

## 第 6 章 : 多次元超立体魔方陣の研究 : 摂田 寛二

### 第 2-3 節 : 四次元超立体二方陣の構成と分解 : Part 2

0. この節では、四次元超立体二方陣の**新構成法**を試す。六次元二方陣の場合で成功した新構成法（第 6 章第 4 節で述べている）を、四次元でも追求したい。

我々の方から注文を付けて目的方陣の基本図に定義条件式を与え、そうして四次元二方陣を解いて、作りたいものを何とか取り出そうというのである。

1. 例えば、あの**標準型四方陣**の解 880 個を全数取り出すことはできないだろうか。

これは今までは想定を超えた目的方陣だったが、四次元超立体二方陣には本当にできないのだろうか？ 試しに作ってみることにしよう。

1-1. 次の基本図を用意する。四次元では、原点  $n1$  の隣の数は、 $n2, n5, n9$  だけでなく  $n3$  も隣にある。それを確認するために 2 通りの基本図を示した。これに次のような定義条件式を与えて、それを解いてみよう。四次元二方陣を解くのと同一ことになるはずだ。

図 0 : 基本図

1#/E1	/E2	T01/	2#/E1	/E2	T02/
1--- 2	3--- 4	1 2 3 4	1--- 3	2--- 4	1 3 2 4
9---10	11---12	5 6 7 8	5--- 7	6--- 8	9 11 10 12
5- - 6	7- - 8	9 10 11 12	9- -11	10- -12	5 7 6 8
13---14	15---16	13 14 15 16	13---15	14---16	13 15 14 16

[四次元超立体二方陣図]

[標準型四方陣用]

[変形標準型用]

\* 定義条件式 :  $C=34$  \*

```

n1+n2+n3+n4=C      ...rw1;  n1+n5+n9+n13=C      ...cl1;  n1+n6+n11+n16=C      ...pd1;
n5+n6+n7+n8=C      ...rw2;  n2+n6+n10+n14=C      ...cl2;
n9+n10+n11+n12=C   ...rw3;  n3+n7+n11+n15=C      ...cl3;
n13+n14+n15+n16=C ...rw4;  n4+n8+n12+n16=C      ...cl4;  n4+n7+n10+n13=C      ...pd8;

```

\* 正規化用の不等式 (「標準解」を出力するため) \*

$n1 < n4 < n13; n1 < 16;$

1-2. 早速にも組んだプログラムを以下に示す (英語版で、申し訳ない)。

```

/** Make the 4-D Extra-Cubic Magic Objects of Order 2 **
/** and Transform them into 2 Types of 'Standard' MS44 **
/** File: 'EC024STMS44.c' Dictated by Kanji Setsuda **
/** in 2005, 2012 and on Jan.19, 2014 **
/** with Apple OSX 10.9.1 & Xcode 5.0.2 **
/**
#include <stdio.h>
/** Variables *
short int cnt, cnt1;
short LSM, CC;
short nm[17], flg[17];
short rnm[881][17];
short t[17], s[17];
short tn[4][37];
/** Sub-Routines *
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 savesol(void);
void prlist(void);
void anschksm(void);
void checksums(short x);

```

```

short fndon(void);
void pr2trns(void);
void pr2t(void);
void trnsf01(void);
void trnsf02(void);
//
/** Main Program *
int main(){
short n;
printf("\n");
printf("*** Make the 4-D Extra-Cubic Magic Objects of Order 2\n");
printf("  and Transform them into 2 Types of 'Standard' MS44 **\n");
printf(" [Solutions of 4-D EC02^4: 2 Types of Standard MS44 Transformed]\n");
for(n=0;n<17;n++){nm[n]=0; flg[n]=0;}
LSM=34; CC=17; cnt=0; cnt1=0;
stp01();      /* Reconstruction of Object Solutions *
prlist();     /* Print the Solution List *
printf(" [Solution Counts(n1=1/All) = %d/%d] OK!\n",cnt1,cnt);
printf("\n");
printf("*** Calculated and Listed by Kanji Setsuda on\n");
printf("  Jan.19, 2014 with Apple OSX 10.9.1 & Xcode 5.0.2 **\n");
printf("\n");
return 0;
}
//
/** Reconstructions of Object Solutions *
/** Set n1 *
void stp01(){
short a;
for(a=1;a<8;a++){
if(flg[a]==0){
nm[1]=a; flg[a]=1;
stp02();
flg[a]=0;}}
}
/** Set n2 *
void stp02(){
short a;
for(a=1;a<17;a++){
if(flg[a]==0){
nm[2]=a; flg[a]=1;
stp03();
flg[a]=0;}}
}
/** Set n3 *
void stp03(){
short a;
for(a=1;a<17;a++){
if(flg[a]==0){
nm[3]=a; flg[a]=1;
stp04();
flg[a]=0;}}
}
/** Set n4=LSM-n1-n2-n3 & n1<n4 *
void stp04(){
short a;
a=LSM-nm[1]-nm[2]-nm[3];
if((nm[1]<a)&&(a<17)){
if(flg[a]==0){
nm[4]=a; flg[a]=1;
stp05();
flg[a]=0;}}
}
/** Set n5 *
void stp05(){
short a;

```

```

    for(a=1;a<17;a++){
        if(flg[a]==0){
            nm[5]=a; flg[a]=1;
            stp06();
            flg[a]=0;}}
}
/** Set n6 *
void stp06(){
    short a;
    for(a=1;a<17;a++){
        if(flg[a]==0){
            nm[6]=a; flg[a]=1;
            stp07();
            flg[a]=0;}}
}
/** Set n7 *
void stp07(){
    short a;
    for(a=1;a<17;a++){
        if(flg[a]==0){
            nm[7]=a; flg[a]=1;
            stp08();
            flg[a]=0;}}
}
/** Set n8=LSM-n5-n6-n7 *
void stp08(){
    short a;
    a=LSM-nm[5]-nm[6]-nm[7];
    if((0<a)&&(a<17)){
        if(flg[a]==0){
            nm[8]=a; flg[a]=1;
            stp09();
            flg[a]=0;}}
}
/** Level #2 *
/** Set n9 *
void stp09(){
    short a;
    for(a=1;a<17;a++){
        if(flg[a]==0){
            nm[9]=a; flg[a]=1;
            stp10();
            flg[a]=0;}}
}
/** Set n13=LSM-n1-n5-n9 & n4<n13 *
void stp10(){
    short a;
    a=LSM-nm[1]-nm[5]-nm[9];
    if((nm[4]<a)&&(a<17)){
        if(flg[a]==0){
            nm[13]=a; flg[a]=1;
            stp11();
            flg[a]=0;}}
}
/** Set n10=LSM-n4-n7-n13 *
void stp11(){
    short a;
    a=LSM-nm[4]-nm[7]-nm[13];
    if((0<a)&&(a<17)){
        if(flg[a]==0){
            nm[10]=a; flg[a]=1;
            stp12();
            flg[a]=0;}}
}
/** Set n14=LSM-n2-n6-n10 *
void stp12(){

```

```

short a;
a=LSM-nm[2]-nm[6]-nm[10];
if((0<a)&&(a<17)){
    if(flg[a]==0){
        nm[14]=a; flg[a]=1;
        stp13();
        flg[a]=0;}}
}
/* Set n11 *
void stp13(){
    short a;
    for(a=1;a<17;a++){
        if(flg[a]==0){
            nm[11]=a; flg[a]=1;
            stp14();
            flg[a]=0;}}
}
/* Set n12=LSM-n9-n10-n11 *
void stp14(){
    short a;
    a=LSM-nm[9]-nm[10]-nm[11];
    if((0<a)&&(a<17)){
        if(flg[a]==0){
            nm[12]=a; flg[a]=1;
            stp15();
            flg[a]=0;}}
}
/* Set n15=LSM-n3-n7-n11 *
void stp15(){
    short a;
    a=LSM-nm[3]-nm[7]-nm[11];
    if((0<a)&&(a<17)){
        if(flg[a]==0){
            nm[15]=a; flg[a]=1;
            stp16();
            flg[a]=0;}}
}
/* Set n16=LSM-n13-n15-n14 & n1<n16 *
void stp16(){
    short a,b,c;
    a=LSM-nm[13]-nm[14]-nm[15];
    b=LSM-nm[4]-nm[8]-nm[12];
    c=LSM-nm[1]-nm[6]-nm[11];
    if((a==b)&&(nm[1]<a)&&(a<17)){
        if((a==c)&&(flg[a]==0)){
            nm[16]=a; flg[a]=1;
            savesol();
            flg[a]=0;}}
}
//
/* Save each Solution to the Reference Table *
void savesol(){
    short n;
    rnm[cnt][0]=cnt+1;
    for(n=1;n<17;n++){rnm[cnt][n]=nm[n];}
    cnt++; if(nm[1]==1){cnt1++;}
}
//
/* Print the Solution List *
void prlist(){
    short m,n,c,d;
    d=0;
    for(m=0;m<cnt;m++){
        if(m<100){c=(rnm[m][1]*16+rnm[m][2])*16+rnm[m][5];}
        else if((100<=m)&&(m<208)){c=rnm[m][1]*16+rnm[m][2];}

```

```

else if(m>=208){c=rnm[m][1];}
if(c!=d){d=c;
  tn[0][0]=rnm[m][0];
  for(n=1;n<17;n++){s[n]=rnm[m][n];}; trnsf01();
  for(n=1;n<17;n++){tn[0][n]=t[n];}
  checksums(0);
  for(n=1;n<17;n++){s[n]=t[n];}; trnsf02();
  tn[1][0]=fndon();
  for(n=1;n<17;n++){tn[1][n]=t[n];}
  checksums(1);
  anschksm();
}
}
}
//
/* Print Single Answers with precise Check-sums *
void anschksm(){
  printf("\n");
  printf("      /E1      /E2  %10d/",tn[0][0]);
  if(tn[0][33]==16){printf("SC");}
  else if(tn[0][35]==8){printf("PD");}
  else if(tn[0][36]>0){printf("ST");}
  else{printf("  ");}
  printf("lRwlCl\\Pd/pdl %10d/",tn[1][0]);
  if(tn[1][33]==16){printf("SC");}
  else if(tn[1][35]==8){printf("PD");}
  else if(tn[1][36]>0){printf("ST");}
  else{printf("  ");}
  printf("lRwlCl\\Pd/pdl\n");
  printf("%4d---%2d%6d---%2d",tn[0][1],tn[0][2],tn[0][3],tn[0][4]);
  printf("%7d%3d%3d%3d%2d%2d\\%2d/%2d",
    tn[0][1],tn[0][2],tn[0][3],tn[0][4],tn[0][17],tn[0][21],tn[0][25],tn[0][29]);
  printf("%5d%3d%3d%3d%2d%2d\\%2d/%2d\n",
    tn[1][1],tn[1][2],tn[1][3],tn[1][4],tn[1][17],tn[1][21],tn[1][25],tn[1][29]);
  printf("  l%2d---%2d  l%2d---%2d",tn[0][9],tn[0][10],tn[0][11],tn[0][12]);
  printf("%5d%3d%3d%3d%2d%2d\\%2d/%2d",
    tn[0][5],tn[0][6],tn[0][7],tn[0][8],tn[0][18],tn[0][22],tn[0][26],tn[0][30]);
  printf("%5d%3d%3d%3d%2d%2d\\%2d/%2d\n",
    tn[1][5],tn[1][6],tn[1][7],tn[1][8],tn[1][18],tn[1][22],tn[1][26],tn[1][30]);
  printf("%4d-l-%2d l%4d-l-%2d l",tn[0][5],tn[0][6],tn[0][7],tn[0][8]);
  printf("%5d%3d%3d%3d%2d%2d\\%2d/%2d",
    tn[0][9],tn[0][10],tn[0][11],tn[0][12],tn[0][19],tn[0][23],tn[0][27],tn[0][31]);
  printf("%5d%3d%3d%3d%2d%2d\\%2d/%2d\n",
    tn[1][9],tn[1][10],tn[1][11],tn[1][12],tn[1][19],tn[1][23],tn[1][27],tn[1][31]);
  printf("%6d---%2d%6d---%2d",tn[0][13],tn[0][14],tn[0][15],tn[0][16]);
  printf("%5d%3d%3d%3d%2d%2d\\%2d/%2d",
    tn[0][13],tn[0][14],tn[0][15],tn[0][16],tn[0][20],tn[0][24],tn[0][28],tn[0][32]);
  printf("%5d%3d%3d%3d%2d%2d\\%2d/%2d\n",
    tn[1][13],tn[1][14],tn[1][15],tn[1][16],tn[1][20],tn[1][24],tn[1][28],tn[1][32]);
}
//
/* Check Sums of every Conditions *
void checksums(short x){
  short m,n;
  short sc[17]={0,11,12,10, 9,15,16,14,13, 7, 8, 6, 5, 3, 4, 2, 1};
  /* for Basic Conditions *
  tn[x][17]=t[1]+t[2]+t[3]+t[4];      tn[x][18]=t[5]+t[6]+t[7]+t[8];
  tn[x][19]=t[9]+t[10]+t[11]+t[12];  tn[x][20]=t[13]+t[14]+t[15]+t[16];
  tn[x][21]=t[1]+t[5]+t[9]+t[13];    tn[x][22]=t[2]+t[6]+t[10]+t[14];
  tn[x][23]=t[3]+t[7]+t[11]+t[15];   tn[x][24]=t[4]+t[8]+t[12]+t[16];
  /* for Pandiagonal Conditions *
  tn[x][25]=t[1]+t[6]+t[11]+t[16];   tn[x][26]=t[2]+t[7]+t[12]+t[13];
  tn[x][27]=t[3]+t[8]+t[9]+t[14];    tn[x][28]=t[4]+t[5]+t[10]+t[15];
  tn[x][29]=t[1]+t[8]+t[11]+t[14];   tn[x][30]=t[2]+t[5]+t[12]+t[15];
  tn[x][31]=t[3]+t[6]+t[9]+t[16];    tn[x][32]=t[4]+t[7]+t[10]+t[13];
  /* for Conditions of Types **

```

```

tn[x][33]=0;
for(n=1;n<17;n++){m=17-n; if(t[n]+t[m]==17){tn[x][33]++;}}
tn[x][34]=0;
for(n=1;n<17;n++){m=sc[n]; if(t[m]+t[17-m]==17){tn[x][34]++;}}
tn[x][35]=0;
for(n=25;n<33;n++){if(tn[x][n]==34){tn[x][35]++;}}
tn[x][36]=0;
if((tn[x][25]==34)&&(tn[x][32]==34)){tn[x][36]++;}
}
//
/* Find Old Number from the Reference List *
short fndon(){
short m,n,f;
for(m=0;m<cnt;m++){
f=0;
for(n=1;n<17;n++){
if(t[n]==rnm[m][n]){f++;}
else{break;}
}
if(f==16){f=rnm[m][0]; break;}
}
return f;
}
//
/* The 2 Types of Transformations *
void trnsf01(){
t[ 1]=s[ 1]; t[ 2]=s[ 2]; t[ 3]=s[ 3]; t[ 4]=s[ 4];
t[ 5]=s[ 5]; t[ 6]=s[ 6]; t[ 7]=s[ 7]; t[ 8]=s[ 8];
t[ 9]=s[ 9]; t[10]=s[10]; t[11]=s[11]; t[12]=s[12];
t[13]=s[13]; t[14]=s[14]; t[15]=s[15]; t[16]=s[16];};
void trnsf02(){
t[ 1]=s[ 1]; t[ 2]=s[ 3]; t[ 3]=s[ 2]; t[ 4]=s[ 4];
t[ 5]=s[ 9]; t[ 6]=s[11]; t[ 7]=s[10]; t[ 8]=s[12];
t[ 9]=s[ 5]; t[10]=s[ 7]; t[11]=s[ 6]; t[12]=s[ 8];
t[13]=s[13]; t[14]=s[15]; t[15]=s[14]; t[16]=s[16];};
//

```

1-3. 計算して得られた結果は次のようなものだった。抜粋のリストを以下に示そう。  
得られた左側の解の第2・3番目の行列を交換して、果たして別の同じ「標準型」が作れるのかも調べてみた。右側に置いた解がその結果である。  
各解にチェックサム・モニターを付けた。これで正解かどうか分かる。

**\*\* 四次元超立体二方陣を解いて標準型四方陣を得た結果：詳しいチェックサム・モニター付き \*\***

[/ST:標準型; /SC:補数対称型; /PD:汎対角線型; Rw:行和; Cl:列和; \Pd/pd:汎対角線の和:]

/E1	/E2	1/ST Rw Cl\Pd/pd	185/ST Rw Cl\Pd/pd
1--- 5	16---12	1 5 16 12 34 34\34/34	1 16 5 12 34 34\34/24
10--- 4	13--- 7	8 14 3 9 34 34\30/22	10 13 4 7 34 34\44/46
8- -14	3- - 9	10 4 13 7 34 34\46/46	8 3 14 9 34 34\22/32
15---11	2--- 6	15 11 2 6 34 34\26/34	15 2 11 6 34 34\36/34

/E1	/E2	2/ST Rw Cl\Pd/pd	184/ST Rw Cl\Pd/pd
1--- 5	16---12	1 5 16 12 34 34\34/32	1 16 5 12 34 34\34/26
8--- 4	13--- 9	10 14 3 7 34 34\32/26	8 13 4 9 34 34\42/42
10- -14	3- - 7	8 4 13 9 34 34\42/44	10 3 14 7 34 34\26/34
15---11	2--- 6	15 11 2 6 34 34\28/34	15 2 11 6 34 34\34/34

/E1	/E2	3/ST Rw Cl\Pd/pd	149/PD Rw Cl\Pd/pd
1--- 6	15---12	1 6 15 12 34 34\34/22	1 15 6 12 34 34\34/34
8--- 3	10---13	11 16 5 2 34 34\38/34	8 10 3 13 34 34\34/34
11- -16	5- - 2	8 3 10 13 34 34\34/46	11 5 16 2 34 34\34/34
14--- 9	4--- 7	14 9 4 7 34 34\30/34	14 4 9 7 34 34\34/34

/E1	/E2	4/ST	Rw	CI\Pd/pd	148/ST	Rw	CI\Pd/pd
1---6	15---12	1	6	15	12	34	34\34/30
4---7	14---9	16	11	2	5	34	34\30/34
16- -11	2- -5	4	7	14	9	34	34\34/38
13---10	3---8	13	10	3	8	34	34\38/34

/E1	/E2	5/ST	Rw	CI\Pd/pd	187/ST	Rw	CI\Pd/pd
1---6	16---11	1	6	16	11	34	34\34/22
7---4	10---13	12	15	5	2	34	34\38/34
12- -15	5- -2	7	4	10	13	34	34\34/46
14---9	3---8	14	9	3	8	34	34\30/34

/E1	/E2	6/ST	Rw	CI\Pd/pd	186/ST	Rw	CI\Pd/pd
1---6	16---11	1	6	16	11	34	34\34/28
4---7	13---10	15	12	2	5	34	34\32/34
15- -12	2- -5	4	7	13	10	34	34\34/40
14---9	3---8	14	9	3	8	34	34\36/34

/E1	/E2	7/ST	Rw	CI\Pd/pd	63/PD	Rw	CI\Pd/pd
1---7	12---14	1	7	12	14	34	34\34/28
8---2	13---11	10	16	3	5	34	34\36/34
10- -16	3- -5	8	2	13	11	34	34\34/40
15---9	6---4	15	9	6	4	34	34\32/34

/E1	/E2	8/ST	Rw	CI\Pd/pd	121/ST	Rw	CI\Pd/pd
1---7	14---12	1	7	14	12	34	34\34/33
11---2	13---8	6	15	4	9	34	34\35/24
6- -15	4- -9	11	2	13	8	34	34\44/45
16---10	3---5	16	10	3	5	34	34\23/34

/E1	/E2	9/ST	Rw	CI\Pd/pd	120/ST	Rw	CI\Pd/pd
1---7	14---12	1	7	14	12	34	34\34/37
9---4	15---6	8	13	2	11	34	34\31/24
8- -13	2- -11	9	4	15	6	34	34\44/41
16---10	3---5	16	10	3	5	34	34\27/34

/E1	/E2	10/ST	Rw	CI\Pd/pd	119/ST	Rw	CI\Pd/pd
1---7	14---12	1	7	14	12	34	34\34/30
8---2	13---11	9	15	4	6	34	34\38/30
9- -15	4- -6	8	2	13	11	34	34\38/42
16---10	3---5	16	10	3	5	34	34\26/34

/E1	/E2	11/ST	Rw	CI\Pd/pd	118/PD	Rw	CI\Pd/pd
1---7	14---12	1	7	14	12	34	34\34/24
8---2	11---13	10	16	5	3	34	34\40/34
10- -16	5- -3	8	2	11	13	34	34\34/44
15---9	4---6	15	9	4	6	34	34\28/34

/E1	/E2	12/ST	Rw	CI\Pd/pd	117/ST	Rw	CI\Pd/pd
1---7	14---12	1	7	14	12	34	34\34/34
6---4	15---9	11	13	2	8	34	34\34/30
11- -13	2- -8	6	4	15	9	34	34\38/38
16---10	3---5	16	10	3	5	34	34\30/34

/E1	/E2	13/ST	Rw	CI\Pd/pd	116/ST	Rw	CI\Pd/pd
1---7	14---12	1	7	14	12	34	34\34/32
4---6	15---9	16	10	3	5	34	34\32/34
16- -10	3- -5	4	6	15	9	34	34\34/36
13---11	2---8	13	11	2	8	34	34\36/34

/E1	/E2	14/ST	Rw	CI\Pd/pd	189/ST	Rw	CI\Pd/pd
1---7	16---10	1	7	16	10	34	34\34/24
6---4	11---13	12	14	5	3	34	34\40/34
12- -14	5- -3	6	4	11	13	34	34\34/44
15---9	2---8	15	9	2	8	34	34\28/34

/E1	/E2	15/ST	Rw	CI\Pd/pd	188/ST	Rw	CI\Pd/pd
1---7	16---10	1	7	16	10	34	34\34/28
4---6	13---11	14	12	3	5	34	34\36/34
14- -12	3- -5	4	6	13	11	34	34\34/40
15---9	2---8	15	9	2	8	34	34\32/34

/E1	/E2	16/PD	Rw	CI\Pd/pd	52/ST	Rw	CI\Pd/pd
1---8	11---14	1	8	11	14	34	34\34/34
6---3	16---9	12	13	2	7	34	34\34/34
12- -13	2- -7	6	3	16	9	34	34\34/34
15---10	5---4	15	10	5	4	34	34\34/34

/E1	/E2	17/ST	Rw	CI\Pd/pd	67/ST	Rw	CI\Pd/pd
1---8	12---13	1	8	12	13	34	34\34/34
11---2	14---7	6	15	3	10	34	34\34/26
6- -15	3- -10	11	2	14	7	34	34\42/42
16---9	5---4	16	9	5	4	34	34\26/34

/E1	/E2	18/ST	Rw	CI\Pd/pd	66/ST	Rw	CI\Pd/pd
1---8	12---13	1	8	12	13	34	34\34/36
10---3	15---6	7	14	2	11	34	34\32/26
7- -14	2- -11	10	3	15	6	34	34\42/40
16---9	5---4	16	9	5	4	34	34\28/34

/E1	/E2	19/ST	Rw	CI\Pd/pd	65/ST	Rw	CI\Pd/pd
1---8	12---13	1	8	12	13	34	34\34/30
7---2	14---11	10	15	3	6	34	34\38/34
10- -15	3- -6	7	2	14	11	34	34\34/38
16---9	5---4	16	9	5	4	34	34\30/34

/E1	/E2	20/ST	Rw	CI\Pd/pd	64/ST	Rw	CI\Pd/pd
1---8	12---13	1	8	12	13	34	34\34/32
6---3	15---10	11	14	2	7	34	34\36/34
11- -14	2- -7	6	3	15	10	34	34\34/36
16---9	5---4	16	9	5	4	34	34\32/34

/E1	/E2	21/ST	Rw	CI\Pd/pd	98/ST	Rw	CI\Pd/pd
1---8	13---12	1	8	13	12	34	34\34/31
16---2	11---5	3	15	6	10	34	34\33/20
3- -15	6- -10	16	2	11	5	34	34\48/51
14---9	4---7	14	9	4	7	34	34\21/34

/E1	/E2	22/PD	Rw	CI\Pd/pd	96/ST	Rw	CI\Pd/pd
1---8	13---12	1	8	13	12	34	34\34/34
4---5	16---9	14	11	2	7	34	34\34/34
14- -11	2- -7	4	5	16	9	34	34\34/34
15---10	3---6	15	10	3	6	34	34\34/34

/E1	/E2	23/PD	Rw	CI\Pd/pd	97/ST	Rw	CI\Pd/pd
1---8	13---12	1	8	13	12	34	34\34/34
4---5	16---9	15	10	3	6	34	34\34/34
15- -10	3- -6	4	5	16	9	34	34\34/34
14---11	2---7	14	11	2	7	34	34\34/34



/E1	/E2	24/ST	Rw	CI\Pd/pd	95/ST	Rw	CI\Pd/pd
1---8	13---12	1 8 13 12		34 34\34/30	1 13 8 12		34 34\34/26
3---6	15---10	16 11 2 5		34 34\34/38	3 15 6 10		34 34\38/30
16- -11	2- -5	3 6 15 10		34 34\30/34	16 2 11 5		34 34\38/46
14---9	4---7	14 9 4 7		34 34\38/34	14 4 9 7		34 34\26/34

/E1	/E2	25/ST	Rw	CI\Pd/pd	125/ST	Rw	CI\Pd/pd
1---8	14---11	1 8 14 11		34 34\34/32	1 14 8 11		34 34\34/26
13---2	12---7	4 15 5 10		34 34\36/22	13 12 2 7		34 34\42/46
4- -15	5- -10	13 2 12 7		34 34\46/48	4 5 15 10		34 34\22/30
16---9	3---6	16 9 3 6		34 34\20/34	16 3 9 6		34 34\38/34

/E1	/E2	26/ST	Rw	CI\Pd/pd	124/ST	Rw	CI\Pd/pd
1---8	14---11	1 8 14 11		34 34\34/38	1 14 8 11		34 34\34/20
10---5	15---4	7 12 2 13		34 34\30/22	10 15 5 4		34 34\48/46
7- -12	2- -13	10 5 15 4		34 34\46/42	7 2 12 13		34 34\22/36
16---9	3---6	16 9 3 6		34 34\26/34	16 3 9 6		34 34\32/34

/E1	/E2	27/ST	Rw	CI\Pd/pd	123/ST	Rw	CI\Pd/pd
1---8	14---11	1 8 14 11		34 34\34/26	1 14 8 11		34 34\34/32
7---2	12---13	10 15 5 4		34 34\42/34	7 12 2 13		34 34\36/34
10- -15	5- -4	7 2 12 13		34 34\34/42	10 5 15 4		34 34\34/36
16---9	3---6	16 9 3 6		34 34\26/34	16 3 9 6		34 34\32/34

/E1	/E2	28/ST	Rw	CI\Pd/pd	122/ST	Rw	CI\Pd/pd
1---8	14---11	1 8 14 11		34 34\34/32	1 14 8 11		34 34\34/26
4---5	15---10	13 12 2 7		34 34\36/34	4 15 5 10		34 34\42/34
13- -12	2- -7	4 5 15 10		34 34\34/36	13 2 12 7		34 34\34/42
16---9	3---6	16 9 3 6		34 34\32/34	16 3 9 6		34 34\26/34

/E1	/E2	29/ST	Rw	CI\Pd/pd	153/ST	Rw	CI\Pd/pd
1---8	15---10	1 8 15 10		34 34\34/33	1 15 8 10		34 34\34/23
13---3	12---6	4 14 5 11		34 34\35/20	13 12 3 6		34 34\45/48
4- -14	5- -11	13 3 12 6		34 34\48/49	4 5 14 11		34 34\20/31
16---9	2---7	16 9 2 7		34 34\19/34	16 2 9 7		34 34\37/34

/E1	/E2	30/ST	Rw	CI\Pd/pd	152/ST	Rw	CI\Pd/pd
1---8	15---10	1 8 15 10		34 34\34/37	1 15 8 10		34 34\34/19
11---5	14---4	6 12 3 13		34 34\31/20	11 14 5 4		34 34\49/48
6- -12	3- -13	11 5 14 4		34 34\48/45	6 3 12 13		34 34\20/35
16---9	2---7	16 9 2 7		34 34\23/34	16 2 9 7		34 34\33/34

/E1	/E2	31/ST	Rw	CI\Pd/pd	151/ST	Rw	CI\Pd/pd
1---8	15---10	1 8 15 10		34 34\34/26	1 15 8 10		34 34\34/30
6---3	12---13	11 14 5 4		34 34\42/34	6 12 3 13		34 34\38/34
11- -14	5- -4	6 3 12 13		34 34\34/42	11 5 14 4		34 34\34/38
16---9	2---7	16 9 2 7		34 34\26/34	16 2 9 7		34 34\30/34

/E1	/E2	32/ST	Rw	CI\Pd/pd	150/ST	Rw	CI\Pd/pd
1---8	15---10	1 8 15 10		34 34\34/30	1 15 8 10		34 34\34/26
4---5	14---11	13 12 3 6		34 34\38/34	4 14 5 11		34 34\42/34
13- -12	3- -6	4 5 14 11		34 34\34/38	13 3 12 6		34 34\34/42
16---9	2---7	16 9 2 7		34 34\30/34	16 2 9 7		34 34\26/34

/E1	/E2	33/ST	Rw	CI\Pd/pd	193/ST	Rw	CI\Pd/pd
1---9	16---8	1 9 16 8		34 34\34/32	1 16 9 8		34 34\34/18
14---6	11---3	4 12 5 13		34 34\32/18	14 11 6 3		34 34\50/50
4- -12	5- -13	14 6 11 3		34 34\50/52	4 5 12 13		34 34\18/34
15---7	2---10	15 7 2 10		34 34\20/34	15 2 7 10		34 34\34/34

/E1	/E2	35/ST	Rw	CI\Pd/pd	192/ST	Rw	CI\Pd/pd
1---9	16---8	1 9 16 8		34 34\34/14	1 16 9 8		34 34\34/30
14---4	5---11	7 15 10 2		34 34\42/30	14 5 4 11		34 34\34/38
7- -15	10- -2	14 4 5 11		34 34\38/58	7 10 15 2		34 34\30/34
12---6	3---13	12 6 3 13		34 34\22/34	12 3 6 13		34 34\38/34

/E1	/E2	36/ST	Rw	CI\Pd/pd	191/ST	Rw	CI\Pd/pd
1---9	16---8	1 9 16 8		34 34\34/22	1 16 9 8		34 34\34/28
4---6	11---13	14 12 5 3		34 34\42/38	4 11 6 13		34 34\40/30
14- -12	5- -3	4 6 11 13		34 34\30/42	14 5 12 3		34 34\38/44
15---7	2---10	15 7 2 10		34 34\30/34	15 2 7 10		34 34\24/34

/E1	/E2	37/ST	Rw	CI\Pd/pd	190/ST	Rw	CI\Pd/pd
1---9	16---8	1 9 16 8		34 34\34/19	1 16 9 8		34 34\34/29
4---7	10---13	15 12 5 2		34 34\41/40	4 10 7 13		34 34\39/28
15- -12	5- -2	4 7 10 13		34 34\28/43	15 5 12 2		34 34\40/45
14---6	3---11	14 6 3 11		34 34\33/34	14 3 6 11		34 34\23/34

/E1	/E2	38/ST	Rw	CI\Pd/pd	162/ST	Rw	CI\Pd/pd
1---10	15---8	1 10 15 8		34 34\34/31	1 15 10 8		34 34\34/21
14---4	11---5	3 13 6 12		34 34\37/20	14 11 4 5		34 34\47/48
3- -13	6- -12	14 4 11 5		34 34\48/51	3 6 13 12		34 34\20/33
16---7	2---9	16 7 2 9		34 34\17/34	16 2 7 9		34 34\35/34

/E1	/E2	39/ST	Rw	CI\Pd/pd	160/ST	Rw	CI\Pd/pd
1---10	15---8	1 10 15 8		34 34\34/35	1 15 10 8		34 34\34/17
12---6	13---3	5 11 4 14		34 34\33/20	12 13 6 3		34 34\51/48
5- -11	4- -14	12 6 13 3		34 34\48/47	5 4 11 14		34 34\20/37
16---7	2---9	16 7 2 9		34 34\21/34	16 2 7 9		34 34\31/34

/E1	/E2	41/ST	Rw	CI\Pd/pd	159/PD	Rw	CI\Pd/pd
1---10	15---8	1 10 15 8		34 34\34/14	1 15 10 8		34 34\34/34
12---3	6---13	7 16 9 2		34 34\46/34	12 6 3 13		34 34\34/34
7- -16	9- -2	12 3 6 13		34 34\34/54	7 9 16 2		34 34\34/34
14---5	4---11	14 5 4 11		34 34\22/34	14 4 5 11		34 34\34/34

/E1	/E2	43/ST	Rw	CI\Pd/pd	156/ST	Rw	CI\Pd/pd
1---10	15---8	1 10 15 8		34 34\34/22	1 15 10 8		34 34\34/30
5---4	11---14	12 13 6 3		34 34\46/38	5 11 4 14		34 34\38/30
12- -13	6- -3	5 4 11 14		34 34\30/42	12 6 13 3		34 34\38/42
16---7	2---9	16 7 2 9		34 34\26/34	16 2 7 9		34 34\26/34

/E1	/E2	44/ST	Rw	CI\Pd/pd	154/ST	Rw	CI\Pd/pd
1---10	15---8	1 10 15 8		34 34\34/26	1 15 10 8		34 34\34/26
3---6	13---12	14 11 4 5		34 34\42/38	3 13 6 12		34 34\42/30
14- -11	4- -5	3 6 13 12		34 34\30/38	14 4 11 5		34 34\38/46
16---7	2---9	16 7 2 9		34 34\30/34	16 2 7 9		34 34\22/34

/E1	/E2	45/ST	Rw	CI\Pd/pd	157/ST	Rw	CI\Pd/pd
1---10	15---8	1 10 15 8		34 34\34/31	1 15 10 8		34 34\34/13
5---11	14---4	16 6 3 9		34 34\29/32	5 14 11 4		34 34\47/36
16- -6	3- -9	5 11 14 4		34 34\36/39	16 3 6 9		34 34\32/53
12---7	2---13	12 7 2 13		34 34\37/34	12 2 7 13		34 34\23/34

/E1	/E2	48/ST	Rw	CI\Pd/pd	197/ST	Rw	CI\Pd/pd
1---10	16---7	1 10 16 7		34 34\34/14	1 16 10 7		34 34\34/32
11---4	6---13	8 15 9 2		34 34\46/34	11 6 4 13		34 34\36/34
8- -15	9- -2	11 4 6 13		34 34\34/54	8 9 15 2		34 34\34/36
14---5	3---12	14 5 3 12		34 34\22/34	14 3 5 12		34 34\32/34

/E1	/E2	50/ST	Rw	CI\Pd/pd	195/ST	Rw	CI\Pd/pd
1---10	16--- 7	1 10 16	7	34 34\34/28	1 16 10	7	34 34\34/18
4---11	13--- 6	15 8 2	9	34 34\32/34	4 13 11	6	34 34\50/34
15- - 8	2- - 9	4 11 13	6	34 34\34/40	15 2 8	9	34 34\34/50
14--- 5	3---12	14 5 3	12	34 34\36/34	14 3 5	12	34 34\18/34

/E1	/E2	52/ST	Rw	CI\Pd/pd	16/PD	Rw	CI\Pd/pd
1---11	8---14	1 11 8	14	34 34\34/28	1 8 11	14	34 34\34/34
12--- 2	13--- 7	6 16 3	9	34 34\36/34	12 13 2	7	34 34\34/34
6- -16	3- - 9	12 2 13	7	34 34\34/40	6 3 16	9	34 34\34/34
15--- 5	10--- 4	15 5 10	4	34 34\32/34	15 10 5	4	34 34\34/34

/E1	/E2	55/ST	Rw	CI\Pd/pd	131/ST	Rw	CI\Pd/pd
1---11	14--- 8	1 11 14	8	34 34\34/14	1 14 11	8	34 34\34/26
16--- 4	5--- 9	7 13 12	2	34 34\42/30	16 5 4	9	34 34\30/38
7- -13	12- - 2	16 4 5	9	34 34\38/58	7 12 13	2	34 34\30/38
10--- 6	3---15	10 6 3	15	34 34\22/34	10 3 6	15	34 34\42/34

/E1	/E2	56/ST	Rw	CI\Pd/pd	128/ST	Rw	CI\Pd/pd
1---11	14--- 8	1 11 14	8	34 34\34/29	1 14 11	8	34 34\34/11
7---12	13--- 2	16 5 4	9	34 34\27/32	7 13 12	2	34 34\45/36
16- - 5	4- - 9	7 12 13	2	34 34\36/41	16 4 5	9	34 34\32/55
10--- 6	3---15	10 6 3	15	34 34\39/34	10 3 6	15	34 34\25/34

/E1	/E2	59/ST	Rw	CI\Pd/pd	201/ST	Rw	CI\Pd/pd
1---11	16--- 6	1 11 16	6	34 34\34/16	1 16 11	6	34 34\34/30
10--- 4	7---13	8 14 9	3	34 34\48/34	10 7 4	13	34 34\38/34
8- -14	9- - 3	10 4 7	13	34 34\34/52	8 9 14	3	34 34\34/38
15--- 5	2---12	15 5 2	12	34 34\20/34	15 2 5	12	34 34\30/34

/E1	/E2	61/ST	Rw	CI\Pd/pd	199/ST	Rw	CI\Pd/pd
1---11	16--- 6	1 11 16	6	34 34\34/28	1 16 11	6	34 34\34/18
4---10	13--- 7	14 8 3	9	34 34\36/34	4 13 10	7	34 34\50/34
14- - 8	3- - 9	4 10 13	7	34 34\34/40	14 3 8	9	34 34\34/50
15--- 5	2---12	15 5 2	12	34 34\32/34	15 2 5	12	34 34\18/34

/E1	/E2	63/PD	Rw	CI\Pd/pd	7/ST	Rw	CI\Pd/pd
1---12	7---14	1 12 7	14	34 34\34/34	1 7 12	14	34 34\34/28
10--- 3	16--- 5	8 13 2	11	34 34\34/34	10 16 3	5	34 34\36/34
8- -13	2- -11	10 3 16	5	34 34\34/34	8 2 13	11	34 34\34/40
15--- 6	9--- 4	15 6 9	4	34 34\34/34	15 9 6	4	34 34\32/34

/E1	/E2	64/ST	Rw	CI\Pd/pd	20/ST	Rw	CI\Pd/pd
1---12	8---13	1 12 8	13	34 34\34/30	1 8 12	13	34 34\34/32
11--- 2	14--- 7	6 15 3	10	34 34\38/34	11 14 2	7	34 34\36/34
6- -15	3- -10	11 2 14	7	34 34\34/38	6 3 15	10	34 34\34/36
16--- 5	9--- 4	16 5 9	4	34 34\30/34	16 9 5	4	34 34\32/34

/E1	/E2	65/ST	Rw	CI\Pd/pd	19/ST	Rw	CI\Pd/pd
1---12	8---13	1 12 8	13	34 34\34/32	1 8 12	13	34 34\34/30
10--- 3	15--- 6	7 14 2	11	34 34\36/34	10 15 3	6	34 34\38/34
7- -14	2- -11	10 3 15	6	34 34\34/36	7 2 14	11	34 34\34/38
16--- 5	9--- 4	16 5 9	4	34 34\32/34	16 9 5	4	34 34\30/34

/E1	/E2	66/ST	Rw	CI\Pd/pd	18/ST	Rw	CI\Pd/pd
1---12	8---13	1 12 8	13	34 34\34/26	1 8 12	13	34 34\34/36
7--- 2	14---11	10 15 3	6	34 34\42/42	7 14 2	11	34 34\32/26
10- -15	3- - 6	7 2 14	11	34 34\26/34	10 3 15	6	34 34\42/40
16--- 5	9--- 4	16 5 9	4	34 34\34/34	16 9 5	4	34 34\28/34

/E1	/E2	67/ST	Rw	CI\Pd/pd	17/ST	Rw	CI\Pd/pd
1---12	8---13	1 12 8 13		34 34\34/28	1 8 12 13		34 34\34/34
6--- 3	15---10	11 14 2 7		34 34\40/42	6 15 3 10		34 34\34/26
11- -14	2- - 7	6 3 15 10		34 34\26/32	11 2 14 7		34 34\42/42
16--- 5	9--- 4	16 5 9 4		34 34\36/34	16 9 5 4		34 34\26/34

/E1	/E2	68/ST	Rw	CI\Pd/pd	106/ST	Rw	CI\Pd/pd
1---12	13--- 8	1 12 13 8		34 34\34/27	1 13 12 8		34 34\34/25
15--- 3	10--- 6	2 14 7 11		34 34\41/24	15 10 3 6		34 34\43/44
2- -14	7- -11	15 3 10 6		34 34\44/51	2 7 14 11		34 34\24/33
16--- 5	4--- 9	16 5 4 9		34 34\17/34	16 4 5 9		34 34\35/34

/E1	/E2	69/ST	Rw	CI\Pd/pd	109/ST	Rw	CI\Pd/pd
1---12	13--- 8	1 12 13 8		34 34\34/19	1 13 12 8		34 34\34/29
16--- 2	7--- 9	3 15 10 6		34 34\45/28	16 7 2 9		34 34\35/40
3- -15	10- - 6	16 2 7 9		34 34\40/55	3 10 15 6		34 34\28/33
14--- 5	4---11	14 5 4 11		34 34\17/34	14 4 5 11		34 34\39/34

/E1	/E2	70/ST	Rw	CI\Pd/pd	105/ST	Rw	CI\Pd/pd
1---12	13--- 8	1 12 13 8		34 34\34/35	1 13 12 8		34 34\34/17
11--- 7	14--- 2	6 10 3 15		34 34\33/24	11 14 7 2		34 34\51/44
6- -10	3- -15	11 7 14 2		34 34\44/43	6 3 10 15		34 34\24/41
16--- 5	4--- 9	16 5 4 9		34 34\25/34	16 4 5 9		34 34\27/34

/E1	/E2	72/ST	Rw	CI\Pd/pd	108/ST	Rw	CI\Pd/pd
1---12	13--- 8	1 12 13 8		34 34\34/10	1 13 12 8		34 34\34/30
16--- 5	4--- 9	7 14 11 2		34 34\42/34	16 4 5 9		34 34\30/34
7- -14	11- - 2	16 5 4 9		34 34\34/58	7 11 14 2		34 34\34/38
10--- 3	6---15	10 3 6 15		34 34\26/34	10 6 3 15		34 34\38/34

/E1	/E2	73/ST	Rw	CI\Pd/pd	104/ST	Rw	CI\Pd/pd
1---12	13--- 8	1 12 13 8		34 34\34/18	1 13 12 8		34 34\34/34
6--- 3	10---15	11 14 7 2		34 34\50/42	6 10 3 15		34 34\34/26
11- -14	7- - 2	6 3 10 15		34 34\26/42	11 7 14 2		34 34\42/42
16--- 5	4--- 9	16 5 4 9		34 34\26/34	16 4 5 9		34 34\26/34

/E1	/E2	74/PD	Rw	CI\Pd/pd	102/ST	Rw	CI\Pd/pd
1---12	13--- 8	1 12 13 8		34 34\34/34	1 13 12 8		34 34\34/16
4--- 9	16--- 5	14 7 2 11		34 34\34/34	4 16 9 5		34 34\48/34
14- - 7	2- -11	4 9 16 5		34 34\34/34	14 2 7 11		34 34\34/52
15--- 6	3---10	15 6 3 10		34 34\34/34	15 3 6 10		34 34\20/34

/E1	/E2	75/PD	Rw	CI\Pd/pd	103/ST	Rw	CI\Pd/pd
1---12	13--- 8	1 12 13 8		34 34\34/34	1 13 12 8		34 34\34/14
4--- 9	16--- 5	15 6 3 10		34 34\34/34	4 16 9 5		34 34\46/34
15- - 6	3- -10	4 9 16 5		34 34\34/34	15 3 6 10		34 34\34/54
14--- 7	2---11	14 7 2 11		34 34\34/34	14 2 7 11		34 34\22/34

/E1	/E2	77/ST	Rw	CI\Pd/pd	101/ST	Rw	CI\Pd/pd
1---12	13--- 8	1 12 13 8		34 34\34/30	1 13 12 8		34 34\34/18
3---10	15--- 6	16 7 2 9		34 34\34/38	3 15 10 6		34 34\46/30
16- - 7	2- - 9	3 10 15 6		34 34\30/34	16 2 7 9		34 34\38/54
14--- 5	4---11	14 5 4 11		34 34\38/34	14 4 5 11		34 34\18/34

/E1	/E2	79/ST	Rw	CI\Pd/pd	136/ST	Rw	CI\Pd/pd
1---12	14--- 7	1 12 14 7		34 34\34/20	1 14 12 7		34 34\34/30
13--- 2	8---11	4 15 9 6		34 34\48/30	13 8 2 11		34 34\38/38
4- -15	9- - 6	13 2 8 11		34 34\38/52	4 9 15 6		34 34\30/34
16--- 5	3---10	16 5 3 10		34 34\16/34	16 3 5 10		34 34\34/34

$/E1$	$/E2$	80/ST	Rw	CI\Pd/pd	135/ST	Rw	CI\Pd/pd
1---12	14--- 7	1 12 14 7	34	34\34/18	1 14 12 7	34	34\34/32
11--- 2	8---13	6 15 9 4	34	34\50/34	11 8 2 13	34	34\36/34
6- -15	9- - 4	11 2 8 13	34	34\34/50	6 9 15 4	34	34\34/36
16--- 5	3---10	16 5 3 10	34	34\18/34	16 3 5 10	34	34\32/34
$/E1$	$/E2$	81/SC	Rw	CI\Pd/pd	137/SC	Rw	CI\Pd/pd
1---12	14--- 7	1 12 14 7	34	34\34/10	1 14 12 7	34	34\34/28
15--- 6	4--- 9	8 13 11 2	34	34\42/34	15 4 6 9	34	34\32/34
8- -13	11- - 2	15 6 4 9	34	34\34/58	8 11 13 2	34	34\34/40
10--- 3	5---16	10 3 5 16	34	34\26/34	10 5 3 16	34	34\36/34
$/E1$	$/E2$	82/ST	Rw	CI\Pd/pd	133/ST	Rw	CI\Pd/pd
1---12	14--- 7	1 12 14 7	34	34\34/34	1 14 12 7	34	34\34/16
6--- 9	15--- 4	11 8 2 13	34	34\34/30	6 15 9 4	34	34\52/38
11- - 8	2- -13	6 9 15 4	34	34\38/38	11 2 8 13	34	34\30/48
16--- 5	3---10	16 5 3 10	34	34\30/34	16 3 5 10	34	34\20/34
$/E1$	$/E2$	83/ST	Rw	CI\Pd/pd	132/ST	Rw	CI\Pd/pd
1---12	14--- 7	1 12 14 7	34	34\34/32	1 14 12 7	34	34\34/18
4--- 9	15--- 6	13 8 2 11	34	34\36/34	4 15 9 6	34	34\50/34
13- - 8	2- -11	4 9 15 6	34	34\34/36	13 2 8 11	34	34\34/50
16--- 5	3---10	16 5 3 10	34	34\32/34	16 3 5 10	34	34\18/34
$/E1$	$/E2$	84/SC	Rw	CI\Pd/pd	134/SC	Rw	CI\Pd/pd
1---12	14--- 7	1 12 14 7	34	34\34/24	1 14 12 7	34	34\34/14
8---13	11--- 2	15 6 4 9	34	34\28/34	8 11 13 2	34	34\46/34
15- - 6	4- - 9	8 13 11 2	34	34\34/44	15 4 6 9	34	34\34/54
10--- 3	5---16	10 3 5 16	34	34\40/34	10 5 3 16	34	34\22/34
$/E1$	$/E2$	85/ST	Rw	CI\Pd/pd	173/ST	Rw	CI\Pd/pd
1---12	15--- 6	1 12 15 6	34	34\34/25	1 15 12 6	34	34\34/23
14--- 4	9--- 7	3 13 8 10	34	34\43/24	14 9 4 7	34	34\45/44
3- -13	8- -10	14 4 9 7	34	34\44/53	3 8 13 10	34	34\24/35
16--- 5	2---11	16 5 2 11	34	34\15/34	16 2 5 11	34	34\33/34
$/E1$	$/E2$	86/ST	Rw	CI\Pd/pd	171/ST	Rw	CI\Pd/pd
1---12	15--- 6	1 12 15 6	34	34\34/21	1 15 12 6	34	34\34/27
13--- 3	8---10	4 14 9 7	34	34\47/28	13 8 3 10	34	34\41/40
4- -14	9- - 7	13 3 8 10	34	34\40/53	4 9 14 7	34	34\28/35
16--- 5	2---11	16 5 2 11	34	34\15/34	16 2 5 11	34	34\33/34
$/E1$	$/E2$	87/ST	Rw	CI\Pd/pd	170/ST	Rw	CI\Pd/pd
1---12	15--- 6	1 12 15 6	34	34\34/33	1 15 12 6	34	34\34/15
10--- 8	13--- 3	7 9 4 14	34	34\35/24	10 13 8 3	34	34\53/44
7- - 9	4- -14	10 8 13 3	34	34\44/45	7 4 9 14	34	34\24/43
16--- 5	2---11	16 5 2 11	34	34\23/34	16 2 5 11	34	34\25/34
$/E1$	$/E2$	89/SC	Rw	CI\Pd/pd	172/SC	Rw	CI\Pd/pd
1---12	15--- 6	1 12 15 6	34	34\34/10	1 15 12 6	34	34\34/28
14--- 7	4--- 9	8 13 10 3	34	34\42/34	14 4 7 9	34	34\36/34
8- -13	10- - 3	14 7 4 9	34	34\34/58	8 10 13 3	34	34\34/40
11--- 2	5---16	11 2 5 16	34	34\26/34	11 5 2 16	34	34\32/34
$/E1$	$/E2$	90/ST	Rw	CI\Pd/pd	167/ST	Rw	CI\Pd/pd
1---12	15--- 6	1 12 15 6	34	34\34/33	1 15 12 6	34	34\34/15
7--- 9	14--- 4	10 8 3 13	34	34\35/28	7 14 9 4	34	34\53/40
10- - 8	3- -13	7 9 14 4	34	34\40/41	10 3 8 13	34	34\28/47
16--- 5	2---11	16 5 2 11	34	34\27/34	16 2 5 11	34	34\21/34

/E1	/E2	92/ST	Rw	CI\Pd/pd	165/ST	Rw	CI\Pd/pd
1---12	15--- 6	1 12 15 6		34 34\34/30	1 15 12 6		34 34\34/18
4--- 9	14--- 7	13 8 3 10		34 34\38/34	4 14 9 7		34 34\50/34
13- - 8	3- -10	4 9 14 7		34 34\34/38	13 3 8 10		34 34\34/50
16--- 5	2---11	16 5 2 11		34 34\30/34	16 2 5 11		34 34\18/34

/E1	/E2	93/SC	Rw	CI\Pd/pd	168/SC	Rw	CI\Pd/pd
1---12	15--- 6	1 12 15 6		34 34\34/22	1 15 12 6		34 34\34/16
8---13	10--- 3	14 7 4 9		34 34\30/34	8 10 13 3		34 34\48/34
14- - 7	4- - 9	8 13 10 3		34 34\34/46	14 4 7 9		34 34\34/52
11--- 2	5---16	11 2 5 16		34 34\38/34	11 5 2 16		34 34\20/34

/E1	/E2	95/ST	Rw	CI\Pd/pd	24/ST	Rw	CI\Pd/pd
1---13	8---12	1 13 8 12		34 34\34/26	1 8 13 12		34 34\34/30
16--- 2	11--- 5	3 15 6 10		34 34\38/30	16 11 2 5		34 34\34/38
3- -15	6- -10	16 2 11 5		34 34\38/46	3 6 15 10		34 34\30/34
14--- 4	9--- 7	14 4 9 7		34 34\26/34	14 9 4 7		34 34\38/34

/E1	/E2	116/ST	Rw	CI\Pd/pd	13/ST	Rw	CI\Pd/pd
1---14	7---12	1 14 7 12		34 34\34/22	1 7 14 12		34 34\34/32
16--- 3	10--- 5	4 15 6 9		34 34\38/34	16 10 3 5		34 34\32/34
4- -15	6- - 9	16 3 10 5		34 34\34/46	4 6 15 9		34 34\34/36
13--- 2	11--- 8	13 2 11 8		34 34\30/34	13 11 2 8		34 34\36/34

/E1	/E2	148/ST	Rw	CI\Pd/pd	4/ST	Rw	CI\Pd/pd
1---15	6---12	1 15 6 12		34 34\34/24	1 6 15 12		34 34\34/30
16--- 2	11--- 5	4 14 7 9		34 34\40/34	16 11 2 5		34 34\30/34
4- -14	7- - 9	16 2 11 5		34 34\34/44	4 7 14 9		34 34\34/38
13--- 3	10--- 8	13 3 10 8		34 34\28/34	13 10 3 8		34 34\38/34

/E1	/E2	184/ST	Rw	CI\Pd/pd	2/ST	Rw	CI\Pd/pd
1---16	5---12	1 16 5 12		34 34\34/26	1 5 16 12		34 34\34/32
10--- 3	14--- 7	8 13 4 9		34 34\42/42	10 14 3 7		34 34\32/26
8- -13	4- - 9	10 3 14 7		34 34\26/34	8 4 13 9		34 34\42/44
15--- 2	11--- 6	15 2 11 6		34 34\34/34	15 11 2 6		34 34\28/34

/E1	/E2	209/ST	Rw	CI\Pd/pd	353/ST	Rw	CI\Pd/pd
2--- 5	15---12	2 5 15 12		34 34\34/22	2 15 5 12		34 34\34/36
8--- 3	9---14	11 16 6 1		34 34\38/34	8 9 3 14		34 34\32/34
11- -16	6- - 1	8 3 9 14		34 34\34/46	11 6 16 1		34 34\34/32
13---10	4--- 7	13 10 4 7		34 34\30/34	13 4 10 7		34 34\36/34

/E1	/E2	409/ST	Rw	CI\Pd/pd	554/ST	Rw	CI\Pd/pd
3--- 2	16---13	3 2 16 13		34 34\34/36	3 16 2 13		34 34\34/30
9--- 6	12--- 7	8 15 1 10		34 34\24/22	9 12 6 7		34 34\46/46
8- -15	1- -10	9 6 12 7		34 34\46/44	8 1 15 10		34 34\22/26
14---11	5--- 4	14 11 5 4		34 34\32/34	14 5 11 4		34 34\34/34

/E1	/E2	575/ST	Rw	CI\Pd/pd	689/ST	Rw	CI\Pd/pd
4--- 5	14---11	4 5 14 11		34 34\34/35	4 14 5 11		34 34\34/29
16--- 2	9--- 7	1 15 8 10		34 34\33/16	16 9 2 7		34 34\39/52
1- -15	8- -10	16 2 9 7		34 34\52/51	1 8 15 10		34 34\16/21
13---12	3--- 6	13 12 3 6		34 34\17/34	13 3 12 6		34 34\47/34

/E1	/E2	753/ST	Rw	CI\Pd/pd	805/ST	Rw	CI\Pd/pd
5--- 2	16---11	5 2 16 11		34 34\34/34	5 16 2 11		34 34\34/32
13---10	8--- 3	4 15 1 14		34 34\18/18	13 8 10 3		34 34\52/50
4- -15	1- -14	13 10 8 3		34 34\50/50	4 1 15 14		34 34\18/20
12--- 7	9--- 6	12 7 9 6		34 34\34/34	12 9 7 6		34 34\32/34

/E1	/E2	817/ST Rw CI\Pd/pd				860/ST Rw CI\Pd/pd											
6---3	15---10	6	3	15	10	34	34	34	42	6	15	3	10	34	34	34	24
16---5	9---4	1	12	8	13	34	34	26	10	16	9	5	4	34	34	44	58
1- -12	8- -13	16	5	9	4	34	34	58	50	1	8	12	13	34	34	10	20
11---14	2---7	11	14	2	7	34	34	18	34	11	2	14	7	34	34	48	34

/E1	/E2	865/ST Rw CI\Pd/pd				880/ST Rw CI\Pd/pd											
7---2	16---9	7	2	16	9	34	34	34	26	7	16	2	9	34	34	34	40
11---14	4---5	6	15	1	12	34	34	18	26	11	4	14	5	34	34	52	42
6- -15	1- -12	11	14	4	5	34	34	42	50	6	1	15	12	34	34	26	20
10---3	13---8	10	3	13	8	34	34	42	34	10	13	3	8	34	34	24	34

[Solution Counts(n1=1/All) = 208/880] OK!

\*\* Calculated and Listed by Kanji Setsuda on  
Jan. 19, 2014 with Apple OSX 10.9.1 & Xcode 5.0.2 \*\*

確かに標準型四角陣の解 880 個が得られた。この中には、補数対称型 48 個と汎対角線型 48 個（同時に完全型，あるいは正方連結型完全型でもある）が当然含まれている。そして，試した二-三変換法では，他の 880 個に確かに変換された。しかし，全体として見ると，同じ標準解の集合が再構成されている。内部では，一切重複が無かった。この結果は，2つの解集合が，共に同じ四次元超立体二角陣の「展開形」であることを示している。即ち，原点 n1 の隣の数が n2, n3, n5, n9 の 4 個であることの証左である。

2. 新構成法本来の特長が発揮されるのは，やはり次のような魔方陣を作る場合だ。次は，補数対称型四角陣の原始解を多数個，同時に作ってみよう。

2-1. 与える基本図と定義条件式は次の通り。

二次元四角陣の展開形を何と 24 個も用意した。これは，第 7 章での研究成果を反映している。四次元二角陣の n1 = 1 の時の「展開見取り図」が 24 個もあるためだ。

1#/E1	/E2	T01/		T02/		3#/E1	/E2	T03/		T04/									
1---2	3---4	1	2	3	4	1	2	3	4	1---2	5---6	1	2	5	6	1	2	5	6
9---10	11---12	5	6	7	8	9	10	11	12	9---10	13---14	3	4	7	8	9	10	13	14
5- -6	7- -8	9	10	11	12	5	6	7	8	3- -4	7- -8	9	10	13	14	3	4	7	8
13---14	15---16	13	14	15	16	13	14	15	16	11---12	15---16	11	12	15	16	11	12	15	16
5#/E1	/E2	T05/		T06/		7#/E1	/E2	T07/		T08/									
1---2	9---10	1	2	9	10	1	2	9	10	1---3	2---4	1	3	2	4	1	3	2	4
5---6	13---14	3	4	11	12	5	6	13	14	9---11	10---12	5	7	6	8	9	11	10	12
3- -4	11- -12	5	6	13	14	3	4	11	12	5- -7	6- -8	9	11	10	12	5	7	6	8
7---8	15---16	7	8	15	16	7	8	15	16	13---15	14---16	13	15	14	16	13	15	14	16
9#/E1	/E2	T09/		T10/		11#/E1	/E2	T11/		T12/									
1---3	5---7	1	3	5	7	1	3	5	7	1---3	9---11	1	3	9	11	1	3	9	11
9---11	13---15	2	4	6	8	9	11	13	15	5---7	13---15	2	4	10	12	5	7	13	15
2- -4	6- -8	9	11	13	15	2	4	6	8	2- -4	10- -12	5	7	13	15	2	4	10	12
10---12	14---16	10	12	14	16	10	12	14	16	6---8	14---16	6	8	14	16	6	8	14	16
13#/E1	/E2	T13/		T14/		15#/E1	/E2	T15/		T16/									
1---5	2---6	1	5	2	6	1	5	2	6	1---5	3---7	1	5	3	7	1	5	3	7
9---13	10---14	3	7	4	8	9	13	10	14	9---13	11---15	2	6	4	8	9	13	11	15
3- -7	4- -8	9	13	10	14	3	7	4	8	2- -6	4- -8	9	13	11	15	2	6	4	8
11---15	12---16	11	15	12	16	11	15	12	16	10---14	12---16	10	14	12	16	10	14	12	16
17#/E1	/E2	T17/		T18/		19#/E1	/E2	T19/		T20/									
1---5	9---13	1	5	9	13	1	5	9	13	1---9	2---10	1	9	2	10	1	9	2	10
3---7	11---15	2	6	10	14	3	7	11	15	5---13	6---14	3	11	4	12	5	13	6	14
2- -6	10- -14	3	7	11	15	2	6	10	14	3- -11	4- -12	5	13	6	14	3	11	4	12
4---8	12---16	4	8	12	16	4	8	12	16	7---15	8---16	7	15	8	16	7	15	8	16

21#/E1	/E2	T21/	T22/	23#/E1	/E2	T23/	T24/
1---9	3---11	1 9 3 11	1 9 3 11	1---9	5---13	1 9 5 13	1 9 5 13
5---13	7---15	2 10 4 12	5 13 7 15	3---11	7---15	2 10 6 14	3 11 7 15
2- -10	4- -12	5 13 7 15	2 10 4 12	2- -10	6- -14	3 11 7 15	2 10 6 14
6---14	8---16	6 14 8 16	6 14 8 16	4---12	8---16	4 12 8 16	4 12 8 16

\* 補数対称四方陣の定義式 :  $C=34$  \*

```

n1+n2+n3+n4=C;      n1+n2+n5+n6=C;      n1+n2+n9+n10=C;
n1+n3+n5+n7=C;      n1+n3+n9+n11=C;      n1+n5+n9+n13=C;
n2+n4+n6+n8=C;      n2+n4+n10+n12=C;     n2+n6+n10+n14=C;
n3+n4+n7+n8=C;      n3+n4+n11+n12=C;     n3+n7+n11+n15=C;
n4+n8+n12+n16=C;    n5+n6+n7+n8=C;      n5+n6+n13+n14=C;
n5+n7+n13+n15=C;    n6+n8+n14+n16=C;     n7+n8+n15+n16=C;
n9+n10+n11+n12=C;   n9+n10+n13+n14=C;    n9+n11+n13+n15=C;
n10+n12+n14+n16=C;  n11+n12+n15+n16=C;   n13+n14+n15+n16=C;
n1+n6+n11+n16=C    ...pd1;      n4+n7+n10+n13=C ...pd2;

```

\* 対称型補数の条件式 :  $CC=17$  \*

```
n1+n16=n2+n15=n3+n14=n4+n13=n5+n12=n6+n11=n7+n10=n8+n9=CC=C/2 ...cc;
```

普通なら定義条件式はこんなに多数は必要ないのだが、四次元二方陣に敬意を表して、可能な全条件を並べて見た。全部立てても、出来る目的方陣に変わりはない。

2-2. では、補数対称型四方陣を目的方陣と定めて、それを実際に作ってみよう。上の基本図と条件式からプログラムを次のように書いた。

```

/** Make the 4-D Extra-Cubic Magic Objects of Order 2 and **
/** Transform them into Self-Complementary Magic Squares 4x4 **
/** File: 'EC024SC.c' Dictated by Kanji Setsuda **
/** in 2005, 2012 and on May 5, 2014 **
/** with Apple OSX 10.9.2 & Xcode 5.1.1; **
//
#include <stdio.h>
//
/** Variables *
short int cnt, cnt1, cnt3;
short LSM, CC;
short nm[17], flg[17];
short rnm[385][17];
short an[7][18];
short t[17], s[17];
short tn[25][17];
short cb[3][9], cp[3][9];
short css[3];
//
/** Sub-Routines *
void stp01(void), stp02(void), stp03(void);
void stp04(void), stp05(void), stp06(void);
void stp07(void), stp08(void), stp09(void);
void savesol(void);
void prlist(void);
void pr6x4trns(void);
void prtans(short x);
void anschksm(void);
void ans2chksm(short x);
void checksums(short x);
void trns24(void), pr24trns(void);
void pr4t(short x);
short fndon(short x);
//
void trnsf01(void), trnsf02(void), trnsf03(void), trnsf04(void);
void trnsf05(void), trnsf06(void), trnsf07(void), trnsf08(void);
void trnsf09(void), trnsf10(void), trnsf11(void), trnsf12(void);
void trnsf13(void), trnsf14(void), trnsf15(void), trnsf16(void);

```



```

void trnsf17(void), trnsf18(void), trnsf19(void), trnsf20(void);
void trnsf21(void), trnsf22(void), trnsf23(void), trnsf24(void);
//
/** Main Program *
int main(){
  short n;
  pr24trns();
  printf("\n");
  printf("*** Make the 4-D Extra-Cubic Magic Objects of Order 2 and\n");
  printf("  Transform them into Self-Complementary Magic Squares 4x4 **\n");
  printf(" [Primitive Solutions: 4-D EC02^4; Self-Complementary MS44;]\n");
  for(n=0;n<17;n++){nm[n]=0; flg[n]=0;}
  LSM=34; CC=17; cnt=0; cnt1=0; cnt3=0;
  stp01();      /* Reconstruction of Object Solutions *
  prlist();     /* Print the Solution List *
  if(cnt3>0){ans2chksm(cnt3);}
  printf(" [Solution Counts(n1=1/All) = %d/%d] OK!\n",cnt1,cnt);
  printf("\n");
  printf("*** Calculated and Listed by Kanji Setsuda on\n");
  printf("  May 5, 2014 with Apple OSX 10.9.2 & Xcode 5.1.1 **\n");
  printf("\n");
  return 0;
}
//
/** Reconstructions of Object Solutions *
/** Set n1 & n16 *
void stp01(){
  short a,b;
  for(a=1;a<17;a++){b=CC-a;
    if((flg[a]==0)&&(flg[b]==0)){
      nm[1]=a; nm[16]=b;
      flg[a]=1; flg[b]=1;
      stp02();
      flg[b]=0; flg[a]=0;}
  }
}
/** Set n2 & n15 *
void stp02(){
  short a,b;
  for(a=1;a<17;a++){b=CC-a;
    if((flg[a]==0)&&(flg[b]==0)){
      nm[2]=a; nm[15]=b;
      flg[a]=1; flg[b]=1;
      stp03();
      flg[b]=0; flg[a]=0;}
  }
}
/** Set n3 & n14 *
void stp03(){
  short a,b;
  for(a=1;a<17;a++){b=CC-a;
    if((flg[a]==0)&&(flg[b]==0)){
      nm[3]=a; nm[14]=b;
      flg[a]=1; flg[b]=1;
      stp04();
      flg[b]=0; flg[a]=0;}
  }
}
/** Set n4=LSM-n1-n2-n3 & n13 *
void stp04(){
  short a,b;
  a=LSM-nm[1]-nm[2]-nm[3];
  b=LSM-nm[16]-nm[15]-nm[14];
  if((0<a)&&(a<17)&&(a+b==CC)){
    if((flg[a]==0)&&(flg[b]==0)){
      nm[4]=a; nm[13]=b;

```

```

        flg[a]=1; flg[b]=1;
        stp05();
        flg[b]=0; flg[a]=0;}}
}
/** Set n5 & n12 *
void stp05(){
    short a,b;
    for(a=1;a<17;a++){b=CC-a;
        if((flg[a]==0)&&(flg[b]==0)){
            nm[5]=a; nm[12]=b;
            flg[a]=1; flg[b]=1;
            stp06();
            flg[b]=0; flg[a]=0;}
        }
}
/** Set n9=LSM-n1-n5-n13 & n8 *
void stp06(){
    short a,b;
    a=LSM-nm[1]-nm[5]-nm[13];
    b=LSM-nm[16]-nm[12]-nm[4];
    if((0<a)&&(a<17)&&(a+b==CC)){
        if((flg[a]==0)&&(flg[b]==0)){
            nm[9]=a; nm[8]=b;
            flg[a]=1; flg[b]=1;
            stp07();
            flg[b]=0; flg[a]=0;}}
}
/** Set n6=LSM-n1-n2-n5 & n11 *
void stp07(){
    short a,b,sm;
    a=LSM-nm[1]-nm[2]-nm[5];
    b=LSM-nm[16]-nm[15]-nm[12];
    if((0<a)&&(a<17)&&(a+b==CC)){
        if((flg[a]==0)&&(flg[b]==0)){
            nm[6]=a; nm[11]=b;
            flg[a]=1; flg[b]=1;
            sm=nm[1]+nm[6]+nm[11]+nm[16];
            if(sm==LSM){stp08();}
            flg[b]=0; flg[a]=0;}}
}
/** Set n7=LSM-n5-n6-n8 & n10 *
void stp08(){
    short a,b,c,d,sm;
    a=LSM-nm[5]-nm[6]-nm[8];
    b=LSM-nm[12]-nm[11]-nm[9];
    c=LSM-nm[3]-nm[11]-nm[15];
    d=LSM-nm[14]-nm[6]-nm[2];
    if((0<a)&&(a<17)&&(a+b==CC)){
        if((a==c)&&(flg[a]==0)&&(flg[b]==0)){
            nm[7]=a; nm[10]=b;
            flg[a]=1; flg[b]=1;
            sm=nm[4]+nm[7]+nm[10]+nm[13];
            if(sm==LSM){stp09();}
            flg[b]=0; flg[a]=0;}}
}
/** Check sums of other lines *
void stp09(){
    short sm1,sm2,sm3,sm4;
    sm1=nm[1]+nm[2]+nm[9]+nm[10];
    sm2=nm[1]+nm[3]+nm[5]+nm[7];
    sm3=nm[1]+nm[3]+nm[9]+nm[11];
    sm4=nm[5]+nm[7]+nm[13]+nm[15];
    if((sm1==LSM)&&(sm2==LSM)&&(sm3==LSM)&&(sm4==LSM)){savesol();}
}
//
/** Save each Solution to the Reference Table *

```

```

void savesol(){
    short n;
    rnm[cnt][0]=cnt+1;
    for(n=1;n<17;n++){rnm[cnt][n]=nm[n];}
    cnt++; if(nm[1]==1){cnt1++;}
}
//
/* Print the Solution List *
void prlist(){
    short m,n,c,d;
    d=0;
    for(m=0;m<cnt;m++){
        if(m>=24){c=rnm[m][1];}
        else{c=(rnm[m][1]*16+rnm[m][2])*16+rnm[m][5];}
        if(c!=d){d=c;
            for(n=0;n<17;n++){an[cnt3][n]=rnm[m][n];}
            checksums(cnt3);
            if(m<12){anschksm(); pr6x4trns();}
            else{cnt3++;
                if(cnt3==2){ans2chksm(cnt3); cnt3=0;}}
        }
    }
}
//
/* Find Old Number from the Reference List *
short fndon(short x){
    short m,n,f=0;
    for(m=0;m<cnt;m++){
        f=0;
        for(n=1;n<17;n++){
            if(an[x][n]==rnm[m][n]){f++;}
            else{break;}
        }
        if(f==16){f=rnm[m][0]; break;}
    }
    return f;
}
//
/* Print the 24 Transformations for each Answer *
void pr6x4trns(){
    short m,n;
    trns24();
    cnt3=0;
    for(m=0;m<24;m++){
        an[cnt3][0]=m+1;
        for(n=1;n<17;n++){an[cnt3][n]=tn[m][n];}
        an[cnt3][17]=fndon(cnt3);
        cnt3++;
        if(cnt3==6){prtans(cnt3); cnt3=0;}
    }
}
//
/* Print 6 Trnsfs *
void prtans(short x){
    short m,n,l,l4;
    printf(" ");
    for(m=0;m<x;m++){
        printf("T%02d:%6d/",an[m][0],an[m][17]);
        if(m+1<x){printf(" ");}
    }
    printf("\n");
    for(l=0;l<4;l++){l4=l*4;
        printf(" ");
        for(m=0;m<x;m++){
            for(n=1;n<5;n++){printf("%3d",an[m][l4+n]);}
            if(m+1<x){printf(" ");}
        }
    }
}

```

```

    }
    printf("\n");}
};
//
/** Print Single Answers with precise Check-sums *
void anschksm(){
    printf("\n");
    printf("%6d/E1          /E2          ",an[0][0]);
    if(css[0]==16){printf("/SC");}
    else{printf(" ");}
    printf(" |Rw|Cl\\Pd/pd|\n");
    printf("%4d---%2d%6d---%2d",an[0][1],an[0][2],an[0][3],an[0][4]);
    printf("%7d%3d%3d%3d|%2d|%2d\\%2d/%2d|\n",
        an[0][1],an[0][2],an[0][3],an[0][4],cb[0][1],cb[0][5],cp[0][1],cp[0][5]);
    printf(" |%2d---%2d |%2d---%2d",an[0][9],an[0][10],an[0][11],an[0][12]);
    printf("%5d%3d%3d%3d|%2d|%2d\\%2d/%2d|\n",
        an[0][5],an[0][6],an[0][7],an[0][8],cb[0][2],cb[0][6],cp[0][2],cp[0][6]);
    printf("%4d-l-%2d |%4d-l-%2d |",an[0][5],an[0][6],an[0][7],an[0][8]);
    printf("%5d%3d%3d%3d|%2d|%2d\\%2d/%2d|\n",
        an[0][9],an[0][10],an[0][11],an[0][12],cb[0][3],cb[0][7],cp[0][3],cp[0][7]);
    printf("%6d---%2d%6d---%2d",an[0][13],an[0][14],an[0][15],an[0][16]);
    printf("%5d%3d%3d%3d|%2d|%2d\\%2d/%2d|\n",
        an[0][13],an[0][14],an[0][15],an[0][16],cb[0][4],cb[0][8],cp[0][4],cp[0][8]);
}
//
/** Print Two Answers with precise Check-sums *
void ans2chksm(short x){
    short m;
    printf("\n");
    for(m=0;m<x;m++){
        printf("%5d/E1          /E2          ",an[m][0]);
        if(css[m]==16){printf("/SC");}
        else{printf(" ");}
        printf("\\%2d/%2d|",cp[m][1],cp[m][8]);
        if(m==0){printf(" ");}
    }
    printf("\n");
    for(m=0;m<x;m++){
        printf("%3d---%2d%6d---%2d",an[m][1],an[m][2],an[m][3],an[m][4]);
        printf("%6d%3d%3d%3d|%2d|%2d|",
            an[m][1],an[m][2],an[m][3],an[m][4],cb[m][1],cb[m][5]);
        if(m==0){printf(" ");}
    }
    printf("\n");
    for(m=0;m<x;m++){
        printf(" |%2d---%2d |%2d---%2d",an[m][9],an[m][10],an[m][11],an[m][12]);
        printf("%4d%3d%3d%3d|%2d|%2d|",
            an[m][5],an[m][6],an[m][7],an[m][8],cb[m][2],cb[m][6]);
        if(m==0){printf(" ");}
    }
    printf("\n");
    for(m=0;m<x;m++){
        printf("%3d-l-%2d |%4d-l-%2d |",an[m][5],an[m][6],an[m][7],an[m][8]);
        printf("%4d%3d%3d%3d|%2d|%2d|",
            an[m][9],an[m][10],an[m][11],an[m][12],cb[m][3],cb[m][7]);
        if(m==0){printf(" ");}
    }
    printf("\n");
    for(m=0;m<x;m++){
        printf("%5d---%2d%6d---%2d",an[m][13],an[m][14],an[m][15],an[m][16]);
        printf("%4d%3d%3d%3d|%2d|%2d|",
            an[m][13],an[m][14],an[m][15],an[m][16],cb[m][4],cb[m][8]);
        if(m==0){printf(" ");}
    }
    printf("\n");
}
}

```

```

//
/** Check Sums of every Conditions */
void checksums(short x){
    short m,n;
    /** for Basic Conditions */
    cb[x][1]=an[x][1]+an[x][2]+an[x][3]+an[x][4];
    cb[x][2]=an[x][5]+an[x][6]+an[x][7]+an[x][8];
    cb[x][3]=an[x][9]+an[x][10]+an[x][11]+an[x][12];
    cb[x][4]=an[x][13]+an[x][14]+an[x][15]+an[x][16];
    cb[x][5]=an[x][1]+an[x][5]+an[x][9]+an[x][13];
    cb[x][6]=an[x][2]+an[x][6]+an[x][10]+an[x][14];
    cb[x][7]=an[x][3]+an[x][7]+an[x][11]+an[x][15];
    cb[x][8]=an[x][4]+an[x][8]+an[x][12]+an[x][16];
    /** for Pandiagonal Conditions */
    cp[x][1]=an[x][1]+an[x][6]+an[x][11]+an[x][16];
    cp[x][2]=an[x][2]+an[x][7]+an[x][12]+an[x][13];
    cp[x][3]=an[x][3]+an[x][8]+an[x][9]+an[x][14];
    cp[x][4]=an[x][4]+an[x][5]+an[x][10]+an[x][15];
    cp[x][5]=an[x][1]+an[x][8]+an[x][11]+an[x][14];
    cp[x][6]=an[x][2]+an[x][5]+an[x][12]+an[x][15];
    cp[x][7]=an[x][3]+an[x][6]+an[x][9]+an[x][16];
    cp[x][8]=an[x][4]+an[x][7]+an[x][10]+an[x][13];
    /** for Conditions of Type; SC **
    css[x]=0;
    for(n=1;n<17;n++){m=17-n; if(an[x][n]+an[x][m]==17){css[x]++;}}
}
//
/** The 24 Transformations for Each Answer */
void trns24C(){
    short m,n;
    m=0; for(n=1;n<17;n++){s[n]=an[0][n];}; trnsf01(); for(n=1;n<17;n++){tn[m][n]=t[n];};
    m=1; for(n=1;n<17;n++){s[n]=an[0][n];}; trnsf02(); for(n=1;n<17;n++){tn[m][n]=t[n];};
    m=2; for(n=1;n<17;n++){s[n]=an[0][n];}; trnsf03(); for(n=1;n<17;n++){tn[m][n]=t[n];};
    m=3; for(n=1;n<17;n++){s[n]=an[0][n];}; trnsf04(); for(n=1;n<17;n++){tn[m][n]=t[n];};
    m=4; for(n=1;n<17;n++){s[n]=an[0][n];}; trnsf05(); for(n=1;n<17;n++){tn[m][n]=t[n];};
    m=5; for(n=1;n<17;n++){s[n]=an[0][n];}; trnsf06(); for(n=1;n<17;n++){tn[m][n]=t[n];};
    m=6; for(n=1;n<17;n++){s[n]=an[0][n];}; trnsf07(); for(n=1;n<17;n++){tn[m][n]=t[n];};
    m=7; for(n=1;n<17;n++){s[n]=an[0][n];}; trnsf08(); for(n=1;n<17;n++){tn[m][n]=t[n];};
    m=8; for(n=1;n<17;n++){s[n]=an[0][n];}; trnsf09(); for(n=1;n<17;n++){tn[m][n]=t[n];};
    m=9; for(n=1;n<17;n++){s[n]=an[0][n];}; trnsf10(); for(n=1;n<17;n++){tn[m][n]=t[n];};
    m=10; for(n=1;n<17;n++){s[n]=an[0][n];}; trnsf11(); for(n=1;n<17;n++){tn[m][n]=t[n];};
    m=11; for(n=1;n<17;n++){s[n]=an[0][n];}; trnsf12(); for(n=1;n<17;n++){tn[m][n]=t[n];};
    m=12; for(n=1;n<17;n++){s[n]=an[0][n];}; trnsf13(); for(n=1;n<17;n++){tn[m][n]=t[n];};
    m=13; for(n=1;n<17;n++){s[n]=an[0][n];}; trnsf14(); for(n=1;n<17;n++){tn[m][n]=t[n];};
    m=14; for(n=1;n<17;n++){s[n]=an[0][n];}; trnsf15(); for(n=1;n<17;n++){tn[m][n]=t[n];};
    m=15; for(n=1;n<17;n++){s[n]=an[0][n];}; trnsf16(); for(n=1;n<17;n++){tn[m][n]=t[n];};
    m=16; for(n=1;n<17;n++){s[n]=an[0][n];}; trnsf17(); for(n=1;n<17;n++){tn[m][n]=t[n];};
    m=17; for(n=1;n<17;n++){s[n]=an[0][n];}; trnsf18(); for(n=1;n<17;n++){tn[m][n]=t[n];};
    m=18; for(n=1;n<17;n++){s[n]=an[0][n];}; trnsf19(); for(n=1;n<17;n++){tn[m][n]=t[n];};
    m=19; for(n=1;n<17;n++){s[n]=an[0][n];}; trnsf20(); for(n=1;n<17;n++){tn[m][n]=t[n];};
    m=20; for(n=1;n<17;n++){s[n]=an[0][n];}; trnsf21(); for(n=1;n<17;n++){tn[m][n]=t[n];};
    m=21; for(n=1;n<17;n++){s[n]=an[0][n];}; trnsf22(); for(n=1;n<17;n++){tn[m][n]=t[n];};
    m=22; for(n=1;n<17;n++){s[n]=an[0][n];}; trnsf23(); for(n=1;n<17;n++){tn[m][n]=t[n];};
    m=23; for(n=1;n<17;n++){s[n]=an[0][n];}; trnsf24(); for(n=1;n<17;n++){tn[m][n]=t[n];};
}
//
/** Print the Concept of 24 Transformations */
void pr24trns(){
    short m,n;
    printf("\n");
    printf("*** Print the Conceptual Diagrams of 24 Transformations **\n");
    m=0; for(n=1;n<17;n++){s[n]=n;}; trnsf01(); for(n=1;n<17;n++){tn[m][n]=t[n];};
    m=1; for(n=1;n<17;n++){s[n]=n;}; trnsf02(); for(n=1;n<17;n++){tn[m][n]=t[n];};
    m=2; for(n=1;n<17;n++){s[n]=n;}; trnsf03(); for(n=1;n<17;n++){tn[m][n]=t[n];};
    m=3; for(n=1;n<17;n++){s[n]=n;}; trnsf04(); for(n=1;n<17;n++){tn[m][n]=t[n];};
    pr4t(0);
}

```

```

m=4; for(n=1;n<17;n++){s[n]=n;}; trnsf05(); for(n=1;n<17;n++){tn[m][n]=t[n];}
m=5; for(n=1;n<17;n++){s[n]=n;}; trnsf06(); for(n=1;n<17;n++){tn[m][n]=t[n];}
m=6; for(n=1;n<17;n++){s[n]=n;}; trnsf07(); for(n=1;n<17;n++){tn[m][n]=t[n];}
m=7; for(n=1;n<17;n++){s[n]=n;}; trnsf08(); for(n=1;n<17;n++){tn[m][n]=t[n];}
pr4t(4);
m=8; for(n=1;n<17;n++){s[n]=n;}; trnsf09(); for(n=1;n<17;n++){tn[m][n]=t[n];}
m=9; for(n=1;n<17;n++){s[n]=n;}; trnsf10(); for(n=1;n<17;n++){tn[m][n]=t[n];}
m=10; for(n=1;n<17;n++){s[n]=n;}; trnsf11(); for(n=1;n<17;n++){tn[m][n]=t[n];}
m=11; for(n=1;n<17;n++){s[n]=n;}; trnsf12(); for(n=1;n<17;n++){tn[m][n]=t[n];}
pr4t(8);
m=12; for(n=1;n<17;n++){s[n]=n;}; trnsf13(); for(n=1;n<17;n++){tn[m][n]=t[n];}
m=13; for(n=1;n<17;n++){s[n]=n;}; trnsf14(); for(n=1;n<17;n++){tn[m][n]=t[n];}
m=14; for(n=1;n<17;n++){s[n]=n;}; trnsf15(); for(n=1;n<17;n++){tn[m][n]=t[n];}
m=15; for(n=1;n<17;n++){s[n]=n;}; trnsf16(); for(n=1;n<17;n++){tn[m][n]=t[n];}
pr4t(12);
m=16; for(n=1;n<17;n++){s[n]=n;}; trnsf17(); for(n=1;n<17;n++){tn[m][n]=t[n];}
m=17; for(n=1;n<17;n++){s[n]=n;}; trnsf18(); for(n=1;n<17;n++){tn[m][n]=t[n];}
m=18; for(n=1;n<17;n++){s[n]=n;}; trnsf19(); for(n=1;n<17;n++){tn[m][n]=t[n];}
m=19; for(n=1;n<17;n++){s[n]=n;}; trnsf20(); for(n=1;n<17;n++){tn[m][n]=t[n];}
pr4t(16);
m=20; for(n=1;n<17;n++){s[n]=n;}; trnsf21(); for(n=1;n<17;n++){tn[m][n]=t[n];}
m=21; for(n=1;n<17;n++){s[n]=n;}; trnsf22(); for(n=1;n<17;n++){tn[m][n]=t[n];}
m=22; for(n=1;n<17;n++){s[n]=n;}; trnsf23(); for(n=1;n<17;n++){tn[m][n]=t[n];}
m=23; for(n=1;n<17;n++){s[n]=n;}; trnsf24(); for(n=1;n<17;n++){tn[m][n]=t[n];}
pr4t(20);
}
//
/* Print 2 Answers with 2 Transformations */
void pr4t(short x){
short m;
printf("%3d# /E1      /E2      T%02d/      T%02d/" ,x+1,x+1,x+2);
printf("%6d# /E1      /E2      T%02d/      T%02d/\n" ,x+3,x+3,x+4);
for(m=0;m<4;m=m+2){
printf("%2d---%2d%5d---%2d%6d%3d%3d%3d%4d%3d%3d%3d" ,
tn[x+m][1],tn[x+m][2],tn[x+m][3],tn[x+m][4],
tn[x+m][1],tn[x+m][2],tn[x+m][3],tn[x+m][4],
tn[x+m+1][1],tn[x+m+1][2],tn[x+m+1][3],tn[x+m+1][4]);
if(m==0){printf(" ");}
printf("\n");
for(m=0;m<4;m=m+2){
printf(" |%2d---%2d |%2d---%2d%4d%3d%3d%3d%4d%3d%3d%3d" ,
tn[x+m][9],tn[x+m][10],tn[x+m][11],tn[x+m][12],
tn[x+m][5],tn[x+m][6],tn[x+m][7],tn[x+m][8],
tn[x+m+1][5],tn[x+m+1][6],tn[x+m+1][7],tn[x+m+1][8]);
if(m==0){printf(" ");}
printf("\n");
for(m=0;m<4;m=m+2){
printf("%2d-l-%2d |%3d-l-%2d |%4d%3d%3d%3d%4d%3d%3d%3d" ,
tn[x+m][5],tn[x+m][6],tn[x+m][7],tn[x+m][8],
tn[x+m][9],tn[x+m][10],tn[x+m][11],tn[x+m][12],
tn[x+m+1][9],tn[x+m+1][10],tn[x+m+1][11],tn[x+m+1][12]);
if(m==0){printf(" ");}
printf("\n");
for(m=0;m<4;m=m+2){
printf("%4d---%2d%5d---%2d%4d%3d%3d%3d%4d%3d%3d%3d" ,
tn[x+m][13],tn[x+m][14],tn[x+m][15],tn[x+m][16],
tn[x+m][13],tn[x+m][14],tn[x+m][15],tn[x+m][16],
tn[x+m+1][13],tn[x+m+1][14],tn[x+m+1][15],tn[x+m+1][16]);
if(m==0){printf(" ");}
printf("\n\n");
}
//
/* The 24 Types of Transformations */
void trnsf01(){
t[ 1]=s[ 1]; t[ 2]=s[ 2]; t[ 3]=s[ 3]; t[ 4]=s[ 4];
t[ 5]=s[ 5]; t[ 6]=s[ 6]; t[ 7]=s[ 7]; t[ 8]=s[ 8];

```

```

t[ 9]=s[ 9]; t[10]=s[10]; t[11]=s[11]; t[12]=s[12];
t[13]=s[13]; t[14]=s[14]; t[15]=s[15]; t[16]=s[16];};
void trnsf02(){
t[ 1]=s[ 1]; t[ 2]=s[ 2]; t[ 3]=s[ 3]; t[ 4]=s[ 4];
t[ 5]=s[ 9]; t[ 6]=s[10]; t[ 7]=s[11]; t[ 8]=s[12];
t[ 9]=s[ 5]; t[10]=s[ 6]; t[11]=s[ 7]; t[12]=s[ 8];
t[13]=s[13]; t[14]=s[14]; t[15]=s[15]; t[16]=s[16];};
void trnsf03(){
t[ 1]=s[ 1]; t[ 2]=s[ 2]; t[ 3]=s[ 5]; t[ 4]=s[ 6];
t[ 5]=s[ 3]; t[ 6]=s[ 4]; t[ 7]=s[ 7]; t[ 8]=s[ 8];
t[ 9]=s[ 9]; t[10]=s[10]; t[11]=s[13]; t[12]=s[14];
t[13]=s[11]; t[14]=s[12]; t[15]=s[15]; t[16]=s[16];};
void trnsf04(){
t[ 1]=s[ 1]; t[ 2]=s[ 2]; t[ 3]=s[ 5]; t[ 4]=s[ 6];
t[ 5]=s[ 9]; t[ 6]=s[10]; t[ 7]=s[13]; t[ 8]=s[14];
t[ 9]=s[ 3]; t[10]=s[ 4]; t[11]=s[ 7]; t[12]=s[ 8];
t[13]=s[11]; t[14]=s[12]; t[15]=s[15]; t[16]=s[16];};
void trnsf05(){
t[ 1]=s[ 1]; t[ 2]=s[ 2]; t[ 3]=s[ 9]; t[ 4]=s[10];
t[ 5]=s[ 3]; t[ 6]=s[ 4]; t[ 7]=s[11]; t[ 8]=s[12];
t[ 9]=s[ 5]; t[10]=s[ 6]; t[11]=s[13]; t[12]=s[14];
t[13]=s[ 7]; t[14]=s[ 8]; t[15]=s[15]; t[16]=s[16];};
void trnsf06(){
t[ 1]=s[ 1]; t[ 2]=s[ 2]; t[ 3]=s[ 9]; t[ 4]=s[10];
t[ 5]=s[ 5]; t[ 6]=s[ 6]; t[ 7]=s[13]; t[ 8]=s[14];
t[ 9]=s[ 3]; t[10]=s[ 4]; t[11]=s[11]; t[12]=s[12];
t[13]=s[ 7]; t[14]=s[ 8]; t[15]=s[15]; t[16]=s[16];};
void trnsf07(){
t[ 1]=s[ 1]; t[ 2]=s[ 3]; t[ 3]=s[ 2]; t[ 4]=s[ 4];
t[ 5]=s[ 5]; t[ 6]=s[ 7]; t[ 7]=s[ 6]; t[ 8]=s[ 8];
t[ 9]=s[ 9]; t[10]=s[11]; t[11]=s[10]; t[12]=s[12];
t[13]=s[13]; t[14]=s[15]; t[15]=s[14]; t[16]=s[16];};
void trnsf08(){
t[ 1]=s[ 1]; t[ 2]=s[ 3]; t[ 3]=s[ 2]; t[ 4]=s[ 4];
t[ 5]=s[ 9]; t[ 6]=s[11]; t[ 7]=s[10]; t[ 8]=s[12];
t[ 9]=s[ 5]; t[10]=s[ 7]; t[11]=s[ 6]; t[12]=s[ 8];
t[13]=s[13]; t[14]=s[15]; t[15]=s[14]; t[16]=s[16];};
void trnsf09(){
t[ 1]=s[ 1]; t[ 2]=s[ 3]; t[ 3]=s[ 5]; t[ 4]=s[ 7];
t[ 5]=s[ 2]; t[ 6]=s[ 4]; t[ 7]=s[ 6]; t[ 8]=s[ 8];
t[ 9]=s[ 9]; t[10]=s[11]; t[11]=s[13]; t[12]=s[15];
t[13]=s[10]; t[14]=s[12]; t[15]=s[14]; t[16]=s[16];};
void trnsf10(){
t[ 1]=s[ 1]; t[ 2]=s[ 3]; t[ 3]=s[ 5]; t[ 4]=s[ 7];
t[ 5]=s[ 9]; t[ 6]=s[11]; t[ 7]=s[13]; t[ 8]=s[15];
t[ 9]=s[ 2]; t[10]=s[ 4]; t[11]=s[ 6]; t[12]=s[ 8];
t[13]=s[10]; t[14]=s[12]; t[15]=s[14]; t[16]=s[16];};
void trnsf11(){
t[ 1]=s[ 1]; t[ 2]=s[ 3]; t[ 3]=s[ 9]; t[ 4]=s[11];
t[ 5]=s[ 2]; t[ 6]=s[ 4]; t[ 7]=s[10]; t[ 8]=s[12];
t[ 9]=s[ 5]; t[10]=s[ 7]; t[11]=s[13]; t[12]=s[15];
t[13]=s[ 6]; t[14]=s[ 8]; t[15]=s[14]; t[16]=s[16];};
void trnsf12(){
t[ 1]=s[ 1]; t[ 2]=s[ 3]; t[ 3]=s[ 9]; t[ 4]=s[11];
t[ 5]=s[ 5]; t[ 6]=s[ 7]; t[ 7]=s[13]; t[ 8]=s[15];
t[ 9]=s[ 2]; t[10]=s[ 4]; t[11]=s[10]; t[12]=s[12];
t[13]=s[ 6]; t[14]=s[ 8]; t[15]=s[14]; t[16]=s[16];};
void trnsf13(){
t[ 1]=s[ 1]; t[ 2]=s[ 5]; t[ 3]=s[ 2]; t[ 4]=s[ 6];
t[ 5]=s[ 3]; t[ 6]=s[ 7]; t[ 7]=s[ 4]; t[ 8]=s[ 8];
t[ 9]=s[ 9]; t[10]=s[13]; t[11]=s[10]; t[12]=s[14];
t[13]=s[11]; t[14]=s[15]; t[15]=s[12]; t[16]=s[16];};
void trnsf14(){
t[ 1]=s[ 1]; t[ 2]=s[ 5]; t[ 3]=s[ 2]; t[ 4]=s[ 6];
t[ 5]=s[ 9]; t[ 6]=s[13]; t[ 7]=s[10]; t[ 8]=s[14];
t[ 9]=s[ 3]; t[10]=s[ 7]; t[11]=s[ 4]; t[12]=s[ 8];

```

```

t[13]=s[11]; t[14]=s[15]; t[15]=s[12]; t[16]=s[16];};
void trnsf15(){
t[ 1]=s[ 1]; t[ 2]=s[ 5]; t[ 3]=s[ 3]; t[ 4]=s[ 7];
t[ 5]=s[ 2]; t[ 6]=s[ 6]; t[ 7]=s[ 4]; t[ 8]=s[ 8];
t[ 9]=s[ 9]; t[10]=s[13]; t[11]=s[11]; t[12]=s[15];
t[13]=s[10]; t[14]=s[14]; t[15]=s[12]; t[16]=s[16];};
void trnsf16(){
t[ 1]=s[ 1]; t[ 2]=s[ 5]; t[ 3]=s[ 3]; t[ 4]=s[ 7];
t[ 5]=s[ 9]; t[ 6]=s[13]; t[ 7]=s[11]; t[ 8]=s[15];
t[ 9]=s[ 2]; t[10]=s[ 6]; t[11]=s[ 4]; t[12]=s[ 8];
t[13]=s[10]; t[14]=s[14]; t[15]=s[12]; t[16]=s[16];};
void trnsf17(){
t[ 1]=s[ 1]; t[ 2]=s[ 5]; t[ 3]=s[ 9]; t[ 4]=s[13];
t[ 5]=s[ 2]; t[ 6]=s[ 6]; t[ 7]=s[10]; t[ 8]=s[14];
t[ 9]=s[ 3]; t[10]=s[ 7]; t[11]=s[11]; t[12]=s[15];
t[13]=s[ 4]; t[14]=s[ 8]; t[15]=s[12]; t[16]=s[16];};
void trnsf18(){
t[ 1]=s[ 1]; t[ 2]=s[ 5]; t[ 3]=s[ 9]; t[ 4]=s[13];
t[ 5]=s[ 3]; t[ 6]=s[ 7]; t[ 7]=s[11]; t[ 8]=s[15];
t[ 9]=s[ 2]; t[10]=s[ 6]; t[11]=s[10]; t[12]=s[14];
t[13]=s[ 4]; t[14]=s[ 8]; t[15]=s[12]; t[16]=s[16];};
void trnsf19(){
t[ 1]=s[ 1]; t[ 2]=s[ 9]; t[ 3]=s[ 2]; t[ 4]=s[10];
t[ 5]=s[ 3]; t[ 6]=s[11]; t[ 7]=s[ 4]; t[ 8]=s[12];
t[ 9]=s[ 5]; t[10]=s[13]; t[11]=s[ 6]; t[12]=s[14];
t[13]=s[ 7]; t[14]=s[15]; t[15]=s[ 8]; t[16]=s[16];};
void trnsf20(){
t[ 1]=s[ 1]; t[ 2]=s[ 9]; t[ 3]=s[ 2]; t[ 4]=s[10];
t[ 5]=s[ 5]; t[ 6]=s[13]; t[ 7]=s[ 6]; t[ 8]=s[14];
t[ 9]=s[ 3]; t[10]=s[11]; t[11]=s[ 4]; t[12]=s[12];
t[13]=s[ 7]; t[14]=s[15]; t[15]=s[ 8]; t[16]=s[16];};
void trnsf21(){
t[ 1]=s[ 1]; t[ 2]=s[ 9]; t[ 3]=s[ 3]; t[ 4]=s[11];
t[ 5]=s[ 2]; t[ 6]=s[10]; t[ 7]=s[ 4]; t[ 8]=s[12];
t[ 9]=s[ 5]; t[10]=s[13]; t[11]=s[ 7]; t[12]=s[15];
t[13]=s[ 6]; t[14]=s[14]; t[15]=s[ 8]; t[16]=s[16];};
void trnsf22(){
t[ 1]=s[ 1]; t[ 2]=s[ 9]; t[ 3]=s[ 3]; t[ 4]=s[11];
t[ 5]=s[ 5]; t[ 6]=s[13]; t[ 7]=s[ 7]; t[ 8]=s[15];
t[ 9]=s[ 2]; t[10]=s[10]; t[11]=s[ 4]; t[12]=s[12];
t[13]=s[ 6]; t[14]=s[14]; t[15]=s[ 8]; t[16]=s[16];};
void trnsf23(){
t[ 1]=s[ 1]; t[ 2]=s[ 9]; t[ 3]=s[ 5]; t[ 4]=s[13];
t[ 5]=s[ 2]; t[ 6]=s[10]; t[ 7]=s[ 6]; t[ 8]=s[14];
t[ 9]=s[ 3]; t[10]=s[11]; t[11]=s[ 7]; t[12]=s[15];
t[13]=s[ 4]; t[14]=s[12]; t[15]=s[ 8]; t[16]=s[16];};
void trnsf24(){
t[ 1]=s[ 1]; t[ 2]=s[ 9]; t[ 3]=s[ 5]; t[ 4]=s[13];
t[ 5]=s[ 3]; t[ 6]=s[11]; t[ 7]=s[ 7]; t[ 8]=s[15];
t[ 9]=s[ 2]; t[10]=s[10]; t[11]=s[ 6]; t[12]=s[14];
t[13]=s[ 4]; t[14]=s[12]; t[15]=s[ 8]; t[16]=s[16];};
//

```

2-3. 計算して得られた結果は、次の通りである。原始解 384 個のリストを示す。  
各解に 24 個の展開形を付けると、リストが長大になるので、大幅に抜粋してある。

**\*\* 四次元二方陣を解いて補数対称型四方陣を得た結果（抜粋）： 24個の展開形つき \*\***  
[/SC:補数対称型; Rw:行和; Cl:列和; Pd,pd:汎対角線の和:]

1/E1	/E2	/SC	Rw	Cl	Pd/pd	
1--- 8	12---13	1	8	12	13	34 34\34/14
15---10	6--- 3	14	11	7	2	34 34\22/34
14- -11	7- - 2	15	10	6	3	34 34\34/54
4--- 5	9---16	4	5	9	16	34 34\46/34



T01: 1/	T02: 2/	T03: 3/	T04: 4/	T05: 5/	T06: 6/
1 8 12 13	1 8 12 13	1 8 14 11	1 8 14 11	1 8 15 10	1 8 15 10
14 11 7 2	15 10 6 3	12 13 7 2	15 10 4 5	12 13 6 3	14 11 4 5
15 10 6 3	14 11 7 2	15 10 4 5	12 13 7 2	14 11 4 5	12 13 6 3
4 5 9 16	4 5 9 16	6 3 9 16	6 3 9 16	7 2 9 16	7 2 9 16
T07: 7/	T08: 8/	T09: 9/	T10: 10/	T11: 11/	T12: 12/
1 12 8 13	1 12 8 13	1 12 14 7	1 12 14 7	1 12 15 6	1 12 15 6
14 7 11 2	15 6 10 3	8 13 11 2	15 6 4 9	8 13 10 3	14 7 4 9
15 6 10 3	14 7 11 2	15 6 4 9	8 13 11 2	14 7 4 9	8 13 10 3
4 9 5 16	4 9 5 16	10 3 5 16	10 3 5 16	11 2 5 16	11 2 5 16
T13: 13/	T14: 14/	T15: 15/	T16: 16/	T17: 17/	T18: 18/
1 14 8 11	1 14 8 11	1 14 12 7	1 14 12 7	1 14 15 4	1 14 15 4
12 7 13 2	15 4 10 5	8 11 13 2	15 4 6 9	8 11 10 5	12 7 6 9
15 4 10 5	12 7 13 2	15 4 6 9	8 11 13 2	12 7 6 9	8 11 10 5
6 9 3 16	6 9 3 16	10 5 3 16	10 5 3 16	13 2 3 16	13 2 3 16
T19: 19/	T20: 20/	T21: 21/	T22: 22/	T23: 23/	T24: 24/
1 15 8 10	1 15 8 10	1 15 12 6	1 15 12 6	1 15 14 4	1 15 14 4
12 6 13 3	14 4 11 5	8 10 13 3	14 4 7 9	8 10 11 5	12 6 7 9
14 4 11 5	12 6 13 3	14 4 7 9	8 10 13 3	12 6 7 9	8 10 11 5
7 9 2 16	7 9 2 16	11 5 2 16	11 5 2 16	13 3 2 16	13 3 2 16

2/E1	/E2	/SC Rw Cl\Pd/pd
1---8	12---13	1 8 12 13 34 34\34/16
14---11	7---2	15 10 6 3 34 34\20/34
15- -10	6- -3	14 11 7 2 34 34\34/52
4---5	9---16	4 5 9 16 34 34\48/34

T01: 2/	T02: 1/	T03: 5/	T04: 6/	T05: 3/	T06: 4/
1 8 12 13	1 8 12 13	1 8 15 10	1 8 15 10	1 8 14 11	1 8 14 11
15 10 6 3	14 11 7 2	12 13 6 3	14 11 4 5	12 13 7 2	15 10 4 5
14 11 7 2	15 10 6 3	14 11 4 5	12 13 6 3	15 10 4 5	12 13 7 2
4 5 9 16	4 5 9 16	7 2 9 16	7 2 9 16	6 3 9 16	6 3 9 16
T07: 8/	T08: 7/	T09: 11/	T10: 12/	T11: 9/	T12: 10/
1 12 8 13	1 12 8 13	1 12 15 6	1 12 15 6	1 12 14 7	1 12 14 7
15 6 10 3	14 7 11 2	8 13 10 3	14 7 4 9	8 13 11 2	15 6 4 9
14 7 11 2	15 6 10 3	14 7 4 9	8 13 10 3	15 6 4 9	8 13 11 2
4 9 5 16	4 9 5 16	11 2 5 16	11 2 5 16	10 3 5 16	10 3 5 16
T13: 19/	T14: 20/	T15: 21/	T16: 22/	T17: 23/	T18: 24/
1 15 8 10	1 15 8 10	1 15 12 6	1 15 12 6	1 15 14 4	1 15 14 4
12 6 13 3	14 4 11 5	8 10 13 3	14 4 7 9	8 10 11 5	12 6 7 9
14 4 11 5	12 6 13 3	14 4 7 9	8 10 13 3	12 6 7 9	8 10 11 5
7 9 2 16	7 9 2 16	11 5 2 16	11 5 2 16	13 3 2 16	13 3 2 16
T19: 13/	T20: 14/	T21: 15/	T22: 16/	T23: 17/	T24: 18/
1 14 8 11	1 14 8 11	1 14 12 7	1 14 12 7	1 14 15 4	1 14 15 4
12 7 13 2	15 4 10 5	8 11 13 2	15 4 6 9	8 11 10 5	12 7 6 9
15 4 10 5	12 7 13 2	15 4 6 9	8 11 13 2	12 7 6 9	8 11 10 5
6 9 3 16	6 9 3 16	10 5 3 16	10 5 3 16	13 2 3 16	13 2 3 16

3/E1	/E2	/SC Rw Cl\Pd/pd
1---8	14---11	1 8 14 11 34 34\34/10
15---10	4---5	12 13 7 2 34 34\26/34
12- -13	7- -2	15 10 4 5 34 34\34/58
6---3	9---16	6 3 9 16 34 34\42/34

T01: 3/	T02: 4/	T03: 1/	T04: 2/	T05: 6/	T06: 5/
1 8 14 11	1 8 14 11	1 8 12 13	1 8 12 13	1 8 15 10	1 8 15 10
12 13 7 2	15 10 4 5	14 11 7 2	15 10 6 3	14 11 4 5	12 13 6 3
15 10 4 5	12 13 7 2	15 10 6 3	14 11 7 2	12 13 6 3	14 11 4 5
6 3 9 16	6 3 9 16	4 5 9 16	4 5 9 16	7 2 9 16	7 2 9 16
T07: 13/	T08: 14/	T09: 15/	T10: 16/	T11: 17/	T12: 18/
1 14 8 11	1 14 8 11	1 14 12 7	1 14 12 7	1 14 15 4	1 14 15 4
12 7 13 2	15 4 10 5	8 11 13 2	15 4 6 9	8 11 10 5	12 7 6 9
15 4 10 5	12 7 13 2	15 4 6 9	8 11 13 2	12 7 6 9	8 11 10 5
6 9 3 16	6 9 3 16	10 5 3 16	10 5 3 16	13 2 3 16	13 2 3 16

T13: 7/	T14: 8/	T15: 9/	T16: 10/	T17: 11/	T18: 12/
1 12 8 13	1 12 8 13	1 12 14 7	1 12 14 7	1 12 15 6	1 12 15 6
14 7 11 2	15 6 10 3	8 13 11 2	15 6 4 9	8 13 10 3	14 7 4 9
15 6 10 3	14 7 11 2	15 6 4 9	8 13 11 2	14 7 4 9	8 13 10 3
4 9 5 16	4 9 5 16	10 3 5 16	10 3 5 16	11 2 5 16	11 2 5 16
T19: 20/	T20: 19/	T21: 23/	T22: 24/	T23: 21/	T24: 22/
1 15 8 10	1 15 8 10	1 15 14 4	1 15 14 4	1 15 12 6	1 15 12 6
14 4 11 5	12 6 13 3	8 10 11 5	12 6 7 9	8 10 13 3	14 4 7 9
12 6 13 3	14 4 11 5	12 6 7 9	8 10 11 5	14 4 7 9	8 10 13 3
7 9 2 16	7 9 2 16	13 3 2 16	13 3 2 16	11 5 2 16	11 5 2 16

4/E1	/E2	/SC Rw Cl\Pd/pd
1---8	14---11	1 8 14 11 34 34\34/16
12---13	7---2	15 10 4 5 34 34\20/34
15- -10	4- -5	12 13 7 2 34 34\34/52
6---3	9---16	6 3 9 16 34 34\48/34

T01: 4/	T02: 3/	T03: 6/	T04: 5/	T05: 1/	T06: 2/
1 8 14 11	1 8 14 11	1 8 15 10	1 8 15 10	1 8 12 13	1 8 12 13
15 10 4 5	12 13 7 2	14 11 4 5	12 13 6 3	14 11 7 2	15 10 6 3
12 13 7 2	15 10 4 5	12 13 6 3	14 11 4 5	15 10 6 3	14 11 7 2
6 3 9 16	6 3 9 16	7 2 9 16	7 2 9 16	4 5 9 16	4 5 9 16

T07: 14/	T08: 13/	T09: 17/	T10: 18/	T11: 15/	T12: 16/
1 14 8 11	1 14 8 11	1 14 15 4	1 14 15 4	1 14 12 7	1 14 12 7
15 4 10 5	12 7 13 2	8 11 10 5	12 7 6 9	8 11 13 2	15 4 6 9
12 7 13 2	15 4 10 5	12 7 6 9	8 11 10 5	15 4 6 9	8 11 13 2
6 9 3 16	6 9 3 16	13 2 3 16	13 2 3 16	10 5 3 16	10 5 3 16

T13: 20/	T14: 19/	T15: 23/	T16: 24/	T17: 21/	T18: 22/
1 15 8 10	1 15 8 10	1 15 14 4	1 15 14 4	1 15 12 6	1 15 12 6
14 4 11 5	12 6 13 3	8 10 11 5	12 6 7 9	8 10 13 3	14 4 7 9
12 6 13 3	14 4 11 5	12 6 7 9	8 10 11 5	14 4 7 9	8 10 13 3
7 9 2 16	7 9 2 16	13 3 2 16	13 3 2 16	11 5 2 16	11 5 2 16

T19: 7/	T20: 8/	T21: 9/	T22: 10/	T23: 11/	T24: 12/
1 12 8 13	1 12 8 13	1 12 14 7	1 12 14 7	1 12 15 6	1 12 15 6
14 7 11 2	15 6 10 3	8 13 11 2	15 6 4 9	8 13 10 3	14 7 4 9
15 6 10 3	14 7 11 2	15 6 4 9	8 13 11 2	14 7 4 9	8 13 10 3
4 9 5 16	4 9 5 16	10 3 5 16	10 3 5 16	11 2 5 16	11 2 5 16

5/E1	/E2	/SC Rw Cl\Pd/pd
1---8	15---10	1 8 15 10 34 34\34/10
14---11	4---5	12 13 6 3 34 34\26/34
12- -13	6- -3	14 11 4 5 34 34\34/58
7---2	9---16	7 2 9 16 34 34\42/34

T01: 5/	T02: 6/	T03: 2/	T04: 1/	T05: 4/	T06: 3/
1 8 15 10	1 8 15 10	1 8 12 13	1 8 12 13	1 8 14 11	1 8 14 11
12 13 6 3	14 11 4 5	15 10 6 3	14 11 7 2	15 10 4 5	12 13 7 2
14 11 4 5	12 13 6 3	14 11 7 2	15 10 6 3	12 13 7 2	15 10 4 5
7 2 9 16	7 2 9 16	4 5 9 16	4 5 9 16	6 3 9 16	6 3 9 16

T07: 19/	T08: 20/	T09: 21/	T10: 22/	T11: 23/	T12: 24/
1 15 8 10	1 15 8 10	1 15 12 6	1 15 12 6	1 15 14 4	1 15 14 4
12 6 13 3	14 4 11 5	8 10 13 3	14 4 7 9	8 10 11 5	12 6 7 9
14 4 11 5	12 6 13 3	14 4 7 9	8 10 13 3	12 6 7 9	8 10 11 5
7 9 2 16	7 9 2 16	11 5 2 16	11 5 2 16	13 3 2 16	13 3 2 16

T13: 8/	T14: 7/	T15: 11/	T16: 12/	T17: 9/	T18: 10/
1 12 8 13	1 12 8 13	1 12 15 6	1 12 15 6	1 12 14 7	1 12 14 7
15 6 10 3	14 7 11 2	8 13 10 3	14 7 4 9	8 13 11 2	15 6 4 9
14 7 11 2	15 6 10 3	14 7 4 9	8 13 10 3	15 6 4 9	8 13 11 2
4 9 5 16	4 9 5 16	11 2 5 16	11 2 5 16	10 3 5 16	10 3 5 16

T19: 14/	T20: 13/	T21: 17/	T22: 18/	T23: 15/	T24: 16/
1 14 8 11	1 14 8 11	1 14 15 4	1 14 15 4	1 14 12 7	1 14 12 7
15 4 10 5	12 7 13 2	8 11 10 5	12 7 6 9	8 11 13 2	15 4 6 9
12 7 13 2	15 4 10 5	12 7 6 9	8 11 10 5	15 4 6 9	8 11 13 2
6 9 3 16	6 9 3 16	13 2 3 16	13 2 3 16	10 5 3 16	10 5 3 16

6/E1	/E2					/SC Rw CI\Pd/pd
1---8	15---10	1	8	15	10	34 34\34/14
12---13	6---3	14	11	4	5	34 34\22/34
14- -11	4- -5	12	13	6	3	34 34\34/54
7---2	9---16	7	2	9	16	34 34\46/34
T01: 6/	T02: 5/	T03: 4/	T04: 3/	T05: 2/	T06: 1/	
1 8 15 10	1 8 15 10	1 8 14 11	1 8 14 11	1 8 12 13	1 8 12 13	
14 11 4 5	12 13 6 3	15 10 4 5	12 13 7 2	15 10 6 3	14 11 7 2	
12 13 6 3	14 11 4 5	12 13 7 2	15 10 4 5	14 11 7 2	15 10 6 3	
7 2 9 16	7 2 9 16	6 3 9 16	6 3 9 16	4 5 9 16	4 5 9 16	
T07: 20/	T08: 19/	T09: 23/	T10: 24/	T11: 21/	T12: 22/	
1 15 8 10	1 15 8 10	1 15 14 4	1 15 14 4	1 15 12 6	1 15 12 6	
14 4 11 5	12 6 13 3	8 10 11 5	12 6 7 9	8 10 13 3	14 4 7 9	
12 6 13 3	14 4 11 5	12 6 7 9	8 10 11 5	14 4 7 9	8 10 13 3	
7 9 2 16	7 9 2 16	13 3 2 16	13 3 2 16	11 5 2 16	11 5 2 16	
T13: 14/	T14: 13/	T15: 17/	T16: 18/	T17: 15/	T18: 16/	
1 14 8 11	1 14 8 11	1 14 15 4	1 14 15 4	1 14 12 7	1 14 12 7	
15 4 10 5	12 7 13 2	8 11 10 5	12 7 6 9	8 11 13 2	15 4 6 9	
12 7 13 2	15 4 10 5	12 7 6 9	8 11 10 5	15 4 6 9	8 11 13 2	
6 9 3 16	6 9 3 16	13 2 3 16	13 2 3 16	10 5 3 16	10 5 3 16	
T19: 8/	T20: 7/	T21: 11/	T22: 12/	T23: 9/	T24: 10/	
1 12 8 13	1 12 8 13	1 12 15 6	1 12 15 6	1 12 14 7	1 12 14 7	
15 6 10 3	14 7 11 2	8 13 10 3	14 7 4 9	8 13 11 2	15 6 4 9	
14 7 11 2	15 6 10 3	14 7 4 9	8 13 10 3	15 6 4 9	8 13 11 2	
4 9 5 16	4 9 5 16	11 2 5 16	11 2 5 16	10 3 5 16	10 3 5 16	

7/E1	/E2					/SC Rw CI\Pd/pd
1---12	8---13	1	12	8	13	34 34\34/22
15---6	10---3	14	7	11	2	34 34\30/34
14- -7	11- -2	15	6	10	3	34 34\34/46
4---9	5---16	4	9	5	16	34 34\38/34
T01: 7/	T02: 8/	T03: 9/	T04: 10/	T05: 11/	T06: 12/	
1 12 8 13	1 12 8 13	1 12 14 7	1 12 14 7	1 12 15 6	1 12 15 6	
14 7 11 2	15 6 10 3	8 13 11 2	15 6 4 9	8 13 10 3	14 7 4 9	
15 6 10 3	14 7 11 2	15 6 4 9	8 13 11 2	14 7 4 9	8 13 10 3	
4 9 5 16	4 9 5 16	10 3 5 16	10 3 5 16	11 2 5 16	11 2 5 16	
T07: 1/	T08: 2/	T09: 3/	T10: 4/	T11: 5/	T12: 6/	
1 8 12 13	1 8 12 13	1 8 14 11	1 8 14 11	1 8 15 10	1 8 15 10	
14 11 7 2	15 10 6 3	12 13 7 2	15 10 4 5	12 13 6 3	14 11 4 5	
15 10 6 3	14 11 7 2	15 10 4 5	12 13 7 2	14 11 4 5	12 13 6 3	
4 5 9 16	4 5 9 16	6 3 9 16	6 3 9 16	7 2 9 16	7 2 9 16	
T13: 15/	T14: 16/	T15: 13/	T16: 14/	T17: 18/	T18: 17/	
1 14 12 7	1 14 12 7	1 14 8 11	1 14 8 11	1 14 15 4	1 14 15 4	
8 11 13 2	15 4 6 9	12 7 13 2	15 4 10 5	12 7 6 9	8 11 10 5	
15 4 6 9	8 11 13 2	15 4 10 5	12 7 13 2	8 11 10 5	12 7 6 9	
10 5 3 16	10 5 3 16	6 9 3 16	6 9 3 16	13 2 3 16	13 2 3 16	
T19: 21/	T20: 22/	T21: 19/	T22: 20/	T23: 24/	T24: 23/	
1 15 12 6	1 15 12 6	1 15 8 10	1 15 8 10	1 15 14 4	1 15 14 4	
8 10 13 3	14 4 7 9	12 6 13 3	14 4 11 5	12 6 7 9	8 10 11 5	
14 4 7 9	8 10 13 3	14 4 11 5	12 6 13 3	8 10 11 5	12 6 7 9	
11 5 2 16	11 5 2 16	7 9 2 16	7 9 2 16	13 3 2 16	13 3 2 16	

8/E1	/E2					/SC Rw CI\Pd/pd
1---12	8---13	1	12	8	13	34 34\34/24
14---7	11---2	15	6	10	3	34 34\28/34
15- -6	10- -3	14	7	11	2	34 34\34/44
4---9	5---16	4	9	5	16	34 34\40/34
T01: 8/	T02: 7/	T03: 11/	T04: 12/	T05: 9/	T06: 10/	
1 12 8 13	1 12 8 13	1 12 15 6	1 12 15 6	1 12 14 7	1 12 14 7	
15 6 10 3	14 7 11 2	8 13 10 3	14 7 4 9	8 13 11 2	15 6 4 9	
14 7 11 2	15 6 10 3	14 7 4 9	8 13 10 3	15 6 4 9	8 13 11 2	
4 9 5 16	4 9 5 16	11 2 5 16	11 2 5 16	10 3 5 16	10 3 5 16	

T07: 2/ 1 8 12 13 15 10 6 3 14 11 7 2 4 5 9 16	T08: 1/ 1 8 12 13 14 11 7 2 15 10 6 3 4 5 9 16	T09: 5/ 1 8 15 10 12 13 6 3 14 11 4 5 7 2 9 16	T10: 6/ 1 8 15 10 14 11 4 5 12 13 6 3 7 2 9 16	T11: 3/ 1 8 14 11 12 13 7 2 15 10 4 5 6 3 9 16	T12: 4/ 1 8 14 11 15 10 4 5 12 13 7 2 6 3 9 16
T13: 21/ 1 15 12 6 8 10 13 3 14 4 7 9 11 5 2 16	T14: 22/ 1 15 12 6 14 4 7 9 8 10 13 3 11 5 2 16	T15: 19/ 1 15 8 10 12 6 13 3 14 4 11 5 7 9 2 16	T16: 20/ 1 15 8 10 14 4 11 5 12 6 13 3 7 9 2 16	T17: 24/ 1 15 14 4 12 6 7 9 8 10 11 5 13 3 2 16	T18: 23/ 1 15 14 4 8 10 11 5 12 6 7 9 13 3 2 16
T19: 15/ 1 14 12 7 8 11 13 2 15 4 6 9 10 5 3 16	T20: 16/ 1 14 12 7 15 4 6 9 8 11 13 2 10 5 3 16	T21: 13/ 1 14 8 11 12 7 13 2 15 4 10 5 6 9 3 16	T22: 14/ 1 14 8 11 15 4 10 5 12 7 13 2 6 9 3 16	T23: 18/ 1 14 15 4 12 7 6 9 8 11 10 5 13 2 3 16	T24: 17/ 1 14 15 4 8 11 10 5 12 7 6 9 13 2 3 16

9/E1 1---12  15---6 8- -13  10---3	/E2 14---7  4---9 11- -2  5---16	/SC Rw C1\Pd/pd  1 12 14 7 34 34\34/10  8 13 11 2 34 34\42/34  15 6 4 9 34 34\34/58  10 3 5 16 34 34\26/34
------------------------------------------------	----------------------------------------------	------------------------------------------------------------------------------------------------------------------------

T01: 9/ 1 12 14 7 8 13 11 2 15 6 4 9 10 3 5 16	T02: 10/ 1 12 14 7 15 6 4 9 8 13 11 2 10 3 5 16	T03: 7/ 1 12 8 13 14 7 11 2 15 6 10 3 4 9 5 16	T04: 8/ 1 12 8 13 15 6 10 3 14 7 11 2 4 9 5 16	T05: 12/ 1 12 15 6 14 7 4 9 8 13 10 3 11 2 5 16	T06: 11/ 1 12 15 6 8 13 10 3 14 7 4 9 11 2 5 16
T07: 15/ 1 14 12 7 8 11 13 2 15 4 6 9 10 5 3 16	T08: 16/ 1 14 12 7 15 4 6 9 8 11 13 2 10 5 3 16	T09: 13/ 1 14 8 11 12 7 13 2 15 4 10 5 6 9 3 16	T10: 14/ 1 14 8 11 15 4 10 5 12 7 13 2 6 9 3 16	T11: 18/ 1 14 15 4 12 7 6 9 8 11 10 5 13 2 3 16	T12: 17/ 1 14 15 4 8 11 10 5 12 7 6 9 13 2 3 16
T13: 1/ 1 8 12 13 14 11 7 2 15 10 6 3 4 5 9 16	T14: 2/ 1 8 12 13 15 10 6 3 14 11 7 2 4 5 9 16	T15: 3/ 1 8 14 11 12 13 7 2 15 10 4 5 6 3 9 16	T16: 4/ 1 8 14 11 15 10 4 5 12 13 7 2 6 3 9 16	T17: 5/ 1 8 15 10 12 13 6 3 14 11 4 5 7 2 9 16	T18: 6/ 1 8 15 10 14 11 4 5 12 13 6 3 7 2 9 16
T19: 22/ 1 15 12 6 14 4 7 9 8 10 13 3 11 5 2 16	T20: 21/ 1 15 12 6 8 10 13 3 14 4 7 9 11 5 2 16	T21: 24/ 1 15 14 4 12 6 7 9 8 10 11 5 13 3 2 16	T22: 23/ 1 15 14 4 8 10 11 5 12 6 7 9 13 3 2 16	T23: 19/ 1 15 8 10 12 6 13 3 14 4 11 5 7 9 2 16	T24: 20/ 1 15 8 10 14 4 11 5 12 6 13 3 7 9 2 16

10/E1 1---12  8---13 15- -6  10---3	/E2 14---7  11---2 4- -9  5---16	/SC Rw C1\Pd/pd  1 12 14 7 34 34\34/24  15 6 4 9 34 34\28/34  8 13 11 2 34 34\34/44  10 3 5 16 34 34\40/34			
T01: 10/ 1 12 14 7 15 6 4 9 8 13 11 2 10 3 5 16	T02: 9/ 1 12 14 7 8 13 11 2 15 6 4 9 10 3 5 16	T03: 12/ 1 12 15 6 14 7 4 9 8 13 10 3 11 2 5 16	T04: 11/ 1 12 15 6 8 13 10 3 14 7 4 9 11 2 5 16	T05: 7/ 1 12 8 13 14 7 11 2 15 6 10 3 4 9 5 16	T06: 8/ 1 12 8 13 15 6 10 3 14 7 11 2 4 9 5 16
T07: 16/ 1 14 12 7 15 4 6 9 8 11 13 2 10 5 3 16	T08: 15/ 1 14 12 7 8 11 13 2 15 4 6 9 10 5 3 16	T09: 18/ 1 14 15 4 12 7 6 9 8 11 10 5 13 2 3 16	T10: 17/ 1 14 15 4 8 11 10 5 12 7 6 9 13 2 3 16	T11: 13/ 1 14 8 11 12 7 13 2 15 4 10 5 6 9 3 16	T12: 14/ 1 14 8 11 15 4 10 5 12 7 13 2 6 9 3 16
T13: 22/ 1 15 12 6 14 4 7 9 8 10 13 3 11 5 2 16	T14: 21/ 1 15 12 6 8 10 13 3 14 4 7 9 11 5 2 16	T15: 24/ 1 15 14 4 12 6 7 9 8 10 11 5 13 3 2 16	T16: 23/ 1 15 14 4 8 10 11 5 12 6 7 9 13 3 2 16	T17: 19/ 1 15 8 10 12 6 13 3 14 4 11 5 7 9 2 16	T18: 20/ 1 15 8 10 14 4 11 5 12 6 13 3 7 9 2 16

T19: 1/ T20: 2/ T21: 3/ T22: 4/ T23: 5/ T24: 6/  
 1 8 12 13 1 8 12 13 1 8 14 11 1 8 14 11 1 8 15 10 1 8 15 10  
 14 11 7 2 15 10 6 3 12 13 7 2 15 10 4 5 12 13 6 3 14 11 4 5  
 15 10 6 3 14 11 7 2 15 10 4 5 12 13 7 2 14 11 4 5 12 13 6 3  
 4 5 9 16 4 5 9 16 6 3 9 16 6 3 9 16 7 2 9 16 7 2 9 16

11/E1 /E2 /SC|Rw|Cl\Pd/pd|  
 1---12 15---6 1 12 15 6|34|34\34/10|  
 |14---7 | 4---9 8 13 10 3|34|34\42/34|  
 8-|-13 | 10-|-3 | 14 7 4 9|34|34\34/58|  
 11---2 5---16 11 2 5 16|34|34\26/34|

T01: 11/ T02: 12/ T03: 8/ T04: 7/ T05: 10/ T06: 9/  
 1 12 15 6 1 12 15 6 1 12 8 13 1 12 8 13 1 12 14 7 1 12 14 7  
 8 13 10 3 14 7 4 9 15 6 10 3 14 7 11 2 15 6 4 9 8 13 11 2  
 14 7 4 9 8 13 10 3 14 7 11 2 15 6 10 3 8 13 11 2 15 6 4 9  
 11 2 5 16 11 2 5 16 4 9 5 16 4 9 5 16 10 3 5 16 10 3 5 16

T07: 21/ T08: 22/ T09: 19/ T10: 20/ T11: 24/ T12: 23/  
 1 15 12 6 1 15 12 6 1 15 8 10 1 15 8 10 1 15 14 4 1 15 14 4  
 8 10 13 3 14 4 7 9 12 6 13 3 14 4 11 5 12 6 7 9 8 10 11 5  
 14 4 7 9 8 10 13 3 14 4 11 5 12 6 13 3 8 10 11 5 12 6 7 9  
 11 5 2 16 11 5 2 16 7 9 2 16 7 9 2 16 13 3 2 16 13 3 2 16

T13: 2/ T14: 1/ T15: 5/ T16: 6/ T17: 3/ T18: 4/  
 1 8 12 13 1 8 12 13 1 8 15 10 1 8 15 10 1 8 14 11 1 8 14 11  
 15 10 6 3 14 11 7 2 12 13 6 3 14 11 4 5 12 13 7 2 15 10 4 5  
 14 11 7 2 15 10 6 3 14 11 4 5 12 13 6 3 15 10 4 5 12 13 7 2  
 4 5 9 16 4 5 9 16 7 2 9 16 7 2 9 16 6 3 9 16 6 3 9 16

T19: 16/ T20: 15/ T21: 18/ T22: 17/ T23: 13/ T24: 14/  
 1 14 12 7 1 14 12 7 1 14 15 4 1 14 15 4 1 14 8 11 1 14 8 11  
 15 4 6 9 8 11 13 2 12 7 6 9 8 11 10 5 12 7 13 2 15 4 10 5  
 8 11 13 2 15 4 6 9 8 11 10 5 12 7 6 9 15 4 10 5 12 7 13 2  
 10 5 3 16 10 5 3 16 13 2 3 16 13 2 3 16 6 9 3 16 6 9 3 16

12/E1 /E2 /SC|Rw|Cl\Pd/pd|  
 1---12 15---6 1 12 15 6|34|34\34/22|  
 | 8---13 |10---3 14 7 4 9|34|34\30/34|  
 14-|-7 | 4-|-9 | 8 13 10 3|34|34\34/46|  
 11---2 5---16 11 2 5 16|34|34\38/34|

T01: 12/ T02: 11/ T03: 10/ T04: 9/ T05: 8/ T06: 7/  
 1 12 15 6 1 12 15 6 1 12 14 7 1 12 14 7 1 12 8 13 1 12 8 13  
 14 7 4 9 8 13 10 3 15 6 4 9 8 13 11 2 15 6 10 3 14 7 11 2  
 8 13 10 3 14 7 4 9 8 13 11 2 15 6 4 9 14 7 11 2 15 6 10 3  
 11 2 5 16 11 2 5 16 10 3 5 16 10 3 5 16 4 9 5 16 4 9 5 16

T07: 22/ T08: 21/ T09: 24/ T10: 23/ T11: 19/ T12: 20/  
 1 15 12 6 1 15 12 6 1 15 14 4 1 15 14 4 1 15 8 10 1 15 8 10  
 14 4 7 9 8 10 13 3 12 6 7 9 8 10 11 5 12 6 13 3 14 4 11 5  
 8 10 13 3 14 4 7 9 8 10 11 5 12 6 7 9 14 4 11 5 12 6 13 3  
 11 5 2 16 11 5 2 16 13 3 2 16 13 3 2 16 7 9 2 16 7 9 2 16

T13: 16/ T14: 15/ T15: 18/ T16: 17/ T17: 13/ T18: 14/  
 1 14 12 7 1 14 12 7 1 14 15 4 1 14 15 4 1 14 8 11 1 14 8 11  
 15 4 6 9 8 11 13 2 12 7 6 9 8 11 10 5 12 7 13 2 15 4 10 5  
 8 11 13 2 15 4 6 9 8 11 10 5 12 7 6 9 15 4 10 5 12 7 13 2  
 10 5 3 16 10 5 3 16 13 2 3 16 13 2 3 16 6 9 3 16 6 9 3 16

T19: 2/ T20: 1/ T21: 5/ T22: 6/ T23: 3/ T24: 4/  
 1 8 12 13 1 8 12 13 1 8 15 10 1 8 15 10 1 8 14 11 1 8 14 11  
 15 10 6 3 14 11 7 2 12 13 6 3 14 11 4 5 12 13 7 2 15 10 4 5  
 14 11 7 2 15 10 6 3 14 11 4 5 12 13 6 3 15 10 4 5 12 13 7 2  
 4 5 9 16 4 5 9 16 7 2 9 16 7 2 9 16 6 3 9 16 6 3 9 16

13/E1 /E2 /SC\34/34| 14/E1 /E2 /SC\34/34|  
 1---14 8---11 1 14 8 11|34|34| 1---14 8---11 1 14 8 11|34|34|  
 |15---4 |10---5 12 7 13 2|34|34| |12---7 |13---2 15 4 10 5|34|34|  
 12-|-7 | 13-|-2 | 15 4 10 5|34|34| 15-|-4 | 10-|-5 | 12 7 13 2|34|34|  
 6---9 3---16 6 9 3 16|34|34| 6---9 3---16 6 9 3 16|34|34|

15/E1	/E2	/SC\34/34	16/E1	/E2	/SC\34/34
1---14	12---7	1 14 12 7 34 34	1---14	12---7	1 14 12 7 34 34
15---4	6---9	8 11 13 2 34 34	8---11	13---2	15 4 6 9 34 34
8- -11	13- -2	15 4 6 9 34 34	15- -4	6- -9	8 11 13 2 34 34
10---5	3---16	10 5 3 16 34 34	10---5	3---16	10 5 3 16 34 34

17/E1	/E2	/SC\34/34	18/E1	/E2	/SC\34/34
1---14	15---4	1 14 15 4 34 34	1---14	15---4	1 14 15 4 34 34
12---7	6---9	8 11 10 5 34 34	8---11	10---5	12 7 6 9 34 34
8- -11	10- -5	12 7 6 9 34 34	12- -7	6- -9	8 11 10 5 34 34
13---2	3---16	13 2 3 16 34 34	13---2	3---16	13 2 3 16 34 34

19/E1	/E2	/SC\34/34	20/E1	/E2	/SC\34/34
1---15	8---10	1 15 8 10 34 34	1---15	8---10	1 15 8 10 34 34
14---4	11---5	12 6 13 3 34 34	12---6	13---3	14 4 11 5 34 34
12- -6	13- -3	14 4 11 5 34 34	14- -4	11- -5	12 6 13 3 34 34
7---9	2---16	7 9 2 16 34 34	7---9	2---16	7 9 2 16 34 34

21/E1	/E2	/SC\34/34	22/E1	/E2	/SC\34/34
1---15	12---6	1 15 12 6 34 34	1---15	12---6	1 15 12 6 34 34
14---4	7---9	8 10 13 3 34 34	8---10	13---3	14 4 7 9 34 34
8- -10	13- -3	14 4 7 9 34 34	14- -4	7- -9	8 10 13 3 34 34
11---5	2---16	11 5 2 16 34 34	11---5	2---16	11 5 2 16 34 34

23/E1	/E2	/SC\34/34	24/E1	/E2	/SC\34/34
1---15	14---4	1 15 14 4 34 34	1---15	14---4	1 15 14 4 34 34
12---6	7---9	8 10 11 5 34 34	8---10	11---5	12 6 7 9 34 34
8- -10	11- -5	12 6 7 9 34 34	12- -6	7- -9	8 10 11 5 34 34
13---3	2---16	13 3 2 16 34 34	13---3	2---16	13 3 2 16 34 34

25/E1	/E2	/SC\34/34	49/E1	/E2	/SC\34/34
2---7	11---14	2 7 11 14 34 34	3---6	10---15	3 6 10 15 34 34
16---9	5---4	13 12 8 1 34 34	16---9	5---4	13 12 8 1 34 34
13- -12	8- -1	16 9 5 4 34 34	13- -12	8- -1	16 9 5 4 34 34
3---6	10---15	3 6 10 15 34 34	2---7	11---14	2 7 11 14 34 34

73/E1	/E2	/SC\34/34	97/E1	/E2	/SC\34/34
4---5	9---16	4 5 9 16 34 34	5---4	10---15	5 4 10 15 34 34
15---10	6---3	14 11 7 2 34 34	16---9	3---6	11 14 8 1 34 34
14- -11	7- -2	15 10 6 3 34 34	11- -14	8- -1	16 9 3 6 34 34
1---8	12---13	1 8 12 13 34 34	2---7	13---12	2 7 13 12 34 34

121/E1	/E2	/SC\34/34	145/E1	/E2	/SC\34/34
6---3	9---16	6 3 9 16 34 34	7---2	9---16	7 2 9 16 34 34
15---10	4---5	12 13 7 2 34 34	14---11	4---5	12 13 6 3 34 34
12- -13	7- -2	15 10 4 5 34 34	12- -13	6- -3	14 11 4 5 34 34
1---8	14---11	1 8 14 11 34 34	1---8	15---10	1 8 15 10 34 34

169/E1	/E2	/SC\34/34	193/E1	/E2	/SC\34/34
8---1	10---15	8 1 10 15 34 34	9---4	6---15	9 4 6 15 34 34
13---12	3---6	11 14 5 4 34 34	16---5	3---10	7 14 12 1 34 34
11- -14	5- -4	13 12 3 6 34 34	7- -14	12- -1	16 5 3 10 34 34
2---7	16---9	2 7 16 9 34 34	2---11	13---8	2 11 13 8 34 34

217/E1	/E2	/SC\34/34	241/E1	/E2	/SC\34/34
10---3	5---16	10 3 5 16 34 34	11---2	5---16	11 2 5 16 34 34
15---6	4---9	8 13 11 2 34 34	14---7	4---9	8 13 10 3 34 34
8- -13	11- -2	15 6 4 9 34 34	8- -13	10- -3	14 7 4 9 34 34
1---12	14---7	1 12 14 7 34 34	1---12	15---6	1 12 15 6 34 34

265/E1	/E2	/SC\34/34	289/E1	/E2	/SC\34/34
12---1	6---15	12 1 6 15 34 34	13---2	3---16	13 2 3 16 34 34
13---8	3---10	7 14 9 4 34 34	12---7	6---9	8 11 10 5 34 34
7- -14	9- -4	13 8 3 10 34 34	8- -11	10- -5	12 7 6 9 34 34
2---11	16---5	2 11 16 5 34 34	1---14	15---4	1 14 15 4 34 34

313/E1	/E2	/SC\34/34	337/E1	/E2	/SC\34/34
14--- 1	4---15	14 1 4 15 34 34	15--- 1	4---14	15 1 4 14 34 34
11--- 8	5---10	7 12 9 6 34 34	10--- 8	5---11	6 12 9 7 34 34
7- -12	9- - 6	11 8 5 10 34 34	6- -12	9- - 7	10 8 5 11 34 34
2---13	16--- 3	2 13 16 3 34 34	3---13	16--- 2	3 13 16 2 34 34

361/E1	/E2	/SC\34/34
16--- 2	3---13	16 2 3 13 34 34
9--- 7	6---12	5 11 10 8 34 34
5- -11	10- - 8	9 7 6 12 34 34
4---14	15--- 1	4 14 15 1 34 34

[Solution Counts(n1=1/All) = 24/384] OK!

\*\* Calculated and Listed by Kanji Setsuda on  
May 5, 2014 with Apple OSX 10.9.2 & Xcode 5.1.1 \*\*

上のリストを見ればおわかりのように、24個の展開形のいずれもが補数対称型だ。どの解も、出力する順番こそ違え、全く同じ解集合を再構成した。補数対称型の原始解は、n1=1の時、元々24個しかないのだから、1個の四次元超立体二方阵が、どの解でも、その展開形の全部を知っていることを示している。これは、素晴らしい発見だった。まさに四次元超立体二方阵の特性を示している。しかし、この「奇跡」は、補数対称型のみに見られる。例えば、汎対角線型（あるいは完全型、正方連結完全型）には、24の展開形基本図がそのままでは通用しない。

### 3. 正方連結型完全四方阵を目的方阵とする場合には、次のように定義する。

0/E1	/E2	SC/	CC/ (基本図)
1---- 2	3---- 4	1 2 3 4	1 2 4 3 1
9---+10	11---+12	5 6 7 8	5 6 8 7 5
5- - 6	7- - 8	9 10 11 12	13 14 16 15 13
13----14	15----16	13 14 15 16	9 10 12 11 9
			1 2 4 3 1

\* 正方連結四数の定和式 : C=34 \*

n1+n2+n5+n6=C ... (1);	n2+n4+n6+n8=C ... (2);	n4+n3+n8+n7=C ... (3);
n3+n1+n7+n5=C ... (4);	n5+n6+n13+n14=C ... (5);	n6+n8+n14+n16=C ... (6);
n8+n7+n16+n15=C ... (7);	n7+n5+n15+n13=C ... (8);	n13+n14+n9+n14=C ... (9);
n14+n16+n10+n12=C ..(10);	n16+n15+n12+n11=C ..(11);	n15+n13+n11+n9=C ..(12);
n9+n10+n1+n2=C ..(13);	n10+n12+n2+n4=C ..(14);	n12+n11+n4+n3=C ..(15);
n11+n9+n3+n1=C ..(16);		

\* 基本条件式 : C=34 \*

n1+n2+n4+n3=C ... (1);	n1+n5+n13+n9=C ... (2);	n1+n6+n16+n11=C ... (3);
------------------------	-------------------------	--------------------------

\* 完全型補数の定和式 : CC=17 \*

n1+n16=n2+n15=n3+n14=n4+n13=n5+n12=n6+n11=n7+n10=n8+n9=CC=C/2 ... (4);

#### 3-1. プログラムは、次のように書く。

```

/** Make the 4-D Extra-Cubic Magic Objects of Order 2 and **
/** Transform them into 'Composite & Complete' Magic Squares 4x4 **
/** File: 'ECO24CCDst.c' Dictated by Kanji Setsuda **
/** on Feb.27, 2005; Revised on June 29, 2012 and on **
/** Dec.16, 2013 with Apple OSX 10.9 & Xcode 5.0.2 **
//
#include <stdio.h>
//
/** Variables *
short int cnt, cnt2;
short LSM, CC;
short nm[17], flg[17];
short csb[9], csp1[9], csp2[9], csc1[17], csc2[17], css[7];
//
/** Sub-Routines *

```

```

void stp01(void), stp02(void), stp03(void), stp04(void);
void stp05(void), stp06(void), stp07(void), stp08(void);
void stp09(void), stp10(void);
void anschksm(void);
void checksums(void);
//
/** Main Program *
int main(){
short n;
printf("\n");
printf("*** Make the 4-D Extra-Cubic Magic Objects of Order 2 and\n");
printf("  Transform them into 2 Types of Magic Squares of Order 4 **\n");
printf(" [Primitive Solutions: 4-d EC02^4; Self-complementary MS44; 'C&C' MS44;\n");
printf("  /SC: Self-Complementary Type; /CC: 'Composite & Complete' Type;]\n");
printf("\n");
LSM=34; CC=17; cnt=0;
for(n=0;n<17;n++){nm[n]=0; flg[n]=0;}
stp01();          /* Reconstruction of Object Solutions *
printf(" [Count = %d] OK!\n",cnt);
printf("\n");
printf("*** Calculated and Listed by Kanji Setsuda\n");
printf("  on Dec.16, 2013 with Apple OSX 10.9 & Xcode 5.0.2 **\n");
printf("\n");
return 0;
}
//
/** Reconstructions of Object Solutions *
/** Set n1 & n16 *
void stp01(){
short a,b;
for(a=1;a<17;a++){b=CC-a;
if((flg[a]==0)&&(flg[b]==0)){cnt2=0;
nm[1]=a; nm[16]=b;
flg[a]=1; flg[b]=1;
stp02();
flg[b]=0; flg[a]=0;}
}
}
/** Set n2 & n15 *
void stp02(){
short a,b;
for(a=1;a<17;a++){b=CC-a;
if((flg[a]==0)&&(flg[b]==0)){
nm[2]=a; nm[15]=b;
flg[a]=1; flg[b]=1;
stp03();
flg[b]=0; flg[a]=0;}
}
}
/** Set n3 & n14 *
void stp03(){
short a,b;
for(a=1;a<17;a++){b=CC-a;
if((flg[a]==0)&&(flg[b]==0)){
nm[3]=a; nm[14]=b;
flg[a]=1; flg[b]=1;
stp04();
flg[b]=0; flg[a]=0;}
}
}
/** Set n4=LSM-n1-n2-n4 & n13 *
void stp04(){
short a,b;
a=LSM-nm[1]-nm[2]-nm[3];
b=LSM-nm[16]-nm[15]-nm[14];
if((0<a)&&(a<17)&&(a+b==CC)){

```



```

        if((flg[a]==0)&&(flg[b]==0)){
            nm[4]=a; nm[13]=b;
            flg[a]=1; flg[b]=1;
            stp05();
            flg[b]=0; flg[a]=0;}}
    }
    /* Set n5 & n12 *
void stp05(){
    short a,b;
    for(a=1;a<17;a++){b=CC-a;
        if((flg[a]==0)&&(flg[b]==0)){if(nm[1]==1){cnt2=0;}
            nm[5]=a; nm[12]=b;
            flg[a]=1; flg[b]=1;
            stp06();
            flg[b]=0; flg[a]=0;}}
    }
}
    /* Set n9=LSM-n1-n5-n13 & n8 *
void stp06(){
    short a,b;
    a=LSM-nm[1]-nm[5]-nm[13];
    b=LSM-nm[16]-nm[12]-nm[4];
    if((0<a)&&(a<17)&&(a+b==CC)){
        if((flg[a]==0)&&(flg[b]==0)){
            nm[9]=a; nm[8]=b;
            flg[a]=1; flg[b]=1;
            stp07();
            flg[b]=0; flg[a]=0;}}
    }
}
    /* Set n6=LSM-n1-n2-n5 & n11 *
void stp07(){
    short a,b,c,d;
    a=LSM-nm[1]-nm[2]-nm[5];
    b=LSM-nm[16]-nm[15]-nm[12];
    c=LSM-nm[2]-nm[4]-nm[8];
    d=LSM-nm[15]-nm[13]-nm[9];
    if((0<a)&&(a<17)&&(a+b==CC)){
        if((a==c)&&(b==d)&&(flg[a]==0)&&(flg[b]==0)){
            nm[6]=a; nm[11]=b;
            flg[a]=1; flg[b]=1;
            stp08();
            flg[b]=0; flg[a]=0;}}
    }
}
    /* Set n7=LSM-n4-n3-n8 & n10 *
void stp08(){
    short a,b,c,d;
    a=LSM-nm[4]-nm[3]-nm[8];
    b=LSM-nm[13]-nm[14]-nm[9];
    c=LSM-nm[3]-nm[1]-nm[5];
    d=LSM-nm[14]-nm[16]-nm[12];
    if((0<a)&&(a<17)&&(a+b==CC)){
        if((a==c)&&(b==d)&&(flg[a]==0)&&(flg[b]==0)){
            nm[7]=a; nm[10]=b;
            flg[a]=1; flg[b]=1;
            stp09();
            flg[b]=0; flg[a]=0;}}
    }
}
    /* Check Composite Sums *
void stp09(){
    short sm1,sm2,sm3,sm4;
    sm1=nm[5]+nm[6]+nm[13]+nm[14];
    sm2=nm[6]+nm[8]+nm[14]+nm[16];
    sm3=nm[8]+nm[7]+nm[16]+nm[15];
    sm4=nm[7]+nm[5]+nm[15]+nm[13];
    if((sm1==LSM)&&(sm2==LSM)&&(sm3==LSM)&&(sm4==LSM)){stp10();}
}

```

```

/** Print the Solution *
void stp10(){
    cnt++; cnt2++;
    if(cnt2==1){checksums();
        anschksm();}
}
//
/** Print Single Answers with precise Check-sums *
void anschksm(){
    printf("%4d#/E1      /E2      ", cnt);
    if(css[0]==16){printf("/SC");}
    else if(css[2]==8){printf("/PD");}
    else if(css[4]>0){printf("/ST");}
    else{printf(" ");}
    printf("IRw CLIP1 P2IComposite 4      ");
    if((css[3]==8)&&(css[6]==16)){printf("/CC");}
    else if(css[1]==16){printf("/SC");}
    else if(css[3]==8){printf("/PD");}
    else if(css[5]>0){printf("/ST");}
    else{printf(" ");}
    printf("IRw CLIP1 P2IComposite 4\n");
    printf("%3d---%2d%5d---%2d", nm[1], nm[2], nm[3], nm[4]);
    printf("%7d%3d%3d%3d|%2d%3d|%2d%3d|%2d%3d%3d%3d",
        nm[1], nm[2], nm[3], nm[4], csb[1], csb[5], csp1[1], csp1[5], csc1[1], csc1[2], csc1[3], csc1[4]);
    printf("%4d%3d%3d%3d|%2d%3d|%2d%3d|%2d%3d%3d%3d\n",
        nm[1], nm[2], nm[4], nm[3], csb[1], csb[5], csp2[1], csp2[5], csc2[1], csc2[2], csc2[3], csc2[4]);
    printf(" |%2d---%2d |%2d---%2d", nm[9], nm[10], nm[11], nm[12]);
    printf("%5d%3d%3d%3d|%2d%3d|%2d%3d|%2d%3d%3d%3d",
        nm[5], nm[6], nm[7], nm[8], csb[2], csb[6], csp1[2], csp1[6], csc1[5], csc1[6], csc1[7], csc1[8]);
    printf("%4d%3d%3d%3d|%2d%3d|%2d%3d|%2d%3d%3d%3d\n",
        nm[5], nm[6], nm[8], nm[7], csb[2], csb[6], csp2[2], csp2[6], csc2[5], csc2[6], csc2[7], csc2[8]);
    printf("%3d-|-%2d |%3d-|-%2d |", nm[5], nm[6], nm[7], nm[8]);
    printf("%5d%3d%3d%3d|%2d%3d|%2d%3d|%2d%3d%3d%3d",
        nm[9], nm[10], nm[11], nm[12], csb[3], csb[7], csp1[3], csp1[7], csc1[9], csc1[10], csc1[11], csc1[12]);
    printf("%4d%3d%3d%3d|%2d%3d|%2d%3d|%2d%3d%3d%3d\n",
        nm[13], nm[14], nm[16], nm[15], csb[4], csb[8], csp2[3], csp2[7], csc2[9], csc2[10], csc2[11], csc2[12]);
    printf("%5d---%2d%5d---%2d", nm[13], nm[14], nm[15], nm[16]);
    printf("%5d%3d%3d%3d|%2d%3d|%2d%3d|%2d%3d%3d%3d",
        nm[13], nm[14], nm[15], nm[16], csb[4], csb[8], csp1[4], csp1[8], csc1[13], csc1[14], csc1[15], csc1[16]);
    printf("%4d%3d%3d%3d|%2d%3d|%2d%3d|%2d%3d%3d%3d\n",
        nm[9], nm[10], nm[12], nm[11], csb[3], csb[7], csp2[4], csp2[8], csc2[13], csc2[14], csc2[15], csc2[16]);
    printf("\n");
}
//
/** Check Sums of every Conditions *
void checksums(){
    short m, n;
    short sc[17]={0,11,12,10, 9,15,16,14,13, 7, 8, 6, 5, 3, 4, 2, 1};
    /** for Basic Conditions *
        csb[1]=nm[1]+nm[2]+nm[3]+nm[4];          csb[2]=nm[5]+nm[6]+nm[7]+nm[8];
        csb[3]=nm[9]+nm[10]+nm[11]+nm[12];      csb[4]=nm[13]+nm[14]+nm[15]+nm[16];
        csb[5]=nm[1]+nm[5]+nm[9]+nm[13];        csb[6]=nm[2]+nm[6]+nm[10]+nm[14];
        csb[7]=nm[3]+nm[7]+nm[11]+nm[15];      csb[8]=nm[4]+nm[8]+nm[12]+nm[16];
    /** for Pandiagonal Conditions: Type 1 *
        csp1[1]=nm[1]+nm[6]+nm[11]+nm[16];     csp1[2]=nm[2]+nm[7]+nm[12]+nm[13];
        csp1[3]=nm[3]+nm[8]+nm[9]+nm[14];      csp1[4]=nm[4]+nm[5]+nm[10]+nm[15];
        csp1[5]=nm[1]+nm[8]+nm[11]+nm[14];     csp1[6]=nm[2]+nm[5]+nm[12]+nm[15];
        csp1[7]=nm[3]+nm[6]+nm[9]+nm[16];      csp1[8]=nm[4]+nm[7]+nm[10]+nm[13];
    /** for Pandiagonal Conditions: Type 2 *
        csp2[1]=nm[1]+nm[6]+nm[16]+nm[11];     csp2[2]=nm[2]+nm[8]+nm[15]+nm[9];
        csp2[3]=nm[4]+nm[7]+nm[13]+nm[10];     csp2[4]=nm[3]+nm[5]+nm[14]+nm[12];
        csp2[5]=nm[1]+nm[7]+nm[16]+nm[10];     csp2[6]=nm[2]+nm[5]+nm[15]+nm[12];
        csp2[7]=nm[4]+nm[6]+nm[13]+nm[11];     csp2[8]=nm[3]+nm[8]+nm[14]+nm[9];
    /** for Composite Conditions: Type 1 *
        csc1[1]=nm[1]+nm[2]+nm[5]+nm[6];        csc1[2]=nm[2]+nm[3]+nm[6]+nm[7];
        csc1[3]=nm[3]+nm[4]+nm[7]+nm[8];        csc1[4]=nm[4]+nm[1]+nm[8]+nm[5];

```

```

csc1[5]=nm[5]+nm[6]+nm[9]+nm[10];      csc1[6]=nm[6]+nm[7]+nm[10]+nm[11];
csc1[7]=nm[7]+nm[8]+nm[11]+nm[12];      csc1[8]=nm[8]+nm[5]+nm[12]+nm[9];
csc1[9]=nm[9]+nm[10]+nm[13]+nm[14];      csc1[10]=nm[10]+nm[11]+nm[14]+nm[15];
csc1[11]=nm[11]+nm[12]+nm[15]+nm[16];    csc1[12]=nm[12]+nm[9]+nm[16]+nm[13];
csc1[13]=nm[13]+nm[14]+nm[1]+nm[2];      csc1[14]=nm[14]+nm[15]+nm[2]+nm[3];
csc1[15]=nm[15]+nm[16]+nm[3]+nm[4];      csc1[16]=nm[16]+nm[13]+nm[4]+nm[1];
/* for Composite Conditions: Type 2 *
csc2[1]=nm[1]+nm[2]+nm[5]+nm[6];          csc2[2]=nm[2]+nm[4]+nm[6]+nm[8];
csc2[3]=nm[4]+nm[3]+nm[8]+nm[7];          csc2[4]=nm[3]+nm[1]+nm[7]+nm[5];
csc2[5]=nm[5]+nm[6]+nm[13]+nm[14];        csc2[6]=nm[6]+nm[8]+nm[14]+nm[16];
csc2[7]=nm[8]+nm[7]+nm[16]+nm[15];        csc2[8]=nm[7]+nm[5]+nm[15]+nm[13];
csc2[9]=nm[13]+nm[14]+nm[9]+nm[10];       csc2[10]=nm[14]+nm[16]+nm[10]+nm[12];
csc2[11]=nm[16]+nm[15]+nm[12]+nm[11];     csc2[12]=nm[15]+nm[13]+nm[11]+nm[9];
csc2[13]=nm[9]+nm[10]+nm[1]+nm[2];        csc2[14]=nm[10]+nm[12]+nm[2]+nm[4];
csc2[15]=nm[12]+nm[11]+nm[4]+nm[3];       csc2[16]=nm[11]+nm[9]+nm[3]+nm[1];
/* for Conditions of Types **
css[0]=0;
for(n=1;n<17;n++){m=17-n; if(nm[n]+nm[m]==17){css[0]++;}}
css[1]=0;
for(n=1;n<17;n++){m=sc[n]; if(nm[m]+nm[17-m]==17){css[1]++;}}
css[2]=0;
for(n=1;n<9;n++){if(csp1[n]==34){css[2]++;}}
css[3]=0;
for(n=1;n<9;n++){if(csp2[n]==34){css[3]++;}}
css[4]=0;
if((csp1[1]==34)&&(csp1[8]==34)){css[4]++;}
css[5]=0;
if((csp2[1]==34)&&(csp2[8]==34)){css[5]++;}
css[6]=0;
for(n=1;n<17;n++){if(csc2[n]==34){css[6]++;}}
}
//

```

3-2. \*\* 四次元二方陣を解いて対称型と完全型の四方陣を同時に得た結果：チェックサム付き \*\*

[/SC: 補数対称型; /CC: 正方連結完全型; Rw: 行和; Cl: 列和; P1, P2: 汎対角線の和; C4: 正方連結四数の和;]

1#/E1	/E2	1/SC	Rw	Cl	P1	P2	Composite	4	/CC	Rw	Cl	P1	P2	Composite	4													
1---8	12---13	1	8	12	13	34	34	34	14	34	38	34	30	1	8	13	12	34	34	34	34	34	34	34	34	34	34	
15---10	6---3	14	11	7	2	34	34	22	34	50	34	18	34	14	11	2	7	34	34	34	34	34	34	34	34	34	34	34
14- -11	7- -2	15	10	6	3	34	34	34	54	34	30	34	38	4	5	16	9	34	34	34	34	34	34	34	34	34	34	34
4---5	9---16	4	5	9	16	34	34	46	34	18	34	50	34	15	10	3	6	34	34	34	34	34	34	34	34	34	34	34

  

2#/E1	/E2	2/SC	Rw	Cl	P1	P2	Composite	4	/CC	Rw	Cl	P1	P2	Composite	4													
1---8	12---13	1	8	12	13	34	34	34	16	34	36	34	32	1	8	13	12	34	34	34	34	34	34	34	34	34	34	
14---11	7---2	15	10	6	3	34	34	20	34	50	34	18	34	15	10	3	6	34	34	34	34	34	34	34	34	34	34	34
15- -10	6- -3	14	11	7	2	34	34	34	52	34	32	34	36	4	5	16	9	34	34	34	34	34	34	34	34	34	34	34
4---5	9---16	4	5	9	16	34	34	48	34	18	34	50	34	14	11	2	7	34	34	34	34	34	34	34	34	34	34	34

  

3#/E1	/E2	3/SC	Rw	Cl	P1	P2	Composite	4	/CC	Rw	Cl	P1	P2	Composite	4													
1---8	14---11	1	8	14	11	34	34	34	10	34	42	34	26	1	8	11	14	34	34	34	34	34	34	34	34	34	34	34
15---10	4---5	12	13	7	2	34	34	26	34	50	34	18	34	12	13	2	7	34	34	34	34	34	34	34	34	34	34	34
12- -13	7- -2	15	10	4	5	34	34	34	58	34	26	34	42	6	3	16	9	34	34	34	34	34	34	34	34	34	34	34
6---3	9---16	6	3	9	16	34	34	42	34	18	34	50	34	15	10	5	4	34	34	34	34	34	34	34	34	34	34	34

  

4#/E1	/E2	4/SC	Rw	Cl	P1	P2	Composite	4	/CC	Rw	Cl	P1	P2	Composite	4													
1---8	14---11	1	8	14	11	34	34	34	16	34	36	34	32	1	8	11	14	34	34	34	34	34	34	34	34	34	34	34
12---13	7---2	15	10	4	5	34	34	20	34	50	34	18	34	15	10	5	4	34	34	34	34	34	34	34	34	34	34	34
15- -10	4- -5	12	13	7	2	34	34	34	52	34	32	34	36	6	3	16	9	34	34	34	34	34	34	34	34	34	34	34
6---3	9---16	6	3	9	16	34	34	48	34	18	34	50	34	12	13	2	7	34	34	34	34	34	34	34	34	34	34	34

  

5#/E1	/E2	5/SC	Rw	Cl	P1	P2	Composite	4	/CC	Rw	Cl	P1	P2	Composite	4													
1---8	15---10	1	8	15	10	34	34	34	10	34	42	34	26	1	8	10	15	34	34	34	34	34	34	34	34	34	34	34
14---11	4---5	12	13	6	3	34	34	26	34	50	34	18	34	12	13	3	6	34	34	34	34	34	34	34	34	34	34	34
12- -13	6- -3	14	11	4	5	34	34	34	58	34	26	34	42	7	2	16	9	34	34	34	34	34	34	34	34	34	34	34
7---2	9---16	7	2	9	16	34	34	42	34	18	34	50	34	14	11	5	4	34	34	34	34	34	34	34	34	34	34	34

6#/E1	/E2	6/SC	Rw	CI	P1	P2	Composite 4	/CC	Rw	CI	P1	P2	Composite 4
1---8	15---10	1	8	15	10	34	34	34	14	34	38	34	30
12---13	6---3	14	11	4	5	34	34	22	34	50	34	18	34
14- -11	4- -5	12	13	6	3	34	34	34	54	34	30	34	38
7---2	9---16	7	2	9	16	34	34	46	34	18	34	50	34

7#/E1	/E2	7/SC	Rw	CI	P1	P2	Composite 4	/CC	Rw	CI	P1	P2	Composite 4
1---12	8---13	1	12	8	13	34	34	34	22	34	38	34	30
15---6	10---3	14	7	11	2	34	34	30	34	42	34	26	34
14- -7	11- -2	15	6	10	3	34	34	34	46	34	30	34	38
4---9	5---16	4	9	5	16	34	34	38	34	26	34	42	34

8#/E1	/E2	8/SC	Rw	CI	P1	P2	Composite 4	/CC	Rw	CI	P1	P2	Composite 4
1---12	8---13	1	12	8	13	34	34	34	24	34	36	34	32
14---7	11---2	15	6	10	3	34	34	28	34	42	34	26	34
15- -6	10- -3	14	7	11	2	34	34	34	44	34	32	34	36
4---9	5---16	4	9	5	16	34	34	40	34	26	34	42	34

9#/E1	/E2	9/SC	Rw	CI	P1	P2	Composite 4	/CC	Rw	CI	P1	P2	Composite 4
1---12	14---7	1	12	14	7	34	34	34	10	34	50	34	18
15---6	4---9	8	13	11	2	34	34	42	34	42	34	26	34
8- -13	11- -2	15	6	4	9	34	34	34	58	34	18	34	50
10---3	5---16	10	3	5	16	34	34	26	34	26	34	42	34

10#/E1	/E2	10/SC	Rw	CI	P1	P2	Composite 4	/CC	Rw	CI	P1	P2	Composite 4
1---12	14---7	1	12	14	7	34	34	34	24	34	36	34	32
8---13	11---2	15	6	4	9	34	34	28	34	42	34	26	34
15- -6	4- -9	8	13	11	2	34	34	34	44	34	32	34	36
10---3	5---16	10	3	5	16	34	34	40	34	26	34	42	34

11#/E1	/E2	11/SC	Rw	CI	P1	P2	Composite 4	/CC	Rw	CI	P1	P2	Composite 4
1---12	15---6	1	12	15	6	34	34	34	10	34	50	34	18
14---7	4---9	8	13	10	3	34	34	42	34	42	34	26	34
8- -13	10- -3	14	7	4	9	34	34	34	58	34	18	34	50
11---2	5---16	11	2	5	16	34	34	26	34	26	34	42	34

12#/E1	/E2	12/SC	Rw	CI	P1	P2	Composite 4	/CC	Rw	CI	P1	P2	Composite 4
1---12	15---6	1	12	15	6	34	34	34	22	34	38	34	30
8---13	10---3	14	7	4	9	34	34	30	34	42	34	26	34
14- -7	4- -9	8	13	10	3	34	34	34	46	34	30	34	38
11---2	5---16	11	2	5	16	34	34	38	34	26	34	42	34

13#/E1	/E2	13/SC	Rw	CI	P1	P2	Composite 4	/CC	Rw	CI	P1	P2	Composite 4
1---14	8---11	1	14	8	11	34	34	34	22	34	42	34	26
15---4	10---5	12	7	13	2	34	34	38	34	38	34	30	34
12- -7	13- -2	15	4	10	5	34	34	34	46	34	26	34	42
6---9	3---16	6	9	3	16	34	34	30	34	30	34	38	34

14#/E1	/E2	14/SC	Rw	CI	P1	P2	Composite 4	/CC	Rw	CI	P1	P2	Composite 4
1---14	8---11	1	14	8	11	34	34	34	28	34	36	34	32
12---7	13---2	15	4	10	5	34	34	32	34	38	34	30	34
15- -4	10- -5	12	7	13	2	34	34	34	40	34	32	34	36
6---9	3---16	6	9	3	16	34	34	36	34	30	34	38	34

15#/E1	/E2	15/SC	Rw	CI	P1	P2	Composite 4	/CC	Rw	CI	P1	P2	Composite 4
1---14	12---7	1	14	12	7	34	34	34	14	34	50	34	18
15---4	6---9	8	11	13	2	34	34	46	34	38	34	30	34
8- -11	13- -2	15	4	6	9	34	34	34	54	34	18	34	50
10---5	3---16	10	5	3	16	34	34	22	34	30	34	38	34

16#/E1	/E2	16/SC	Rw	CI	P1	P2	Composite 4	/CC	Rw	CI	P1	P2	Composite 4
1---14	12---7	1	14	12	7	34	34	34	28	34	36	34	32
8---11	13---2	15	4	6	9	34	34	32	34	38	34	30	34
15- -4	6- -9	8	11	13	2	34	34	34	40	34	32	34	36
10---5	3---16	10	5	3	16	34	34	36	34	30	34	38	34

17#/E1	/E2	17/SC	Rw	CI	P1	P2	Composite 4	/CC	Rw	CI	P1	P2	Composite 4		
1---14	15---4	1	14	15	4	34	34	34	14	34	50	34	18		
12---7	6---9	8	11	10	5	34	34	46	34	38	34	30	34		
8- -11	10- -5	12	7	6	9	34	34	34	54	34	18	34	50		
13---2	3---16	13	2	3	16	34	34	22	34	30	34	38	34		
18#/E1	/E2	18/SC	Rw	CI	P1	P2	Composite 4	/CC	Rw	CI	P1	P2	Composite 4		
1---14	15---4	1	14	15	4	34	34	34	22	34	42	34	26		
8---11	10---5	12	7	6	9	34	34	38	34	38	34	30	34		
12- -7	6- -9	8	11	10	5	34	34	34	46	34	26	34	42		
13---2	3---16	13	2	3	16	34	34	30	34	30	34	38	34		
19#/E1	/E2	19/SC	Rw	CI	P1	P2	Composite 4	/CC	Rw	CI	P1	P2	Composite 4		
1---15	8---10	1	15	8	10	34	34	34	24	34	42	34	26		
14---4	11---5	12	6	13	3	34	34	40	34	36	34	32	34		
12- -6	13- -3	14	4	11	5	34	34	34	44	34	26	34	42		
7---9	2---16	7	9	2	16	34	34	28	34	32	34	36	34		
20#/E1	/E2	20/SC	Rw	CI	P1	P2	Composite 4	/CC	Rw	CI	P1	P2	Composite 4		
1---15	8---10	1	15	8	10	34	34	34	28	34	38	34	30		
12---6	13---3	14	4	11	5	34	34	36	34	36	34	32	34		
14- -4	11- -5	12	6	13	3	34	34	34	40	34	30	34	38		
7---9	2---16	7	9	2	16	34	34	32	34	32	34	36	34		
21#/E1	/E2	21/SC	Rw	CI	P1	P2	Composite 4	/CC	Rw	CI	P1	P2	Composite 4		
1---15	12---6	1	15	12	6	34	34	34	16	34	50	34	18		
14---4	7---9	8	10	13	3	34	34	48	34	36	34	32	34		
8- -10	13- -3	14	4	7	9	34	34	34	52	34	18	34	50		
11---5	2---16	11	5	2	16	34	34	20	34	32	34	36	34		
22#/E1	/E2	22/SC	Rw	CI	P1	P2	Composite 4	/CC	Rw	CI	P1	P2	Composite 4		
1---15	12---6	1	15	12	6	34	34	34	28	34	38	34	30		
8---10	13---3	14	4	7	9	34	34	36	34	36	34	32	34		
14- -4	7- -9	8	10	13	3	34	34	34	40	34	30	34	38		
11---5	2---16	11	5	2	16	34	34	32	34	32	34	36	34		
23#/E1	/E2	23/SC	Rw	CI	P1	P2	Composite 4	/CC	Rw	CI	P1	P2	Composite 4		
1---15	14---4	1	15	14	4	34	34	34	16	34	50	34	18		
12---6	7---9	8	10	11	5	34	34	48	34	36	34	32	34		
8- -10	11- -5	12	6	7	9	34	34	34	52	34	18	34	50		
13---3	2---16	13	3	2	16	34	34	20	34	32	34	36	34		
24#/E1	/E2	24/SC	Rw	CI	P1	P2	Composite 4	/CC	Rw	CI	P1	P2	Composite 4		
1---15	14---4	1	15	14	4	34	34	34	24	34	42	34	26		
8---10	11---5	12	6	7	9	34	34	40	34	36	34	32	34		
12- -6	7- -9	8	10	11	5	34	34	34	44	34	26	34	42		
13---3	2---16	13	3	2	16	34	34	28	34	32	34	36	34		
25#/E1	/E2	25/SC		/CC	49#/E1	/E2	49/SC		/CC						
2---7	11---14	2	7	11	14	2	7	14	11	3---6	10---15	3	6	10	15
16---9	5---4	13	12	8	1	13	12	1	8	16---9	5---4	13	12	8	1
13- -12	8- -1	16	9	5	4	3	6	15	10	13- -12	8- -1	16	9	5	4
3---6	10---15	3	6	10	15	16	9	4	5	2---7	11---14	2	7	11	14
73#/E1	/E2	73/SC		/CC	97#/E1	/E2	97/SC		/CC						
4---5	9---16	4	5	9	16	4	5	16	9	5---4	10---15	5	4	10	15
15---10	6---3	14	11	7	2	14	11	2	7	16---9	3---6	11	14	8	1
14- -11	7- -2	15	10	6	3	1	8	13	12	11- -14	8- -1	16	9	3	6
1---8	12---13	1	8	12	13	15	10	3	6	2---7	13---12	2	7	13	12
121#/E1	/E2	121/SC		/CC	145#/E1	/E2	145/SC		/CC						
6---3	9---16	6	3	9	16	6	3	16	9	7---2	9---16	7	2	9	16
15---10	4---5	12	13	7	2	12	13	2	7	14---11	4---5	12	13	6	3
12- -13	7- -2	15	10	4	5	1	8	11	14	12- -13	6- -3	14	11	4	5
1---8	14---11	1	8	14	11	15	10	5	4	1---8	15---10	1	8	15	10

169#/E1	/E2	169/SC	/CC	193#/E1	/E2	193/SC	/CC
8---1	10---15	8 1 10 15	8 1 15 10	9---4	6---15	9 4 6 15	9 4 15 6
13---12	3---6	11 14 5 4	11 14 4 5	16---5	3---10	7 14 12 1	7 14 1 12
11- 14	5- -4	13 12 3 6	2 7 9 16	7- 14	12- -1	16 5 3 10	2 11 8 13
2---7	16---9	2 7 16 9	13 12 6 3	2---11	13---8	2 11 13 8	16 5 10 3
217#/E1	/E2	217/SC	/CC	241#/E1	/E2	241/SC	/CC
10---3	5---16	10 3 5 16	10 3 16 5	11---2	5---16	11 2 5 16	11 2 16 5
15---6	4---9	8 13 11 2	8 13 2 11	14---7	4---9	8 13 10 3	8 13 3 10
8- 13	11- -2	15 6 4 9	1 12 7 14	8- 13	10- -3	14 7 4 9	1 12 6 15
1---12	14---7	1 12 14 7	15 6 9 4	1---12	15---6	1 12 15 6	14 7 9 4
265#/E1	/E2	265/SC	/CC	289#/E1	/E2	289/SC	/CC
12---1	6---15	12 1 6 15	12 1 15 6	13---2	3---16	13 2 3 16	13 2 16 3
13---8	3---10	7 14 9 4	7 14 4 9	12---7	6---9	8 11 10 5	8 11 5 10
7- 14	9- -4	13 8 3 10	2 11 5 16	8- 11	10- -5	12 7 6 9	1 14 4 15
2---11	16---5	2 11 16 5	13 8 10 3	1---14	15---4	1 14 15 4	12 7 9 6
313#/E1	/E2	313/SC	/CC	337#/E1	/E2	337/SC	/CC
14---1	4---15	14 1 4 15	14 1 15 4	15---1	4---14	15 1 4 14	15 1 14 4
11---8	5---10	7 12 9 6	7 12 6 9	10---8	5---11	6 12 9 7	6 12 7 9
7- 12	9- -6	11 8 5 10	2 13 3 16	6- 12	9- -7	10 8 5 11	3 13 2 16
2---13	16---3	2 13 16 3	11 8 10 5	3---13	16---2	3 13 16 2	10 8 11 5
361#/E1	/E2	361/SC	/CC				
16---2	3---13	16 2 3 13	16 2 13 3				
9---7	6---12	5 11 10 8	5 11 8 10				
5- 11	10- -8	9 7 6 12	4 14 1 15				
4---14	15---1	4 14 15 1	9 7 12 6				

[Count = 384] OK!

\*\* Calculated and Listed by Kanji Setsuda on Dec. 16, 2013 with Apple OSX 10.9 & Xcode 5.0.2 \*\*

3-3. 補数対称型と正方連結完全型では、今までの作り方ならば定義も解法プログラムも全く違うのに、今回はそれぞれの結果が同時に得られた。何故なのか？

理由の一つは、同じ四次元超立体二方陣を解いて得たから。その証拠には、2つのリストの最初の2つの小立方体の構成内容が全く同じである。元が同じならば、それを展開して得られる2種類の二次元四方陣が全く同じ顔ぶれになるのは当然の話だ。

もう一つの理由は、この2種類の二次元四方陣の間の緊密な関係だ。採用した三-四変換法がそれを示している。基本図に埋め込んだこの変換法が正しい効果を発揮している。

正方連結型完全四方陣では、定義に16本の「正方連結四数の定和式」が登場する。これらが、四次元超立体二方陣の2つの小立方体二方陣の各面の「頂点連結四数の定和」に緊密に結び付いていることが、基本図のリストを見ればおわかりになる。

つまり、正方連結型完全四方陣は、四次元超立体二方陣の紛れのない申し子なのだ。

4. 昔から不思議に思われて来たことだが、四方陣の世界では、汎対角線型、完全型、正方連結完全型の3タイプが同じ解集合を持ち、論理的に「同値」の関係にある。八次の場合には全然別の解集合を持つのに、四次の場合は何故に同じになるのか。

その根本的な理由が、同一の四次元超立体二方陣に由来すると、私は考えている。

その構造を具体的に示せるのが、新構成法の特長の一つだと、思っている。

(あなたが何でも疑う人ならば、汎対角線型、完全型の四方陣を自分で定義してみて、実際に作って見られよ。ただし、出発点の基本図だけは、正方連結完全型と同じにして …。)

しかし、今あらためて不思議に思う。四次元超立体二方陣とは一体何なのだろうか。

哲学好きの人ならば、この超立体二方陣を理想世界の「アイデア」と考えるだろう。或いは、創造主が胸中に抱いた「概念」と考えるかも知れない。四次元世界を直視できる完全存在＝神ならば、きっとこの魔方陣の完全なる対称性をそのまま楽しめるだろうから。

それが現実の世界に「現象」として押し出されて来ると、我らが慣れ親しんだ何種類もの二次元四方陣群になる。

低次元に分解する時に、元の対称性の一部が失われてしまうが、そのかわりに「個性」を獲得して、補数対称型・完全型・正方連結型などになる。…  
そして、これらの「贈り物」ならば理解できる我々不完全なる人間たちを、大いに楽しませてくれているわけだ（実は苦しませるだけかもしれないが …）。

(最新稿 : Written by 摂田 寛二 (Kanji Setsuda) on May 5, 2014;  
計算・作表・執筆 : with Apple OSX 10.9.2 and Xcode 5.1.1;

---

E-Mail Address: <jag12001@nifty.com>