Level #1: *
/** Set n1{1,1,1,1} *
void stp01(){
    short r=1, c=1, d=1, b=1;
    short a;
    for(a=0;a<7;a++){
        if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
            nm[1]=a;
            rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
            stp02();
            rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
        }
    }
}
/** Set n49{7,7,1,6} *
void stp02(){
    short r=7, c=7, d=1, b=6;
    short a;
    for(a=6;a>=0;a--){
        if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
            nm[49]=a;
            rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
            stp03();
            rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
        }
    }
}
/** Set n9{2,2,1,3} *
void stp03(){
    short r=2, c=2, d=1, b=3;
    short a;
    for(a=0;a<7;a++){
        if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
            nm[9]=a;
            rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
            stp04();
            rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
        }
    }
}
/** Set n41{6,6,1,4} *
void stp04(){
    short r=6, c=6, d=1, b=4;
    short a;

```



```

for(a=6;a>=0;a--){
    if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
        nm[41]=a;
        rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
        stp05();
        rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
    }
}
}
/* Set n17{3,3,1,5} *
void stp05(){
    short r=3, c=3, d=1, b=5;
    short a;
    for(a=0;a<7;a++){
        if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
            nm[17]=a;
            rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
            stp06();
            rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
        }
    }
}
/* Set n33{5,5,1,2} *
void stp06(){
    short r=5, c=5, d=1, b=2;
    short a;
    for(a=6;a>=0;a--){
        if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
            nm[33]=a;
            rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
            stp07();
            rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
        }
    }
}
}
/* Set n25{4,4,1,7} *
void stp07(){
    short r=4, c=4, d=1, b=7;
    short a;
    for(a=0;a<7;a++){
        if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
            nm[25]=a;
            rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
            stp08();
            rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
        }
    }
}
}
/* Level #2: *
/* Set n48{1,2,2,2} *
void stp08(){
    short r=1, c=2, d=2, b=2;
    short a;
    for(a=6;a>=0;a--){
        if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
            nm[48]=a;
            rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
            stp09();
            rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
        }
    }
}
}
/* Set n2{7,6,7,5} *
void stp09(){
    short r=7, c=6, d=7, b=5;
    short a;
    for(a=0;a<7;a++){
        if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
            nm[2]=a;
            rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;

```

```

        stp10();
        rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
    }}
}
/* Set n39{1,3,3,3} *
void stp10(){
    short r=1, c=3, d=3, b=3;
    short a;
    for(a=6;a>=0;a--){
        if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
            nm[39]=a;
            rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
            stp11();
            rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
        }}
}
/* Set n11{7,5,6,4} *
void stp11(){
    short r=7, c=5, d=6, b=4;
    short a;
    for(a=0;a<7;a++){
        if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
            nm[11]=a;
            rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
            stp12();
            rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
        }}
}
/* Set n30{1,4,4,4} *
void stp12(){
    short r=1, c=4, d=4, b=4;
    short a;
    for(a=6;a>=0;a--){
        if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
            nm[30]=a;
            rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
            stp13();
            rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
        }}
}
/* Set n20{7,4,5,3} *
void stp13(){
    short r=7, c=4, d=5, b=3;
    short a;
    for(a=0;a<7;a++){
        if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
            nm[20]=a;
            rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
            stp14();
            rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
        }}
}
/* Set n28{1,5,5,5} *
void stp14(){
    short r=1, c=5, d=5, b=5;
    short a;
    for(a=6;a>=0;a--){
        if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
            nm[28]=a;
            rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
            stp15();
            rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
        }}
}
/* Set n22{7,3,4,2} *

```

```

void stp15(){
  short r=7, c=3, d=4, b=2;
  short a;
  for(a=0;a<7;a++){
    if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
      nm[22]=a;
      rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
      stp16();
      rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
    }
  }
}
/* Set n19{1,6,6,6} *
void stp16(){
  short r=1, c=6, d=6, b=6;
  short a;
  for(a=6;a>=0;a--){
    if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
      nm[19]=a;
      rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
      stp17();
      rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
    }
  }
}
/* Set n31{7,2,3,1} *
void stp17(){
  short r=7, c=2, d=3, b=1;
  short a;
  for(a=6;a>=0;a--){
    if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
      nm[31]=a;
      rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
      stp18();
      rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
    }
  }
}
/* Set n10{1,7,7,7} *
void stp18(){
  short r=1, c=7, d=7, b=7;
  short a;
  for(a=0;a<7;a++){
    if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
      nm[10]=a;
      rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
      stp19();
      rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
    }
  }
}
/* Set n40{7,1,2,7} *
void stp19(){
  short r=7, c=1, d=2, b=7;
  short a;
  for(a=6;a>=0;a--){
    if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
      nm[40]=a;
      rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
      stp20();
      rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
    }
  }
}
/* Level #3: *
/* Set n45{4,1,5,4} *
void stp20(){
  short r=4, c=1, d=5, b=4;
  short a;
  for(a=6;a>=0;a--){

```

```

    if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
        nm[45]=a;
        rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
        stp21();
        rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
    }
}
/* Set n5{4,7,4,3} *
void stp21(){
    short r=4, c=7, d=4, b=3;
    short a;
    for(a=0;a<7;a++){
        if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
            nm[5]=a;
            rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
            stp22();
            rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
        }
    }
}
/* Set n13{5,1,4,5} *
void stp22(){
    short r=5, c=1, d=4, b=5;
    short a;
    for(a=0;a<7;a++){
        if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
            nm[13]=a;
            rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
            stp23();
            rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
        }
    }
}
/* Set n37{3,7,5,2} *
void stp23(){
    short r=3, c=7, d=5, b=2;
    short a;
    for(a=6;a>=0;a--){
        if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
            nm[37]=a;
            rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
            stp24();
            rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
        }
    }
}
/* Set n35{3,1,6,3} *
void stp24(){
    short r=3, c=1, d=6, b=3;
    short a;
    for(a=6;a>=0;a--){
        if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
            nm[35]=a;
            rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
            stp25();
            rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
        }
    }
}
/* Set n15{5,7,3,4} *
void stp25(){
    short r=5, c=7, d=3, b=4;
    short a;
    for(a=0;a<7;a++){
        if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
            nm[15]=a;
            rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
            stp26();
            rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
        }
    }
}

```

```

    }}
}
/** Set n18{2,1,7,2} *
void stp26(){
    short r=2, c=1, d=7, b=2;
    short a;
    for(a=0;a<7;a++){
        if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
            nm[18]=a;
            rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
            stp27();
            rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
        }
    }
}
/** Set n32{6,7,2,5} *
void stp27(){
    short r=6, c=7, d=2, b=5;
    short a;
    for(a=6;a>=0;a--){
        if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
            nm[32]=a;
            rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
            stp28();
            rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
        }
    }
}
/** Set n23{6,1,3,6} *
void stp28(){
    short r=6, c=1, d=3, b=6;
    short a;
    for(a=0;a<7;a++){
        if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
            nm[23]=a;
            rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
            stp29();
            rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
        }
    }
}
/** Set n27{2,7,6,1} *
void stp29(){
    short r=2, c=7, d=6, b=1;
    short a;
    for(a=6;a>=0;a--){
        if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
            nm[27]=a;
            rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
            stp30();
            rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
        }
    }
}
/** Level #4: *
/** Set n47{2,4,3,5} *
void stp30(){
    short r=2, c=4, d=3, b=5;
    short a;
    for(a=6;a>=0;a--){
        if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
            nm[47]=a;
            rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
            stp31();
            rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
        }
    }
}
/** Set n36{4,2,6,5} *
void stp31(){

```

```

short r=4, c=2, d=6, b=5;
short a;
for(a=6;a>=0;a--){
    if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
        nm[36]=a;
        rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
        stp32();
        rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
    }
}
}
/* Set n3{6,4,6,2} *
void stp32(){
short r=6, c=4, d=6, b=2;
short a;
for(a=0;a<7;a++){
    if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
        nm[3]=a;
        rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
        stp33();
        rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
    }
}
}
/* Set n14{4,6,3,2} *
void stp33(){
short r=4, c=6, d=3, b=2;
short a;
for(a=0;a<7;a++){
    if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
        nm[14]=a;
        rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
        stp34();
        rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
    }
}
}
}
/* Set n6{3,5,3,7} *
void stp34(){
short r=3, c=5, d=3, b=7;
short a;
for(a=0;a<7;a++){
    if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
        nm[6]=a;
        rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
        stp35();
        rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
    }
}
}
}
/* Set n44{5,3,6,7} *
void stp35(){
short r=5, c=3, d=6, b=7;
short a;
for(a=6;a>=0;a--){
    if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
        nm[44]=a;
        rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
        stp36();
        rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
    }
}
}
}
/* Set n7{2,3,2,4} *
void stp36(){
short r=2, c=3, d=2, b=4;
short a;
for(a=0;a<7;a++){
    if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
        nm[7]=a;

```

```

        rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
        stp37();
        rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
    }}
}
/** Set n26{3,2,7,4} *
void stp37(){
    short r=3, c=2, d=7, b=4;
    short a;
    for(a=6;a>=0;a--){
        if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
            nm[26]=a;
            rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
            stp38();
            rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
        }}
}
/** Set n43{6,5,7,3} *
void stp38(){
    short r=6, c=5, d=7, b=3;
    short a;
    for(a=6;a>=0;a--){
        if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
            nm[43]=a;
            rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
            stp39();
            rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
        }}
}
/** Set n24{5,6,2,3} *
void stp39(){
    short r=5, c=6, d=2, b=3;
    short a;
    for(a=0;a<7;a++){
        if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
            nm[24]=a;
            rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
            stp40();
            rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
        }}
}
/** Level #5: *
/** Set n46{3,6,4,1} *
void stp40(){
    short r=3, c=6, d=4, b=1;
    short a;
    for(a=6;a>=0;a--){
        if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
            nm[46]=a;
            rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
            stp41();
            rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
        }}
}
/** Set n8{3,4,2,6} *
void stp41(){
    short r=3, c=4, d=2, b=6;
    short a;
    for(a=0;a<7;a++){
        if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
            nm[8]=a;
            rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
            stp42();
            rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
        }}
}

```

```

}
/* Set n4{5,2,5,6} *
void stp42(){
  short r=5, c=2, d=5, b=6;
  short a;
  for(a=0;a<7;a++){
    if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
      nm[4]=a;
      rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
      stp43();
      rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
    }
  }
}
/* Set n42{5,4,7,1} *
void stp43(){
  short r=5, c=4, d=7, b=1;
  short a;
  for(a=6;a>=0;a--){
    if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
      nm[42]=a;
      rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
      stp44();
      rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
    }
  }
}
/* Set n34{4,3,7,6} *
void stp44(){
  short r=4, c=3, d=7, b=6;
  short a;
  for(a=6;a>=0;a--){
    if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
      nm[34]=a;
      rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
      stp45();
      rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
    }
  }
}
/* Set n16{4,5,2,1} *
void stp45(){
  short r=4, c=5, d=2, b=1;
  short a;
  for(a=0;a<7;a++){
    if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
      nm[16]=a;
      rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
      stp46();
      rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
    }
  }
}
/* Set n38{2,5,4,6} *
void stp46(){
  short r=2, c=5, d=4, b=6;
  short a;
  for(a=6;a>=0;a--){
    if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
      nm[38]=a;
      rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
      stp47();
      rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
    }
  }
}
/* Set n29{2,6,5,7} *
void stp47(){
  short r=2, c=6, d=5, b=7;
  short a;

```



```

for(a=6;a>=0;a--){
    if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
        nm[29]=a;
        rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
        stp48();
        rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
    }
}
/* Set n12{6,3,5,1} *
void stp48(){
    short r=6, c=3, d=5, b=1;
    short a;
    for(a=0;a<7;a++){
        if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
            nm[12]=a;
            rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
            stp49();
            rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
        }
    }
}
/* Set n21{6,2,4,7} *
void stp49(){
    short r=6, c=2, d=4, b=7;
    short a;
    for(a=0;a<7;a++){
        if((rw[r][a]==0)&&(cl[c][a]==0)&&(pd[d][a]==0)&&(pb[b][a]==0)){
            nm[21]=a;
            rw[r][a]=1; cl[c][a]=1; pd[d][a]=1; pb[b][a]=1;
            ansrecord();
            rw[r][a]=0; cl[c][a]=0; pd[d][a]=0; pb[b][a]=0;
        }
    }
}
//
/* Record the Answers *
void ansrecord(){
    short n;
    tl[cnt][0]=cnt+1;
    for(n=1;n<50;n++){tl[cnt][n]=nm[n];}
    if((nm[1]<nm[49])&&(nm[1]<nm[10])&&(nm[1]<nm[40])&&(nm[48]>nm[18])){
        th[hcnt][0]=hcnt+1;
        for(n=1;n<50;n++){th[hcnt][n]=nm[n];}
        hcnt++;
    }
    cnt++;
}
//
/* Classify and Print the Latin Units *
void prlunit(){
    short t,l,m,n;
    printf(" [Counts of Latin Units = %dH/%ldL]\n",hcnt,cnt);
    for(m=0;m<hcnt;m++){
        for(n=0;n<cnt;n++){
            t=0;
            for(l=1;l<50;l++){if(th[m][l]==tl[n][l]){t++;}}
            mtc[m][n]=t;
            t=0;
            for(l=1;l<50;l++){if(th[m][l]+tl[n][l]==6){t++;}}
            if(mtc[m][n]<t){mtc[m][n]=t;}
            if(mtc[m][n]==7){bcnt++;}
        }
    }
    printf("\n");
}
//
/* Combine, Compose and Print *

```

```

void cmbcmp(){
short n,d,fc;
for(u1=0;u1<hcnt;u1++){cnt2=0;
for(u2=0;u2<lcnt;u2++){
if(mtc[u1][u2]==7){
for(n=1;n<50;n++){uflg[n]=0;}
for(n=1;n<50;n++){
d=th[u1][n]*7+tl[u2][n]+1;
nm[n]=d; uflg[d]++;
}
fc=0;
for(n=1;n<50;n++){if(uflg[n]==1){fc++;}else{break;}}
if(fc==49){
if((nm[1]<nm[49])&&(nm[1]<nm[10])&&(nm[1]<nm[40])){
if(nm[48]>nm[18]){prans();}
}}}
}
}
}
//
/* Print the Answers *
void prans(){
short n;
cnt++; cnt2++;
if(cnt2==1){
cntr[cnt3]=cnt;
for(n=1;n<50;n++){anm[cnt3][n]=nm[n];}
anm[cnt3][51]=u1+1; anm[cnt3][52]=u2+1;
cnt3++;
if(cnt3==2){pr2ans(); cnt3=0;}
}
}
//
/* Print 2 Answers *
void pr2ans(){
short l,l7,m,n;
long cm;
for(m=0;m<2;m++){cm=cntr[m];
for(n=1;n<50;n++){c[n]=anm[m][n];}
damt7();
for(n=1;n<50;n++){anm[m+2][n]=d[n];}
printf(" PT:%7d/H%6d/L Sol:%15ld#",anm[m][51],anm[m][52],cm);
if(m<1){printf(" ");}
}
printf("\n");
for(l=0;l<7;l++){l7=l*7;
for(m=0;m<2;m++){
for(n=1;n<8;n++){printf("%3d",anm[m][l7+n]);}
printf(" ");
for(n=1;n<8;n++){printf("%3d",anm[m+2][l7+n]);}
if(m<1){printf(" ");}
}
printf("\n");
}
printf("\n");
}
//
/* Do-it-After-the-Model Transformation *
void damt7(){
d[1]=c[1]; d[2]=c[48]; d[3]=c[39]; d[4]=c[30]; d[5]=c[28]; d[6]=c[19]; d[7]=c[10];
d[8]=c[18]; d[9]=c[9]; d[10]=c[7]; d[11]=c[47]; d[12]=c[38]; d[13]=c[29]; d[14]=c[27];
d[15]=c[35]; d[16]=c[26]; d[17]=c[17]; d[18]=c[8]; d[19]=c[6]; d[20]=c[46]; d[21]=c[37];
d[22]=c[45]; d[23]=c[36]; d[24]=c[34]; d[25]=c[25]; d[26]=c[16]; d[27]=c[14]; d[28]=c[5];
d[29]=c[13]; d[30]=c[4]; d[31]=c[44]; d[32]=c[42]; d[33]=c[33]; d[34]=c[24]; d[35]=c[15];

```

```

d[36]=c[23]; d[37]=c[21]; d[38]=c[12]; d[39]=c[3]; d[40]=c[43]; d[41]=c[41]; d[42]=c[32];
d[43]=c[40]; d[44]=c[31]; d[45]=c[22]; d[46]=c[20]; d[47]=c[11]; d[48]=c[2]; d[49]=c[49];
}
//
/* Sort the Latin Units *
void hlsort(short x){
short mx,n,f;
int d1,d2,d3,d4;
mx=x-1;
do{f=0;
for(n=0;n<mx;n++){
d1=(((th[n][1]*7+th[n][2])*7+th[n][3])*7+th[n][4])*7+th[n][5])*7+th[n][6];
d2=(((th[n+1][1]*7+th[n+1][2])*7+th[n+1][3])*7+th[n+1][4])*7+th[n+1][5])*7+th[n+1][6];
d3=(((th[n][8]*7+th[n][15])*7+th[n][22])*7+th[n][29])*7+th[n][36];
d4=(((th[n+1][8]*7+th[n+1][15])*7+th[n+1][22])*7+th[n+1][29])*7+th[n+1][36];
if(d1>d2){excpth(n); f=1;}
else if((d1==d2)&&(d3>d4)){excpth(n); f=1;}
}
mx--;
}while(f>0);
}
//
void excpth(short x){
short n,d;
for(n=0;n<50;n++){d=th[x][n]; th[x][n]=th[x+1][n]; th[x+1][n]=d;}
}
//
/* Sort the Latin Units *
void llusort(short x){
short mx,n,f;
int d1,d2,d3,d4;
mx=x-1;
do{f=0;
for(n=0;n<mx;n++){
d1=(((tl[n][1]*7+tl[n][2])*7+tl[n][3])*7+tl[n][4])*7+tl[n][5])*7+tl[n][6];
d2=(((tl[n+1][1]*7+tl[n+1][2])*7+tl[n+1][3])*7+tl[n+1][4])*7+tl[n+1][5])*7+tl[n+1][6];
d3=(((tl[n][8]*7+tl[n][15])*7+tl[n][22])*7+tl[n][29])*7+tl[n][36];
d4=(((tl[n+1][8]*7+tl[n+1][15])*7+tl[n+1][22])*7+tl[n+1][29])*7+tl[n+1][36];
if(d1>d2){excptl(n); f=1;}
else if((d1==d2)&&(d3>d4)){excptl(n); f=1;}
}
mx--;
}while(f>0);
}
//
void excptl(short x){
short n,d;
for(n=0;n<50;n++){d=tl[x][n]; tl[x][n]=tl[x+1][n]; tl[x+1][n]=d;}
}
//

```

プログラムで定義して行く変数の順序が、鍵だった。n1, n2, n3, n4, n5, n6, n7, n8, n9, n10, n11, ... ,n47, n48, n49 と順序正しく決定するのでなく、あくまで「ひな形」に即して効率良く決めて行けば良かったのである。こうすることで、極めて高速に作業を進めることができた。やっと実用的なプログラムが出来上がった。

ただし、その副作用として、解の出力が必ずしも綺麗ではなくなった。後で新たに順序正しくソートする必要が生じた。だが、ラテン方阵の単位面だけのソートを行えば良い。その手間ひまは多寡が知れている。待てないほどの遅延にはならない（3千8百万個もの最終解のソートはやらない方が良いけれど）。

次のリストが、最近の実験結果の概要である。何もかがやたらと多いので、さわりだけを抜粋して示す。これは何せ「高次」の宿命なので、御勘弁いただきたい。

\* 原初七方陣の構成とひな形変換法による汎七方陣の再構成 \*

\* 「新オイラー法」によって構成した七進法分解図の単位面 [上位] \*

1/	2/	3/	4/	5/	6/	7/	8/
0000000	0000000	0000000	0000000	0000000	0000000	0000000	0000000
1111111	1111111	1111111	1111111	1111111	1111111	1111111	1111111
2222222	2222222	2222222	2222222	2222222	2222222	2222222	2222222
3333333	3333333	3333333	3333333	3333333	3333333	4444444	4444444
4444444	4444444	5555555	5555555	6666666	6666666	3333333	3333333
5555555	6666666	4444444	6666666	4444444	5555555	5555555	6666666
6666666	5555555	6666666	4444444	5555555	4444444	6666666	5555555
505/	506/	507/	508/	509/	510/	511/	512/
0231456	0231465	0231546	0231564	0231546	0231645	0231654	0231645
0231456	0231465	0231546	0231564	2315460	0231645	0231654	2316450
0231456	0231465	0231546	0231564	3154602	0231645	0231654	3164502
0231456	0231465	0231546	0231564	1546023	0231645	0231654	1645023
0231456	0231465	0231546	0231564	5460231	0231645	0231654	6450231
0231456	0231465	0231546	0231564	4602315	0231645	0231654	4502316
0231456	0231465	0231546	0231564	6023154	0231645	0231654	5023164
1009/	1010/	1011/	1012/	1013/	1014/	1015/	1016/
0514263	0514236	0514263	0514362	0514326	0514362	0514326	0514623
0514263	2360514	2630514	0514362	3260514	3620514	5143260	5146230
0514263	5142360	5142630	0514362	5143260	5143620	1432605	1462305
0514263	3605142	6305142	0514362	2605143	6205143	4326051	4623051
0514263	1423605	1426305	0514362	1432605	1436205	3260514	6230514
0514263	6051423	3051426	0514362	6051432	2051436	2605143	2305146
0514263	4236051	4263051	0514362	4326051	4362051	6051432	3051462
1513/	1514/	1515/	1516/	1517/	1518/	1519/	1520/
1052346	1052364	1052436	1052436	1052463	1052634	1052643	1052634
1052346	1052364	0524361	1052436	1052463	0526341	0526431	1052634
1052346	1052364	5243610	1052436	1052463	5263410	5264310	1052634
1052346	1052364	2436105	1052436	1052463	2634105	2643105	1052634
1052346	1052364	4361052	1052436	1052463	6341052	6431052	1052634
1052346	1052364	3610524	1052436	1052463	3410526	4310526	1052634
1052346	1052364	6105243	1052436	1052463	4105263	3105264	1052634
2017/	2018/	2019/	2020/	2021/	2022/	2023/	2024/
1526430	1530246	1530264	1530426	1530462	1530624	1530642	1532046
5264301	1530246	1530264	1530426	1530462	1530624	1530642	0461532
2643015	1530246	1530264	1530426	1530462	1530624	1530642	5320461
6430152	1530246	1530264	1530426	1530462	1530624	1530642	4615320
4301526	1530246	1530264	1530426	1530462	1530624	1530642	3204615
3015264	1530246	1530264	1530426	1530462	1530624	1530642	6153204
0152643	1530246	1530264	1530426	1530462	1530624	1530642	2046153

\* 「新オイラー法」によって構成した七進法分解図の単位面 [下位] \*

1/	2/	3/	4/	5/	6/	7/	8/
0000000	0000000	0000000	0000000	0000000	0000000	0000000	0000000
1111111	1111111	1111111	1111111	1111111	1111111	1111111	1111111
2222222	2222222	2222222	2222222	2222222	2222222	2222222	2222222
3333333	3333333	3333333	3333333	3333333	3333333	4444444	4444444
4444444	4444444	5555555	5555555	6666666	6666666	3333333	3333333
5555555	6666666	4444444	6666666	4444444	5555555	5555555	6666666
6666666	5555555	6666666	4444444	5555555	4444444	6666666	5555555
4033/	4034/	4035/	4036/	4037/	4038/	4039/	4040/
1230456	1230465	1230456	1230465	1230456	1230465	1230546	1230564
1230456	1230465	2304561	2304651	4561230	4651230	1230546	1230564
1230456	1230465	3045612	3046512	2304561	2304651	1230546	1230564
1230456	1230465	0456123	0465123	5612304	6512304	1230546	1230564
1230456	1230465	4561230	4651230	3045612	3046512	1230546	1230564
1230456	1230465	5612304	6512304	6123045	5123046	1230546	1230564
1230456	1230465	6123045	5123046	0456123	0465123	1230546	1230564
8065/	8066/	8067/	8068/	8069/	8070/	8071/	8072/
2530146	2530164	2530146	2530164	2530146	2530164	2530416	2530461
1462530	1642530	2530146	2530164	5301462	5301642	2530416	2530461
5301462	5301642	2530146	2530164	3014625	3016425	2530416	2530461
4625301	6425301	2530146	2530164	0146253	0164253	2530416	2530461
3014625	3016425	2530146	2530164	1462530	1642530	2530416	2530461
6253014	4253016	2530146	2530164	4625301	6425301	2530416	2530461
0146253	0164253	2530146	2530164	6253014	4253016	2530416	2530461

12097/ 4150236 1502364 5023641 0236415 2364150 3641502 6415023	12098/ 4150263 1502634 5026341 0263415 2634150 6341502 3415026	12099/ 4150236 2364150 1502364 3641502 5023641 6415023 0236415	12100/ 4150263 2634150 1502634 6341502 5026341 3415026 0263415	12101/ 4150236 1502364 5023641 0236415 2364150 6341502 3415026	12102/ 4150263 1502634 5026341 0263415 2634150 6341502 3415026	12103/ 4150326 1503264 5032641 0326415 3264150 2641503 6415032	12104/ 4150362 1503624 5036241 0362415 3624150 6241503 2415036
16129/ 5460123 1235460 4601235 2354601 6012354 3546012 0123546	16130/ 5460132 1325460 4601325 3254601 6013254 2546013 0132546	16131/ 5460123 4601235 6012354 0123546 1235460 2354601 3546012	16132/ 5460132 4601325 6013254 0132546 1325460 2546013 0254601	16133/ 5460123 5460123 5460123 5460123 5460123 5460123 5460123	16134/ 5460132 5460132 5460132 5460132 5460132 5460132 5460132	16135/ 5460213 2135460 4602135 1354602 6021354 3546021 0213546	16136/ 5460231 2315460 4602315 3154602 6023154 1546023 0231546

[Counts of Layer Units = 2520H/20160L]

\* 最良の組合せを得るための参照表 \*

T1	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24
1	49	35	35	28	28	35	35	21	28	21	21	28	28	21	35	28	21	21	21	28	28	35	21	21
2	35	49	28	35	35	28	21	35	21	28	28	21	21	28	28	35	21	21	28	21	35	28	21	21
3	35	28	49	35	35	28	28	21	35	28	21	21	35	21	28	21	21	28	28	21	21	21	28	35
4	28	35	35	49	28	35	21	28	28	35	21	21	21	35	21	28	28	21	21	28	21	21	35	28
5	28	35	35	28	49	35	21	28	21	21	35	28	28	21	21	21	28	35	35	21	28	21	21	28
6	35	28	28	35	35	49	28	21	21	21	28	35	21	28	21	21	35	28	21	35	21	28	28	21
7	35	21	28	21	21	28	49	35	35	28	28	35	35	28	28	21	21	21	28	35	21	28	21	21
8	21	35	21	28	28	21	35	49	28	35	35	28	28	35	21	28	21	21	35	28	28	21	21	21
9	28	21	35	28	21	21	35	28	49	35	35	28	28	21	35	21	28	21	21	21	28	21	35	28
10	21	28	28	35	21	21	28	35	35	49	28	35	21	28	21	35	21	28	21	21	21	28	28	35
11	21	28	21	21	35	28	28	35	35	28	49	35	21	21	28	21	35	28	28	21	35	21	28	21
12	28	21	21	21	28	35	35	28	28	35	35	49	21	21	21	28	28	35	21	28	21	35	21	28
13	28	21	35	21	28	21	35	28	28	21	21	21	49	35	35	28	28	35	35	28	21	21	21	28
14	21	28	21	35	21	28	28	35	21	28	21	21	35	49	28	35	35	28	28	35	21	21	28	21
15	35	28	28	21	21	21	28	21	35	21	28	21	35	28	49	35	35	28	21	21	35	28	28	21
16	28	35	21	28	21	21	21	28	21	35	21	28	28	35	35	49	28	35	21	21	28	35	21	28
17	21	21	21	28	28	35	21	21	28	21	35	28	28	35	35	28	49	35	21	28	28	21	35	21
18	21	21	28	21	35	28	21	21	21	28	28	35	35	28	28	35	35	49	28	21	21	28	21	35
19	21	28	28	21	35	21	28	35	21	21	28	21	35	28	21	21	28	49	35	35	28	28	35	28
20	28	21	21	28	21	35	35	28	21	21	28	28	35	21	21	28	21	35	49	28	35	35	28	35
21	28	35	21	21	28	21	21	28	28	21	35	21	21	35	28	28	21	35	28	49	35	35	28	28
22	35	28	21	21	21	28	28	21	21	28	21	35	21	21	28	35	21	28	28	35	35	49	28	35
23	21	21	28	35	21	28	21	21	35	28	28	21	21	28	28	21	35	21	28	35	35	28	49	35
24	21	21	35	28	28	21	21	21	28	35	21	28	28	21	21	28	21	35	35	28	28	35	35	49
300+	61	62	63	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79	80	81	82	83	84
721	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
722	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
723	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
724	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
725	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
726	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
727	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
728	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
729	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
730	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
731	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
732	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
733	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
734	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
735	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
736	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
737	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7
738	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7	7

739| 7  
 740| 7  
 741| 7  
 742| 7  
 743| 7  
 744| 7

\* 原初七方陣の構成とひな形変換法による汎七方陣の再構成 \*

PT:	1/H	721/L	Sol:	1#	PT:	2/H	721/L	Sol:	15121#																		
1	2	3	4	5	6	7	1	48	39	30	28	19	10	1	2	3	4	5	6	7	1	41	46	30	28	19	10
8	9	10	11	12	13	14	18	9	7	47	38	29	27	8	9	10	11	12	13	14	18	9	7	40	45	29	27
15	16	17	18	19	20	21	35	26	17	8	6	46	37	15	16	17	18	19	20	21	35	26	17	8	6	39	44
22	23	24	25	26	27	28	45	36	34	25	16	14	5	22	23	24	25	26	27	28	38	43	34	25	16	14	5
29	30	31	32	33	34	35	13	4	44	42	33	24	15	29	30	31	32	33	34	35	13	4	37	49	33	24	15
36	37	38	39	40	41	42	23	21	12	3	43	41	32	43	44	45	46	47	48	49	23	21	12	3	36	48	32
43	44	45	46	47	48	49	40	31	22	20	11	2	49	36	37	38	39	40	41	42	47	31	22	20	11	2	42

PT:	3/H	721/L	Sol:	30241#	PT:	4/H	721/L	Sol:	45361#																		
1	2	3	4	5	6	7	1	48	32	37	28	19	10	1	2	3	4	5	6	7	1	34	46	37	28	19	10
8	9	10	11	12	13	14	18	9	7	47	31	36	27	8	9	10	11	12	13	14	18	9	7	33	45	36	27
15	16	17	18	19	20	21	42	26	17	8	6	46	30	15	16	17	18	19	20	21	42	26	17	8	6	32	44
22	23	24	25	26	27	28	45	29	41	25	16	14	5	22	23	24	25	26	27	28	31	43	41	25	16	14	5
36	37	38	39	40	41	42	13	4	44	35	40	24	15	36	37	38	39	40	41	42	13	4	30	49	40	24	15
29	30	31	32	33	34	35	23	21	12	3	43	34	39	43	44	45	46	47	48	49	23	21	12	3	29	48	39
43	44	45	46	47	48	49	33	38	22	20	11	2	49	29	30	31	32	33	34	35	47	38	22	20	11	2	35

PT:	5/H	721/L	Sol:	60481#	PT:	6/H	721/L	Sol:	75601#																		
1	2	3	4	5	6	7	1	41	32	44	28	19	10	1	2	3	4	5	6	7	1	34	39	44	28	19	10
8	9	10	11	12	13	14	18	9	7	40	31	43	27	8	9	10	11	12	13	14	18	9	7	33	38	43	27
15	16	17	18	19	20	21	49	26	17	8	6	39	30	15	16	17	18	19	20	21	49	26	17	8	6	32	37
22	23	24	25	26	27	28	38	29	48	25	16	14	5	22	23	24	25	26	27	28	31	36	48	25	16	14	5
43	44	45	46	47	48	49	13	4	37	35	47	24	15	43	44	45	46	47	48	49	13	4	30	42	47	24	15
29	30	31	32	33	34	35	23	21	12	3	36	34	46	36	37	38	39	40	41	42	23	21	12	3	29	41	46
36	37	38	39	40	41	42	33	45	22	20	11	2	42	29	30	31	32	33	34	35	40	45	22	20	11	2	35

PT:	7/H	721/L	Sol:	90721#	PT:	8/H	721/L	Sol:	105841#																		
1	2	3	4	5	6	7	1	48	39	23	35	19	10	1	2	3	4	5	6	7	1	41	46	23	35	19	10
8	9	10	11	12	13	14	18	9	7	47	38	22	34	8	9	10	11	12	13	14	18	9	7	40	45	22	34
15	16	17	18	19	20	21	28	33	17	8	6	46	37	15	16	17	18	19	20	21	28	33	17	8	6	39	44
29	30	31	32	33	34	35	45	36	27	32	16	14	5	29	30	31	32	33	34	35	38	43	27	32	16	14	5
22	23	24	25	26	27	28	13	4	44	42	26	31	15	22	23	24	25	26	27	28	13	4	37	49	26	31	15
36	37	38	39	40	41	42	30	21	12	3	43	41	25	43	44	45	46	47	48	49	30	21	12	3	36	48	25
43	44	45	46	47	48	49	40	24	29	20	11	2	49	36	37	38	39	40	41	42	47	24	29	20	11	2	42

PT:	9/H	721/L	Sol:	120961#	PT:	10/H	721/L	Sol:	136081#																		
1	2	3	4	5	6	7	1	48	25	37	35	19	10	1	2	3	4	5	6	7	1	27	46	37	35	19	10
8	9	10	11	12	13	14	18	9	7	47	24	36	34	8	9	10	11	12	13	14	18	9	7	26	45	36	34
15	16	17	18	19	20	21	42	33	17	8	6	46	23	15	16	17	18	19	20	21	42	33	17	8	6	25	44
29	30	31	32	33	34	35	45	22	41	32	16	14	5	29	30	31	32	33	34	35	24	43	41	32	16	14	5
36	37	38	39	40	41	42	13	4	44	28	40	31	15	36	37	38	39	40	41	42	13	4	23	49	40	31	15
22	23	24	25	26	27	28	30	21	12	3	43	27	39	43	44	45	46	47	48	49	30	21	12	3	22	48	39
43	44	45	46	47	48	49	26	38	29	20	11	2	49	22	23	24	25	26	27	28	47	38	29	20	11	2	28

.....

PT:	501/H	1/L	Sol:	7560001#	PT:	502/H	1/L	Sol:	7575121#																			
1	15	8	36	43	29	22	1	49	20	33	11	24	37	1	15	8	43	29	22	36	1	35	20	26	11	38	44	
16	9	37	44	30	23	2	31	9	22	42	6	47	18	16	9	44	30	23	37	2	24	9	36	49	6	33	18	
10	38	45	31	24	3	17	40	4	45	16	29	14	27	10	45	31	24	38	3	17	47	4	31	16	22	14	41	
39	46	32	25	4	18	11	21	34	12	25	38	2	43	39	46	32	25	39	4	18	11	21	27	12	39	45	2	29
47	33	26	5	19	12	40	23	36	7	48	19	32	10	47	33	26	5	19	12	47	37	43	7	34	19	25	10	
34	27	6	20	13	41	48	46	17	30	8	28	41	5	34	27	6	20	13	48	34	32	17	23	8	42	48	5	
28	7	21	14	42	49	35	13	26	39	3	44	15	35	28	7	21	14	49	35	28	13	40	46	3	30	15	28	

PT: 503/H 1/L	Sol: 7590241#	PT: 504/H 1/L	Sol: 7605361#
1 15 8 43 36 22 29	1 42 20 26 11 31 44	1 15 8 43 36 29 22	1 42 20 33 11 24 44
16 9 44 37 23 30 2	24 9 29 49 6 40 18	16 9 44 37 30 23 2	31 9 22 49 6 40 18
10 45 38 24 31 3 17	47 4 38 16 22 14 34	10 45 38 31 24 3 17	47 4 38 16 29 14 27
46 39 25 32 4 18 11	21 27 12 32 45 2 36	46 39 32 25 4 18 11	21 34 12 25 45 2 36
40 26 33 5 19 12 47	30 43 7 41 19 25 10	40 33 26 5 19 12 47	23 43 7 41 19 32 10
27 34 6 20 13 48 41	39 17 23 8 35 48 5	34 27 6 20 13 48 41	39 17 30 8 28 48 5
35 7 21 14 49 42 28	13 33 46 3 37 15 28	28 7 21 14 49 42 35	13 26 46 3 37 15 35

PT: 505/H 1/L	Sol: 7620481#	PT: 506/H 1/L	Sol: 7635601#
1 15 22 8 29 36 43	1 42 13 19 46 31 23	1 15 22 8 29 43 36	1 49 13 19 39 31 23
2 16 23 9 30 37 44	10 16 43 35 27 5 39	2 16 23 9 30 44 37	10 16 36 35 27 5 46
3 17 24 10 31 38 45	47 32 24 2 36 14 20	3 17 24 10 31 45 38	40 32 24 2 43 14 20
4 18 25 11 32 39 46	28 6 40 11 17 44 29	4 18 25 11 32 46 39	28 6 47 11 17 37 29
5 19 26 12 33 40 47	37 8 21 48 33 25 3	5 19 26 12 33 47 40	44 8 21 41 33 25 3
6 20 27 13 34 41 48	18 45 30 22 7 41 12	6 20 27 13 34 48 41	18 38 30 22 7 48 12
7 21 28 14 35 42 49	34 26 4 38 9 15 49	7 21 28 14 35 49 42	34 26 4 45 9 15 42

PT: 507/H 1/L	Sol: 7650721#	PT: 508/H 1/L	Sol: 7665841#
1 15 22 8 36 29 43	1 35 13 19 46 38 23	1 15 22 8 36 29 43	1 42 20 33 25 45 9
2 16 23 9 37 30 44	10 16 43 42 27 5 32	16 23 9 37 30 44 2	31 23 43 14 6 40 18
3 17 24 10 38 31 45	47 39 24 2 29 14 20	24 10 38 31 45 3 17	12 4 38 16 29 28 48
4 18 25 11 39 32 46	28 6 33 11 17 44 36	11 39 32 46 4 18 25	21 34 26 46 10 2 36
5 19 26 12 40 33 47	30 8 21 48 40 25 3	40 33 47 5 19 26 12	44 8 7 41 19 32 24
6 20 27 13 41 34 48	18 45 37 22 7 34 12	34 48 6 20 27 13 41	39 17 30 22 49 13 5
7 21 28 14 42 35 49	41 26 4 31 9 15 49	49 7 21 28 14 42 35	27 47 11 3 37 15 35

PT: 509/H 1/L	Sol: 7680961#	PT: 510/H 1/L	Sol: 7696081#
1 15 22 8 36 43 29	1 49 13 19 32 38 23	1 15 22 8 43 29 36	1 35 13 19 39 45 23
2 16 23 9 37 44 30	10 16 29 42 27 5 46	2 16 23 9 44 30 37	10 16 36 49 27 5 32
3 17 24 10 38 45 31	33 39 24 2 43 14 20	3 17 24 10 45 31 38	40 46 24 2 29 14 20
4 18 25 11 39 46 32	28 6 47 11 17 30 36	4 18 25 11 46 32 39	28 6 33 11 17 37 43
5 19 26 12 40 47 33	44 8 21 34 40 25 3	5 19 26 12 47 33 40	30 8 21 41 47 25 3
6 20 27 13 41 48 34	18 31 37 22 7 48 12	6 20 27 13 48 34 41	18 38 44 22 7 34 12
7 21 28 14 42 49 35	41 26 4 45 9 15 35	7 21 28 14 49 35 42	48 26 4 31 9 15 42

. . . . .

PT: 1001/H 1/L	Sol: 15120001#	PT: 1002/H 1/L	Sol: 15135121#
1 36 8 22 15 43 29	1 42 13 26 18 45 30	1 36 8 22 29 15 43	1 42 13 26 32 17 44
16 44 30 2 37 9 23	17 44 29 7 41 12 25	30 16 44 2 37 9 23	31 16 43 7 41 12 25
38 10 24 17 45 31 3	40 11 24 16 43 35 6	38 10 24 31 17 45 3	40 11 24 30 15 49 6
46 32 4 39 11 25 18	49 34 5 39 10 23 15	18 46 4 39 11 25 32	21 48 5 39 10 23 29
12 26 19 47 33 5 40	9 22 21 48 33 4 38	12 26 33 19 47 5 40	9 22 35 20 47 4 38
34 6 41 13 27 20 48	32 3 37 8 28 20 47	48 6 41 13 27 34 20	46 3 37 8 28 34 19
28 21 49 35 7 42 14	27 19 46 31 2 36 14	28 35 21 49 7 42 14	27 33 18 45 2 36 14

PT: 1003/H 1/L	Sol: 15150241#	PT: 1004/H 1/L	Sol: 15165361#
1 36 8 22 29 15 43	1 35 41 19 11 45 23	1 36 8 22 29 43 15	1 49 27 40 18 31 9
37 9 23 30 16 44 2	17 9 43 28 6 33 39	2 37 9 23 30 44 16	24 37 15 35 13 5 46
10 24 31 17 45 3 38	26 4 31 37 15 14 48	3 38 10 24 31 45 17	19 32 10 2 43 28 41
25 32 18 46 4 39 11	42 20 12 46 24 2 29	4 39 11 25 32 46 18	14 6 47 25 38 16 29
33 19 47 5 40 12 26	44 22 7 34 40 18 10	5 40 12 26 33 47 19	44 22 42 20 33 11 3
20 48 6 41 13 27 34	32 38 16 8 49 27 5	6 41 13 27 34 48 20	39 17 30 8 7 48 26
49 7 42 14 28 35 21	13 47 25 3 30 36 21	7 42 14 28 35 49 21	34 12 4 45 23 36 21

PT: 1005/H 1/L	Sol: 15180481#	PT: 1006/H 1/L	Sol: 15195601#
1 36 8 22 29 43 15	1 42 13 26 32 45 16	1 36 8 22 43 15 29	1 49 41 19 11 31 23
30 44 16 2 37 9 23	31 44 15 7 41 12 25	37 9 23 44 16 30 2	17 9 29 28 6 47 39
38 10 24 31 45 17 3	40 11 24 30 43 21 6	10 24 45 17 31 3 38	26 4 45 37 15 14 34
46 18 4 39 11 25 32	49 20 5 39 10 23 29	25 46 18 32 4 39 11	42 20 12 32 24 2 43
12 26 33 47 19 5 40	9 22 35 48 19 4 38	47 19 33 5 40 12 26	30 22 7 48 40 18 10
20 6 41 13 27 34 48	18 3 37 8 28 34 47	20 34 6 41 13 27 48	46 38 16 8 35 27 5
28 35 49 21 7 42 14	27 33 46 17 2 36 14	35 7 42 14 28 49 21	13 33 25 3 44 36 21

PT: 1007/H 1/L	Sol: 15210721#	PT: 1008/H 1/L	Sol: 15225841#
1 36 8 22 43 29 15	1 35 27 40 18 45 9	1 36 8 22 43 29 15	1 49 41 33 11 17 23
2 37 9 23 44 30 16	24 37 15 49 13 5 32	37 9 23 44 30 16 2	31 9 15 28 6 47 39
3 38 10 24 45 31 17	19 46 10 2 29 28 41	10 24 45 31 17 3 38	26 4 45 37 29 14 20
4 39 11 25 46 32 18	14 6 33 25 38 16 43	25 46 32 18 4 39 11	42 34 12 18 24 2 43
5 40 12 26 47 33 19	30 22 42 20 47 11 3	47 33 19 5 40 12 26	16 22 7 48 40 32 10
6 41 13 27 48 34 20	39 17 44 8 7 34 26	34 20 6 41 13 27 48	46 38 30 8 21 27 5
7 42 14 28 49 35 21	48 12 4 31 23 36 21	21 7 42 14 28 49 35	13 19 25 3 44 36 35

PT: 1009/H 1/L	Sol: 15240961#	PT: 1010/H 1/L	Sol: 15256081#
1 36 8 29 15 22 43	1 42 13 33 18 24 44	1 36 8 29 15 43 22	1 49 34 40 25 17 9
16 23 44 2 37 9 30	17 23 43 7 41 12 32	2 37 9 30 16 44 23	31 37 22 21 13 5 46
38 10 31 17 24 45 3	40 11 31 16 22 49 6	3 38 10 31 17 45 24	26 18 10 2 43 35 41
25 46 4 39 11 32 18	28 48 5 39 10 30 15	4 39 11 32 18 46 25	14 6 47 32 38 23 15
12 33 19 26 47 5 40	9 29 21 27 47 4 38	5 40 12 33 19 47 26	44 29 42 27 19 11 3
48 6 41 13 34 20 27	46 3 37 8 35 20 26	6 41 13 34 20 48 27	39 24 16 8 7 48 33
35 21 28 49 7 42 14	34 19 25 45 2 36 14	7 42 14 35 21 49 28	20 12 4 45 30 36 28

. . . . .

PT: 2001/H 1/L	Sol: 30240001#	PT: 2002/H 1/L	Sol: 30255121#
8 36 15 1 29 22 43	8 28 6 40 46 31 16	8 36 15 1 29 43 22	8 49 6 40 25 31 16
9 37 16 2 30 23 44	3 37 43 35 20 12 25	9 37 16 2 30 44 23	3 37 22 35 20 12 46
10 38 17 3 31 24 45	47 32 17 9 22 7 41	10 38 17 3 31 45 24	26 32 17 9 43 7 41
11 39 18 4 32 25 46	21 13 26 4 38 44 29	11 39 18 4 32 46 25	21 13 47 4 38 23 29
12 40 19 5 33 26 47	23 1 42 48 33 18 10	12 40 19 5 33 47 26	44 1 42 27 33 18 10
13 41 20 6 34 27 48	39 45 30 15 14 27 5	13 41 20 6 34 48 27	39 24 30 15 14 48 5
14 42 21 7 35 28 49	34 19 11 24 2 36 49	14 42 21 7 35 49 28	34 19 11 45 2 36 28

PT: 2003/H 1/L	Sol: 30270241#	PT: 2004/H 1/L	Sol: 30285361#
8 36 15 1 43 22 29	8 28 6 40 32 45 16	8 36 15 1 43 29 22	8 35 6 40 25 45 16
9 37 16 2 44 23 30	3 37 29 49 20 12 25	9 37 16 2 44 30 23	3 37 22 49 20 12 32
10 38 17 3 45 24 31	33 46 17 9 22 7 41	10 38 17 3 45 31 24	26 46 17 9 29 7 41
11 39 18 4 46 25 32	21 13 26 4 38 30 43	11 39 18 4 46 32 25	21 13 33 4 38 23 43
12 40 19 5 47 26 33	23 1 42 34 47 18 10	12 40 19 5 47 33 26	30 1 42 27 47 18 10
13 41 20 6 48 27 34	39 31 44 15 14 27 5	13 41 20 6 48 34 27	39 24 44 15 14 34 5
14 42 21 7 49 28 35	48 19 11 24 2 36 35	14 42 21 7 49 35 28	48 19 11 31 2 36 28

PT: 2005/H 1/L	Sol: 30300481#	PT: 2006/H 1/L	Sol: 30315601#
8 36 15 22 1 29 43	8 42 20 26 4 31 44	8 36 15 22 1 43 29	8 42 20 26 4 45 30
2 30 44 9 37 16 23	3 30 43 14 41 19 25	2 44 30 9 37 16 23	3 44 29 14 41 19 25
38 17 24 3 31 45 10	40 18 24 2 29 49 13	38 17 24 3 45 31 10	40 18 24 2 43 35 13
32 46 11 39 18 25 4	35 48 12 39 17 23 1	46 32 11 39 18 25 4	49 34 12 39 17 23 1
19 26 5 33 47 12 40	16 22 7 34 47 11 38	19 26 5 47 33 12 40	16 22 7 48 33 11 38
48 13 41 20 27 6 34	46 10 37 15 28 6 33	34 13 41 20 27 6 48	32 10 37 15 28 6 47
28 7 35 49 14 42 21	27 5 32 45 9 36 21	28 7 49 35 14 42 21	27 5 46 31 9 36 21

PT: 2007/H 1/L	Sol: 30330721#	PT: 2008/H 1/L	Sol: 30345841#
8 36 15 22 29 1 43	8 42 20 26 32 3 44	8 36 15 22 43 29 1	8 49 41 33 18 3 23
30 2 44 9 37 16 23	31 2 43 14 41 19 25	37 16 23 44 30 2 9	31 16 1 28 13 47 39
38 17 24 31 3 45 10	40 18 24 30 1 49 13	17 24 45 31 3 10 38	26 11 45 37 29 21 6
4 46 11 39 18 25 32	7 48 12 39 17 23 29	25 46 32 4 11 39 18	42 34 19 4 24 9 43
19 26 33 5 47 12 40	16 22 35 6 47 11 38	47 33 5 12 40 19 26	2 22 14 48 40 32 17
48 13 41 20 27 34 6	46 10 37 15 28 34 5	34 6 13 41 20 27 48	46 38 30 15 7 27 12
28 35 7 49 14 42 21	27 33 4 45 9 36 21	7 14 42 21 28 49 35	20 5 25 10 44 36 35

PT: 2009/H 1/L	Sol: 30360961#	PT: 2010/H 1/L	Sol: 30376081#
8 36 15 29 1 22 43	8 42 20 33 4 24 44	8 36 15 29 1 43 22	8 42 20 33 4 45 23
2 23 44 9 37 16 30	3 23 43 14 41 19 32	2 44 23 9 37 16 30	3 44 22 14 41 19 32
38 17 31 3 24 45 10	40 18 31 2 22 49 13	38 17 31 3 45 24 10	40 18 31 2 43 28 13
25 46 11 39 18 32 4	28 48 12 39 17 30 1	46 25 11 39 18 32 4	49 27 12 39 17 30 1
19 33 5 26 47 12 40	16 29 7 27 47 11 38	19 33 5 47 26 12 40	16 29 7 48 26 11 38
48 13 41 20 34 6 27	46 10 37 15 35 6 26	27 13 41 20 34 6 48	25 10 37 15 35 6 47
35 7 28 49 14 42 21	34 5 25 45 9 36 21	35 7 49 28 14 42 21	34 5 46 24 9 36 21

. . . . .



PT: 2511/H 721/L	Sol: 37951201#	PT: 2512/H 721/L	Sol: 37966321#
22 23 24 25 26 27 28	22 41 32 9 21 5 45	22 23 24 25 26 27 28	22 34 39 9 21 5 45
43 44 45 46 47 48 49	4 44 28 40 31 8 20	43 44 45 46 47 48 49	4 44 28 33 38 8 20
1 2 3 4 5 6 7	14 19 3 43 27 39 30	1 2 3 4 5 6 7	14 19 3 43 27 32 37
15 16 17 18 19 20 21	38 29 13 18 2 49 26	15 16 17 18 19 20 21	31 36 13 18 2 49 26
8 9 10 11 12 13 14	48 25 37 35 12 17 1	8 9 10 11 12 13 14	48 25 30 42 12 17 1
29 30 31 32 33 34 35	16 7 47 24 36 34 11	36 37 38 39 40 41 42	16 7 47 24 29 41 11
36 37 38 39 40 41 42	33 10 15 6 46 23 42	29 30 31 32 33 34 35	40 10 15 6 46 23 35
PT: 2513/H 721/L	Sol: 37981441#	PT: 2514/H 721/L	Sol: 37996561#
22 23 24 25 26 27 28	22 41 32 16 7 12 45	22 23 24 25 26 27 28	22 34 39 16 7 12 45
43 44 45 46 47 48 49	11 44 28 40 31 15 6	43 44 45 46 47 48 49	11 44 28 33 38 15 6
8 9 10 11 12 13 14	21 5 10 43 27 39 30	8 9 10 11 12 13 14	21 5 10 43 27 32 37
1 2 3 4 5 6 7	38 29 20 4 9 49 26	1 2 3 4 5 6 7	31 36 20 4 9 49 26
15 16 17 18 19 20 21	48 25 37 35 19 3 8	15 16 17 18 19 20 21	48 25 30 42 19 3 8
29 30 31 32 33 34 35	2 14 47 24 36 34 18	36 37 38 39 40 41 42	2 14 47 24 29 41 18
36 37 38 39 40 41 42	33 17 1 13 46 23 42	29 30 31 32 33 34 35	40 17 1 13 46 23 35
PT: 2515/H 721/L	Sol: 38011681#	PT: 2516/H 721/L	Sol: 38026801#
22 23 24 25 26 27 28	22 41 32 2 21 12 45	22 23 24 25 26 27 28	22 34 39 2 21 12 45
43 44 45 46 47 48 49	11 44 28 40 31 1 20	43 44 45 46 47 48 49	11 44 28 33 38 1 20
8 9 10 11 12 13 14	7 19 10 43 27 39 30	8 9 10 11 12 13 14	7 19 10 43 27 32 37
15 16 17 18 19 20 21	38 29 6 18 9 49 26	15 16 17 18 19 20 21	31 36 6 18 9 49 26
1 2 3 4 5 6 7	48 25 37 35 5 17 8	1 2 3 4 5 6 7	48 25 30 42 5 17 8
29 30 31 32 33 34 35	16 14 47 24 36 34 4	36 37 38 39 40 41 42	16 14 47 24 29 41 4
36 37 38 39 40 41 42	33 3 15 13 46 23 42	29 30 31 32 33 34 35	40 3 15 13 46 23 35
PT: 2517/H 721/L	Sol: 38041921#	PT: 2518/H 721/L	Sol: 38057041#
22 23 24 25 26 27 28	22 41 32 9 7 19 45	22 23 24 25 26 27 28	22 34 39 9 7 19 45
43 44 45 46 47 48 49	18 44 28 40 31 8 6	43 44 45 46 47 48 49	18 44 28 33 38 8 6
15 16 17 18 19 20 21	14 5 17 43 27 39 30	15 16 17 18 19 20 21	14 5 17 43 27 32 37
1 2 3 4 5 6 7	38 29 13 4 16 49 26	1 2 3 4 5 6 7	31 36 13 4 16 49 26
8 9 10 11 12 13 14	48 25 37 35 12 3 15	8 9 10 11 12 13 14	48 25 30 42 12 3 15
29 30 31 32 33 34 35	2 21 47 24 36 34 11	36 37 38 39 40 41 42	2 21 47 24 29 41 11
36 37 38 39 40 41 42	33 10 1 20 46 23 42	29 30 31 32 33 34 35	40 10 1 20 46 23 35
PT: 2519/H 721/L	Sol: 38072161#	PT: 2520/H 721/L	Sol: 38087281#
22 23 24 25 26 27 28	22 41 32 2 14 19 45	22 23 24 25 26 27 28	22 34 39 2 14 19 45
43 44 45 46 47 48 49	18 44 28 40 31 1 13	43 44 45 46 47 48 49	18 44 28 33 38 1 13
15 16 17 18 19 20 21	7 12 17 43 27 39 30	15 16 17 18 19 20 21	7 12 17 43 27 32 37
8 9 10 11 12 13 14	38 29 6 11 16 49 26	8 9 10 11 12 13 14	31 36 6 11 16 49 26
1 2 3 4 5 6 7	48 25 37 35 5 10 15	1 2 3 4 5 6 7	48 25 30 42 5 10 15
29 30 31 32 33 34 35	9 21 47 24 36 34 4	36 37 38 39 40 41 42	9 21 47 24 29 41 4
36 37 38 39 40 41 42	33 3 8 20 46 23 42	29 30 31 32 33 34 35	40 3 8 20 46 23 35

[Count = 38102400] OK!

\*\* Recompiled and Listed by Kanji Setsuda on  
Feb. 18, 2015 with MacOSX 10.10.2 & Xcode 6.1.1 \*\*

画面上にプリント・アウトする時間も含めて、ほんの1～2分間しか掛からなかった。  
めでたく総数 38102400 個の原初七方陣を作ることができた。そして、それを各個ひな形  
変換して目的の汎対角線七方陣を 38102400 個作ることができた。これで、原初七方陣の解  
と汎七方陣の解との間に一対一対応の美しい関係が存在することが確かめられた。

不思議と言えば、実に不思議である。どの魔方陣にも、それに対応する原初方陣の解集合  
が必ず1つは存在することが、どんどん確からしく思えて来る。

では、その原初方陣とは、いったい何ものなのか。……

小生は今、「基本図・展開図（もしくは変換図）の集合」ではないかと、思っている。

(第1稿: Apr. 16, 2003; 改訂最新稿: Oct. 5, 2006; 計数・作表 with MacOSX & Xcode 2.2;  
再計算書稿: on Feb. 18, 2015 with MacOSX 10.10.2 & Xcode 6.1.1)

著者: 摂田 寛二(Kanji Setsuda) <jag12001@nifty.ne.jp>

**\*\*<付録>\*\***

七次でも、五次と同様に、どのタイプの魔方陣でも「ひな形」を共通にしてある。つまり、汎七方陣も対称汎七方陣も、同じ1つのひな形から構成している。

そうすることで、一つにはいろいろなことが比較分析しやすくなるからであるが、他方ではそのひな形を「大基本解」扱いできるようにもしたいからだ。

では、重要なそのひな形をどう記憶しておくか。次のような提案をしたい。ハンドメイドで「構成する方法」を記憶しておくのである。小生は、いつもそうしている。

**\* 原初七方陣から目的方陣の「ひな形」を手作りで構成する方法 \***

[Start] /原初七方陣/

48	49	43	44	45	46	47	48	49	43	44
6	7	1	2	3	4	5	6	7	1	2
13	14	8	9	10	11	12	13	14	8	9
20	21	15	16	17	18	19	20	21	15	16
27	28	22	23	24	25	26	27	28	22	23
34	35	29	30	31	32	33	34	35	29	30
41	42	36	37	38	39	40	41	42	36	37
48	49	43	44	45	46	47	48	49	43	44
6	7	1	2	3	4	5	6	7	1	2

[Goal] /目的方陣のひな形/

2	49	40	31	22	20	11	2	49	40	31
19	10	1	48	39	30	28	19	10	1	48
29	27	18	9	7	47	38	29	27	18	9
46	37	35	26	17	8	6	46	37	35	26
14	5	45	36	34	25	16	14	5	45	36
24	15	13	4	44	42	33	24	15	13	4
41	32	23	21	12	3	43	41	32	23	21
2	49	40	31	22	20	11	2	49	40	31
19	10	1	48	39	30	28	19	10	1	48

[Step 1] Copy Primary Diagonal

.	49	.	.	.	.	.	.	49	.	.
.	1	.	.	.	.	.	.	1	.	.
.	.	9	.	.	.	.	.	.	9	.
.	.	.	17	.	.	.	.	.	.	.
.	.	.	.	25	.	.	.	.	.	.
.	.	.	.	.	33	.	.	.	.	.
41	.	.	.	.	.	41	.	.	.	.
.	49	.	.	.	.	.	.	49	.	.
.	.	1	.	.	.	.	.	1	.	.

[Step 2] Place 1~7

2	49	.	.	.	.	.	.	2	49	.
.	1	.	.	.	.	.	.	1	.	.
.	.	9	7	.	.	.	.	.	9	.
.	.	.	17	.	6	.	.	.	.	.
.	5	.	.	25	.	.	5	.	.	.
.	.	4	.	.	33	.	.	4	.	.
41	.	.	.	.	3	41	.	.	.	.
2	49	.	.	.	.	.	2	49	.	.
.	.	1	.	.	.	.	.	1	.	.

[Step 3] Place 8~14

2	49	.	.	.	11	2	49	.	.	.
.	10	1	.	.	.	.	10	1	.	.
.	.	.	9	7	.	.	.	.	9	.
.	.	.	.	17	8	6	.	.	.	.
14	5	.	.	.	25	14	5	.	.	.
.	.	13	4	.	.	33	.	13	4	.
41	.	.	.	12	3	.	41	.	.	.
2	49	.	.	.	.	11	2	49	.	.
.	10	1	.	.	.	.	10	1	.	.

[Step 4] Place 15~21

2	49	.	.	.	20	11	2	49	.	.
19	10	1	.	.	.	.	19	10	1	.
.	.	18	9	7	.	.	.	.	18	9
.	.	.	.	17	8	6	.	.	.	.
14	5	.	.	.	25	16	14	5	.	.
.	15	13	4	.	.	33	.	15	13	4
41	.	.	21	12	3	.	41	.	.	21
2	49	.	.	.	.	20	11	2	49	.
19	10	1	.	.	.	.	19	10	1	.

[Step 5] Place 22~28

2	49	.	.	22	20	11	2	49	.	.						
19	10		1		.		.		28		19		10		1	.
.	27		18		9		7		.		.		27		18	9
.	.		.		26		17		8		6		.		.	26
14	5		.		.		25		16		14		5		.	.
24	15		13		4		.		33		24		15		13	4
41	.		23		21		12		3		.		41		23	21
2	49		.		.		22		20		11		2		49	.
19	10	1	.	.	.	28	19	10	1	.	.	.	28	19	10	1

[Step 6] Place 29~35

2	49	.	.	31	22	20	11	2	49	.	31								
19	10		1		.		.		30		28		19		10		1	.	
29	27		18		9		7		.		.		29		27		18	9	
.	.		.		35		26		17		8		6		.		.	35	26
14	5		.		.		34		25		16		14		5		.	.	
24	15		13		4		.		33		24		15		13	4			
41	32		23		21		12		3		.		41		32		23	21	
2	49		.		.		31		22		20		11		2		49	.	31
19	10	1	.	.	.	30	28	19	10	1	.	.	.	30	28	19	10	1	.

[Step 7] Place 36~42

2	49	40	31	22	20	11	2	49	40	31									
19	10		1		.		39		30		28		19		10		1	.	
29	27		18		9		7		.		38		29		27		18	9	
.	37		35		26		17		8		6		.		37		35	26	
14	5		.		36		34		25		16		14		5		.	36	
24	15		13		4		.		42		33		24		15		13	4	
41	32		23		21		12		3		.		41		32		23	21	
2	49		40		31		22		20		11		2		49		40	31	
19	10	1	.	.	39	30	28	19	10	1	.	.	.	39	30	28	19	10	1

[Step 8] Place 43~49

2	49	40	31	22	20	11	2	49	40	31														
19	10		1		48		39		30		28		19		10		1	48						
29	27		18		9		7		47		38		29		27		18	9						
46	37		35		26		17		8		6		46		37		35	26						
14	5		45		36		34		25		16		14		5		45	36						
24	15		13		4		44		42		33		24		15		13	4						
41	32		23		21		12		3		43		41		32		23	21						
2	49		40		31		22		20		11		2		49		40	31						
19	10	1	48	39	30	28	19	10	1	48	39	30	28	19	10	1	48	39	30	28	19	10	1	48

上図のようなヒントのみを掲げる。想像力を働かして、作成法を読み取ってほしい。  
 (主対角線の各数から始めて、「1上がって2左へ」進みながら、数値を1つつ増やして行くのが「コツ」だ。途中で数値の変な「桁下り」がある。あっ、言ってしまった。ごめん！)