

Part 4 : "New Advanced Study of Magic Squares and Cubes"  
 Chapter 4 : Commentary Articles No.2 by Kanji Setsuda:  
 "Various Arts and Tools for Studying Magic Squares"

Section 15-3: Make 'Complete Euler Cubes' of Order 5 for  
 the Simultaneous Type: Both Self-complementary and Pan-triagonal

0. What I like to do in this Section:

I would like to challenge next to make the Simultaneous Magic Cubes of Order 5: Self-complementary and Pan-triagonal. I plan to build the 'Complete Euler Type', under the Positional Number System of Base 5 and count the standard solutions.

1. What the 'Simultaneous' Type means:

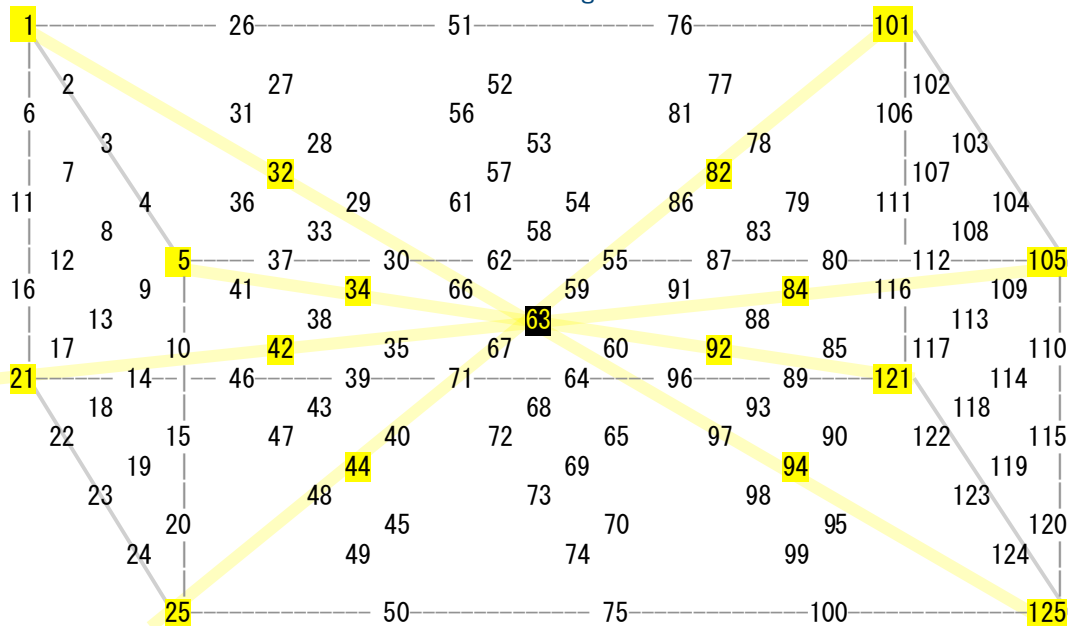
This type has both of the properties in common at the same time: both of (1) the Self-complementary type and of (2) the Pan-triagonal one.

As a consequence, the definition of Simultaneous type reflects both sets of their original definitions and has all of them in itself now.

2. How to define this type of object with the Basic Form and Basic Equations:

Let me show you the definition with the diagram and simultaneous equations.

\*\* Basic Form and Basic Conditions for Magic Cubes of Order 5 \*\*



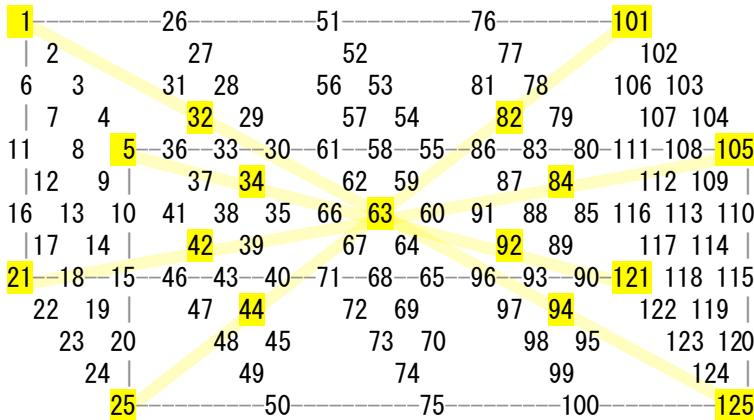
\*\* Basic Conditions for Rows, Columns and Pillars of MC555: C=315 \*\*

$n_1+n_2+n_3+n_4+n_5=C;$	$n_1+n_6+n_{11}+n_{16}+n_{21}=C;$	$n_1+n_{26}+n_{51}+n_{76}+n_{101}=C;$
$n_6+n_7+n_8+n_9+n_{10}=C;$	$n_2+n_7+n_{12}+n_{17}+n_{22}=C;$	$n_2+n_{27}+n_{52}+n_{77}+n_{102}=C;$
$n_{11}+n_{12}+n_{13}+n_{14}+n_{15}=C;$	$n_3+n_8+n_{13}+n_{18}+n_{23}=C;$	$n_3+n_{28}+n_{53}+n_{78}+n_{103}=C;$
$n_{16}+n_{17}+n_{18}+n_{19}+n_{20}=C;$	$n_4+n_9+n_{14}+n_{19}+n_{24}=C;$	$n_4+n_{29}+n_{54}+n_{79}+n_{104}=C;$
$n_{21}+n_{22}+n_{23}+n_{24}+n_{25}=C;$	$n_5+n_{10}+n_{15}+n_{20}+n_{25}=C;$	$n_5+n_{30}+n_{55}+n_{80}+n_{105}=C;$
$n_{26}+n_{27}+n_{28}+n_{29}+n_{30}=C;$	$n_{26}+n_{31}+n_{36}+n_{41}+n_{46}=C;$	$n_6+n_{31}+n_{56}+n_{81}+n_{106}=C;$
$n_{31}+n_{32}+n_{33}+n_{34}+n_{35}=C;$	$n_{27}+n_{32}+n_{37}+n_{42}+n_{47}=C;$	$n_7+n_{32}+n_{57}+n_{82}+n_{107}=C;$
$n_{36}+n_{37}+n_{38}+n_{39}+n_{40}=C;$	$n_{28}+n_{33}+n_{38}+n_{43}+n_{48}=C;$	$n_8+n_{33}+n_{58}+n_{83}+n_{108}=C;$
$n_{41}+n_{42}+n_{43}+n_{44}+n_{45}=C;$	$n_{29}+n_{34}+n_{39}+n_{44}+n_{49}=C;$	$n_9+n_{34}+n_{59}+n_{84}+n_{109}=C;$
$n_{46}+n_{47}+n_{48}+n_{49}+n_{50}=C;$	$n_{30}+n_{35}+n_{40}+n_{45}+n_{50}=C;$	$n_{10}+n_{35}+n_{60}+n_{85}+n_{110}=C;$
$n_{51}+n_{52}+n_{53}+n_{54}+n_{55}=C;$	$n_{51}+n_{56}+n_{61}+n_{66}+n_{71}=C;$	$n_{11}+n_{36}+n_{61}+n_{86}+n_{111}=C;$
$n_{56}+n_{57}+n_{58}+n_{59}+n_{60}=C;$	$n_{52}+n_{57}+n_{62}+n_{67}+n_{72}=C;$	$n_{12}+n_{37}+n_{62}+n_{87}+n_{112}=C;$
$n_{61}+n_{62}+n_{63}+n_{64}+n_{65}=C;$	$n_{53}+n_{58}+n_{63}+n_{68}+n_{73}=C;$	$n_{13}+n_{38}+n_{63}+n_{88}+n_{113}=C;$
$n_{66}+n_{67}+n_{68}+n_{69}+n_{70}=C;$	$n_{54}+n_{59}+n_{64}+n_{69}+n_{74}=C;$	$n_{14}+n_{39}+n_{64}+n_{89}+n_{114}=C;$

$n71+n72+n73+n74+n75=C;$	$n55+n60+n65+n70+n75=C;$	$n15+n40+n65+n90+n115=C;$
$n76+n77+n78+n79+n80=C;$	$n76+n81+n86+n91+n96=C;$	$n16+n41+n66+n91+n116=C;$
$n81+n82+n83+n84+n85=C;$	$n77+n82+n87+n92+n97=C;$	$n17+n42+n67+n92+n117=C;$
$n86+n87+n88+n89+n90=C;$	$n78+n83+n88+n93+n98=C;$	$n18+n43+n68+n93+n118=C;$
$n91+n92+n93+n94+n95=C;$	$n79+n84+n89+n94+n99=C;$	$n19+n44+n69+n94+n119=C;$
$n96+n97+n98+n99+n100=C;$	$n80+n85+n90+n95+n100=C;$	$n20+n45+n70+n95+n120=C;$
$n101+n102+n103+n104+n105=C;$	$n101+n106+n111+n116+n121=C;$	$n21+n46+n71+n96+n121=C;$
$n106+n107+n108+n109+n110=C;$	$n102+n107+n112+n117+n122=C;$	$n22+n47+n72+n97+n122=C;$
$n111+n112+n113+n114+n115=C;$	$n103+n108+n113+n118+n123=C;$	$n23+n48+n73+n98+n123=C;$
$n116+n117+n118+n119+n120=C;$	$n104+n109+n114+n119+n124=C;$	$n24+n49+n74+n99+n124=C;$
$n121+n122+n123+n124+n125=C;$	$n105+n110+n115+n120+n125=C;$	$n25+n50+n75+n100+n125=C;$

**\*\* Conditions for Symmetric Self-Complementary Pairs of MC555: SC=126 \*\***

$n1+n125=n2+n124=n3+n123=n4+n122=n5+n121=n6+n120=n7+n119=n8+n118=n9+n117=$   
 $n10+n116=n11+n115=n12+n114=n13+n113=n14+n112=n15+n111=n16+n110=n17+n109=n18+n108=$   
 $n19+n107=n20+n106=n21+n105=n22+n104=n23+n103=n24+n102=n25+n101=n26+n100=n27+n99=$   
 $n28+n98=n29+n97=n30+n96=n31+n95=n32+n94=n33+n93=n34+n92=n35+n91=n36+n90=$   
 $n37+n89=n38+n88=n39+n87=n40+n86=n41+n85=n42+n84=n43+n83=n44+n82=n45+n81=$   
 $n46+n80=n47+n79=n48+n78=n49+n77=n50+n76=n51+n75=n52+n74=n53+n73=n54+n72=$   
 $n55+n71=n56+n70=n57+n69=n58+n68=n59+n67=n60+n66=n61+n65=n62+n64=n63+n63=SC \dots scc$



**\*\* The 100 Pan-triagonal Conditions for Pan-magic Cubes of Order 5: C=315 \*\***

$n1+n32+n63+n94+n125=C$	... pt11;	$n5+n34+n63+n92+n121=C$	... pt12;
$n21+n42+n63+n84+n105=C$	... pt13;	$n25+n44+n63+n82+n101=C$	... pt14;
$n2+n33+n64+n95+n121=C$	... pt21;	$n1+n35+n64+n93+n122=C$	... pt22;
$n22+n43+n64+n85+n101=C$	... pt23;	$n21+n45+n64+n83+n102=C$	... pt24;
$n3+n34+n65+n91+n122=C$	... pt31;	$n2+n31+n65+n94+n123=C$	... pt32;
$n23+n44+n65+n81+n102=C$	... pt33;	$n22+n41+n65+n84+n103=C$	... pt34;
$n4+n35+n61+n92+n123=C$	... pt41;	$n3+n32+n61+n95+n124=C$	... pt42;
$n24+n45+n61+n82+n103=C$	... pt43;	$n23+n42+n61+n85+n104=C$	... pt44;
$n5+n31+n62+n93+n124=C$	... pt51;	$n4+n33+n62+n91+n125=C$	... pt52;
$n25+n41+n62+n83+n104=C$	... pt53;	$n24+n43+n62+n81+n105=C$	... pt54;
$n6+n37+n68+n99+n105=C$	... pt61;	$n10+n39+n68+n97+n101=C$	... pt62;
$n1+n47+n68+n89+n110=C$	... pt63;	$n5+n49+n68+n87+n106=C$	... pt64;
$n7+n38+n69+n100+n101=C$	... pt71;	$n6+n40+n69+n98+n102=C$	... pt72;
$n2+n48+n69+n90+n106=C$	... pt73;	$n1+n50+n69+n88+n107=C$	... pt74;
$n8+n39+n70+n96+n102=C$	... pt81;	$n7+n36+n70+n99+n103=C$	... pt82;
$n3+n49+n70+n86+n107=C$	... pt83;	$n2+n46+n70+n89+n108=C$	... pt84;
$n9+n40+n66+n97+n103=C$	... pt91;	$n8+n37+n66+n100+n104=C$	... pt92;
$n4+n50+n66+n87+n108=C$	... pt93;	$n3+n47+n66+n90+n109=C$	... pt94;
$n10+n36+n67+n98+n104=C$	... pt101;	$n9+n38+n67+n96+n105=C$	... pt102;
$n5+n46+n67+n88+n109=C$	... pt103;	$n4+n48+n67+n86+n110=C$	... pt104;
$n11+n42+n73+n79+n110=C$	... pt111;	$n15+n44+n73+n77+n106=C$	... pt112;
$n6+n27+n73+n94+n115=C$	... pt113;	$n10+n29+n73+n92+n111=C$	... pt114;
$n12+n43+n74+n80+n106=C$	... pt121;	$n11+n45+n74+n78+n107=C$	... pt122;
$n7+n28+n74+n95+n111=C$	... pt123;	$n6+n30+n74+n93+n112=C$	... pt124;

n13+n44+n75+n76+n107=C	...pt131;	n12+n41+n75+n79+n108=C	...pt132;
n8+n29+n75+n91+n112=C	...pt133;	n7+n26+n75+n94+n113=C	...pt134;
n14+n45+n71+n77+n108=C	...pt141;	n13+n42+n71+n80+n109=C	...pt142;
n9+n30+n71+n92+n113=C	...pt143;	n8+n27+n71+n95+n114=C	...pt144;
n15+n41+n72+n78+n109=C	...pt151;	n14+n43+n72+n76+n110=C	...pt152;
n10+n26+n72+n93+n114=C	...pt153;	n9+n28+n72+n91+n115=C	...pt154;
n16+n47+n53+n84+n115=C	...pt161;	n20+n49+n53+n82+n111=C	...pt162;
n11+n32+n53+n99+n120=C	...pt163;	n15+n34+n53+n97+n116=C	...pt164;
n17+n48+n54+n85+n111=C	...pt171;	n16+n50+n54+n83+n112=C	...pt172;
n12+n33+n54+n100+n116=C	...pt173;	n11+n35+n54+n98+n117=C	...pt174;
n18+n49+n55+n81+n112=C	...pt181;	n17+n46+n55+n84+n113=C	...pt182;
n13+n34+n55+n96+n117=C	...pt183;	n12+n31+n55+n99+n118=C	...pt184;
n19+n50+n51+n82+n113=C	...pt191;	n18+n47+n51+n85+n114=C	...pt192;
n14+n35+n51+n97+n118=C	...pt193;	n13+n32+n51+n100+n119=C	...pt194;
n20+n46+n52+n83+n114=C	...pt201;	n19+n48+n52+n81+n115=C	...pt202;
n15+n31+n52+n98+n119=C	...pt203;	n14+n33+n52+n96+n120=C	...pt204;
n21+n27+n58+n89+n120=C	...pt211;	n25+n29+n58+n87+n116=C	...pt212;
n16+n37+n58+n79+n125=C	...pt213;	n20+n39+n58+n77+n121=C	...pt214;
n22+n28+n59+n90+n116=C	...pt221;	n21+n30+n59+n88+n117=C	...pt222;
n17+n38+n59+n80+n121=C	...pt223;	n16+n40+n59+n78+n122=C	...pt224;
n23+n29+n60+n86+n117=C	...pt231;	n22+n26+n60+n89+n118=C	...pt232;
n18+n39+n60+n76+n122=C	...pt233;	n17+n36+n60+n79+n123=C	...pt234;
n24+n30+n56+n87+n118=C	...pt241;	n23+n27+n56+n90+n119=C	...pt242;
n19+n40+n56+n77+n123=C	...pt243;	n18+n37+n56+n80+n124=C	...pt244;
n25+n26+n57+n88+n119=C	...pt251;	n24+n28+n57+n86+n120=C	...pt252;
n20+n36+n57+n78+n124=C	...pt253;	n19+n38+n57+n76+n125=C	...pt254;

Symmetric Self-complementary Pairs has come back this time, and the 'invariable value' for the variable n63 at the geometrical center of our cube has necessarily come back. n63 must always take the constant value 63.

### 3. How to make our 'Latin Cubes' of Order 5 for the "Complete Euler Type":

Let's build the "Complete Euler Cubes" of Order 5 for the Simultaneous type now, using the Positional Number System of Base 5 (5-th Increment Number System).

Before all we have to make 'Latin Cubes' of Order 5, to combine any three of them and assume the set as the Simulated Decomposed Layer Units of our object.

The Latin Cubes are defined in the same way as above, but you must only change the values of Constant Sums as C=10 and SC=4.

On top of that, Latin Cubes must always be made up under the strict rules:

(1) Any Latin Cube must be simply made up of 5 numerical letters: {0, 1, 2, 3 and 4}.

Each letter should appear as often as 25 times, no more and no less.

(2) Each row, each column and each pillar of Latin Cubes must be made up of {0, 1, 2, 3 and 4} using each number strictly once.

Neither duplication nor lack of any number must be found in it.

These strict conditions should make the constant sum of each row, column and pillar of our final object always realized:

$$((0+1+2+3+4) \times 5_{(Dec)} + (0+1+2+3+4)) \times 5_{(Dec)} + (0+1+2+3+4) = 310_{(Dec)}$$

(in Mathematical Expression of Cube)

(3) Each of the Pan-triagonals must also be made up of 5 numbers: {0, 1, 2, 3 and 4}

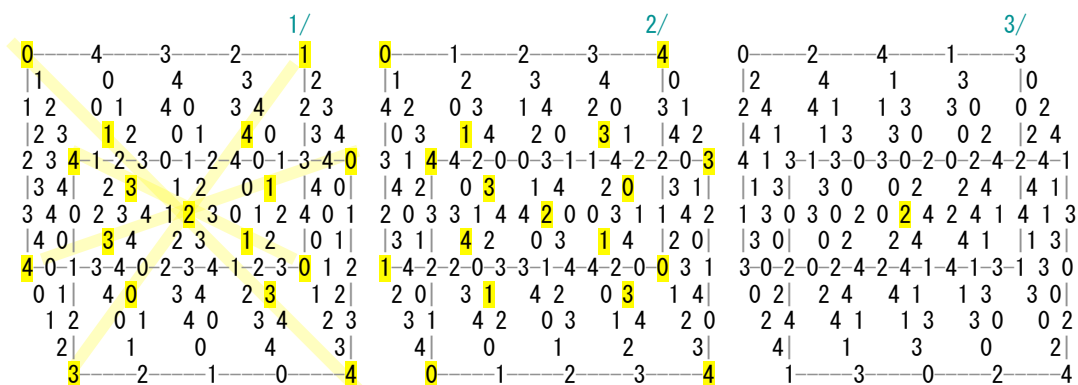
using each one strictly once. Neither duplication nor lack of any number is permitted.

(4) Each of the following three Self-complementary Pairs should appear 25 times in any Latin Unit: (0, 4), (1, 3) and (2, 2)

Constituents of any pair must be placed symmetrically to each other with respect to the geometric center n63 as in (n1, n125), (n2, n124), (n3, n123), (n4, n122), ..., (n60, n66), (n61, n65), (n62, n64). And n63 itself must absolutely take 2.

#### 4. What the 'Latin Cubes' look like:

The following diagrams show some sample solutions of Latin Cubes of Order 5 for the Simultaneous Type, made under the Positional Number System of Base 5.



I want you to watch these samples and check if each of them is successfully made up under those strict conditions above. Will you especially examine each content of rows, columns, pillars and all pan-triagonals, checking if it is made up of {0, 1, 2, 3 and 4} using each one strictly once? Will you also check if each pair of {(0, 4), (1, 3) and (2, 2)} is placed symmetrically with respect to the central n63?

#### 5. How to make the Latin Cubes of Order 5

How can we make such the Latin Cubes as shown above, then? How many of them can we make in all?

I plan to make them here by our 'New Euler's Method' under the Positional Number System of Base 5.

Since we can set two symmetrical variables at the same time in each step, we must have only 62 steps for all variables but n63 consequently. But, we must check if each pair of variables obey the strict Latin Rules in each step, if they successfully make up all their seven lines such as rows, columns, pillars and every four pan-triagonals with {0, 1, 2, 3 and 4} using each number strictly once.

It should be very hard for us to write such a long, complicated program.

I had to use my own new technique, 'Computer Assisted Programming'. I wanted it to make my computer assist me to dictate the actual program routines with rich contents as accurate as possible. I wrote a 'meta-program' for that.

Let me list out only the core part of my actual program here as follows.

```

/**/
short CC;           //Constant Sum of Complementary Pairs: CC=4
short nm[126];
short bc[176][5];  //Flag Table for Basic Conditions: Rw25, Cl25, Pl25 and Pt100
short d[4][129];   //Temporary Data Table for listing out 3 answers
short tlu[33][126]; //Data Table for the 32 Latin Units
short mtc[33][33]; //Reference Table for the Compatibility of every two units
/**/
/** Program Modules for 'Latin Cubes' 5x5x5 of the Simultaneous Type:
    Listed by Kanji Setsuda on May 29, 2010 with MacOSX & Xcode3.1.2 */
/* Level 1: */
/* Set N1 & n125=CC-N1: CC=4 */
void stp01(){
    short a,b;
    for(a=0;a<5;a++){b=CC-a;
        if((bc[1][a]==0)&&(bc[2][a]==0)&&(bc[3][a]==0)&&(bc[76][a]==0)){
            if((bc[81][a]==0)&&(bc[98][a]==0)&&(bc[103][a]==0)){
                if((bc[73][b]==0)&&(bc[74][b]==0)&&(bc[75][b]==0)&&(bc[76][b]==0)){
                    if((bc[93][b]==0)&&(bc[158][b]==0)&&(bc[175][b]==0)){
                        nm[1]=a; nm[125]=b;
                    }
                }
            }
        }
    }
}

```

```

bc[1][a]=1; bc[2][a]=1; bc[3][a]=1; bc[76][a]=1;
bc[81][a]=1; bc[98][a]=1; bc[103][a]=1;
bc[73][b]=1; bc[74][b]=1; bc[75][b]=1; bc[76][b]=1;
bc[93][b]=1; bc[158][b]=1; bc[175][b]=1; cnt2=0;
stp02();
bc[73][b]=0; bc[74][b]=0; bc[75][b]=0; bc[76][b]=0;
bc[93][b]=0; bc[158][b]=0; bc[175][b]=0;
bc[1][a]=0; bc[2][a]=0; bc[3][a]=0; bc[76][a]=0;
bc[81][a]=0; bc[98][a]=0; bc[103][a]=0;}}}}
}
}
/* Set N2 & n124=CC-N2 */
void stp02(){
short a,b;
for(a=0;a<5;a++){b=CC-a;
if((bc[1][a]==0)&&(bc[5][a]==0)&&(bc[6][a]==0)&&(bc[80][a]==0)){
if((bc[85][a]==0)&&(bc[102][a]==0)&&(bc[107][a]==0)){
if((bc[71][b]==0)&&(bc[72][b]==0)&&(bc[73][b]==0)&&(bc[89][b]==0)){
if((bc[92][b]==0)&&(bc[171][b]==0)&&(bc[174][b]==0)){
nm[2]=a; nm[124]=b;
bc[1][a]=1; bc[5][a]=1; bc[6][a]=1; bc[80][a]=1;
bc[85][a]=1; bc[102][a]=1; bc[107][a]=1;
bc[71][b]=1; bc[72][b]=1; bc[73][b]=1; bc[89][b]=1;
bc[92][b]=1; bc[171][b]=1; bc[174][b]=1;
stp03();
bc[71][b]=0; bc[72][b]=0; bc[73][b]=0; bc[89][b]=0;
bc[92][b]=0; bc[171][b]=0; bc[174][b]=0;
bc[1][a]=0; bc[5][a]=0; bc[6][a]=0; bc[80][a]=0;
bc[85][a]=0; bc[102][a]=0; bc[107][a]=0;}}}}
}
}
/* Set N3 & n123=CC-N3 */
void stp03(){
short a,b;
for(a=0;a<5;a++){b=CC-a;
if((bc[1][a]==0)&&(bc[8][a]==0)&&(bc[9][a]==0)&&(bc[84][a]==0)){
if((bc[89][a]==0)&&(bc[106][a]==0)&&(bc[111][a]==0)){
if((bc[68][b]==0)&&(bc[69][b]==0)&&(bc[73][b]==0)&&(bc[85][b]==0)){
if((bc[88][b]==0)&&(bc[167][b]==0)&&(bc[170][b]==0)){
nm[3]=a; nm[123]=b;
bc[1][a]=1; bc[8][a]=1; bc[9][a]=1; bc[84][a]=1;
bc[89][a]=1; bc[106][a]=1; bc[111][a]=1;
bc[68][b]=1; bc[69][b]=1; bc[73][b]=1; bc[85][b]=1;
bc[88][b]=1; bc[167][b]=1; bc[170][b]=1;
stp04();
bc[68][b]=0; bc[69][b]=0; bc[73][b]=0; bc[85][b]=0;
bc[88][b]=0; bc[167][b]=0; bc[170][b]=0;
bc[1][a]=0; bc[8][a]=0; bc[9][a]=0; bc[84][a]=0;
bc[89][a]=0; bc[106][a]=0; bc[111][a]=0;}}}}
}
}
/* Set N4 & n122=CC-N4 */
void stp04(){
short a,b;
for(a=0;a<5;a++){b=CC-a;
if((bc[1][a]==0)&&(bc[11][a]==0)&&(bc[12][a]==0)&&(bc[88][a]==0)){
if((bc[93][a]==0)&&(bc[110][a]==0)&&(bc[115][a]==0)){
if((bc[65][b]==0)&&(bc[66][b]==0)&&(bc[73][b]==0)&&(bc[81][b]==0)){
if((bc[84][b]==0)&&(bc[163][b]==0)&&(bc[166][b]==0)){
nm[4]=a; nm[122]=b;
bc[1][a]=1; bc[11][a]=1; bc[12][a]=1; bc[88][a]=1;
bc[93][a]=1; bc[110][a]=1; bc[115][a]=1;
bc[65][b]=1; bc[66][b]=1; bc[73][b]=1; bc[81][b]=1;
bc[84][b]=1; bc[163][b]=1; bc[166][b]=1;
stp05();
bc[65][b]=0; bc[66][b]=0; bc[73][b]=0; bc[81][b]=0;

```

```

        bc[84][b]=0; bc[163][b]=0; bc[166][b]=0;
        bc[1][a]=0; bc[11][a]=0; bc[12][a]=0; bc[88][a]=0;
        bc[93][a]=0; bc[110][a]=0; bc[115][a]=0;}}}}
    }
}
/* Set N5 & n121=CC-N5 */
void stp05(){
    short a,b;
    for(a=0;a<5;a++){b=CC-a;
        if((bc[1][a]==0)&&(bc[14][a]==0)&&(bc[15][a]==0)&&(bc[77][a]==0)){
            if((bc[92][a]==0)&&(bc[99][a]==0)&&(bc[114][a]==0)){
                if((bc[62][b]==0)&&(bc[63][b]==0)&&(bc[73][b]==0)&&(bc[77][b]==0)){
                    if((bc[80][b]==0)&&(bc[159][b]==0)&&(bc[162][b]==0)){
                        nm[5]=a; nm[121]=b;
                        bc[1][a]=1; bc[14][a]=1; bc[15][a]=1; bc[77][a]=1;
                        bc[92][a]=1; bc[99][a]=1; bc[114][a]=1;
                        bc[62][b]=1; bc[63][b]=1; bc[73][b]=1; bc[77][b]=1;
                        bc[80][b]=1; bc[159][b]=1; bc[162][b]=1;
                        stp06();
                        bc[62][b]=0; bc[63][b]=0; bc[73][b]=0; bc[77][b]=0;
                        bc[80][b]=0; bc[159][b]=0; bc[162][b]=0;
                        bc[1][a]=0; bc[14][a]=0; bc[15][a]=0; bc[77][a]=0;
                        bc[92][a]=0; bc[99][a]=0; bc[114][a]=0;}}}}
                    }
                }
            }
        }
}
/* Set N6 & n120=CC-N6 */
void stp06(){
    short a,b;
    for(a=0;a<5;a++){b=CC-a;
        if((bc[2][a]==0)&&(bc[4][a]==0)&&(bc[18][a]==0)&&(bc[96][a]==0)){
            if((bc[101][a]==0)&&(bc[118][a]==0)&&(bc[123][a]==0)){
                if((bc[60][b]==0)&&(bc[70][b]==0)&&(bc[74][b]==0)&&(bc[138][b]==0)){
                    if((bc[155][b]==0)&&(bc[156][b]==0)&&(bc[173][b]==0)){
                        nm[6]=a; nm[120]=b;
                        bc[2][a]=1; bc[4][a]=1; bc[18][a]=1; bc[96][a]=1;
                        bc[101][a]=1; bc[118][a]=1; bc[123][a]=1;
                        bc[60][b]=1; bc[70][b]=1; bc[74][b]=1; bc[138][b]=1;
                        bc[155][b]=1; bc[156][b]=1; bc[173][b]=1;
                        stp07();
                        bc[60][b]=0; bc[70][b]=0; bc[74][b]=0; bc[138][b]=0;
                        bc[155][b]=0; bc[156][b]=0; bc[173][b]=0;
                        bc[2][a]=0; bc[4][a]=0; bc[18][a]=0; bc[96][a]=0;
                        bc[101][a]=0; bc[118][a]=0; bc[123][a]=0;}}}}
                    }
                }
            }
        }
}
/* Set N7 & n119=CC-N7 */
void stp07(){
    short a,b;
    for(a=0;a<5;a++){b=CC-a;
        if((bc[4][a]==0)&&(bc[5][a]==0)&&(bc[21][a]==0)&&(bc[100][a]==0)){
            if((bc[105][a]==0)&&(bc[122][a]==0)&&(bc[127][a]==0)){
                if((bc[57][b]==0)&&(bc[70][b]==0)&&(bc[71][b]==0)&&(bc[151][b]==0)){
                    if((bc[154][b]==0)&&(bc[169][b]==0)&&(bc[172][b]==0)){
                        nm[7]=a; nm[119]=b;
                        bc[4][a]=1; bc[5][a]=1; bc[21][a]=1; bc[100][a]=1;
                        bc[105][a]=1; bc[122][a]=1; bc[127][a]=1;
                        bc[57][b]=1; bc[70][b]=1; bc[71][b]=1; bc[151][b]=1;
                        bc[154][b]=1; bc[169][b]=1; bc[172][b]=1;
                        stp08();
                        bc[57][b]=0; bc[70][b]=0; bc[71][b]=0; bc[151][b]=0;
                        bc[154][b]=0; bc[169][b]=0; bc[172][b]=0;
                        bc[4][a]=0; bc[5][a]=0; bc[21][a]=0; bc[100][a]=0;
                        bc[105][a]=0; bc[122][a]=0; bc[127][a]=0;}}}}
                    }
                }
            }
        }
}
/* Set N8 & n118=CC-N8 */

```



```

void stp08(){
  short a,b;
  for(a=0;a<5;a++){b=CC-a;
    if((bc[4][a]==0)&&(bc[8][a]==0)&&(bc[24][a]==0)&&(bc[104][a]==0)){
      if((bc[109][a]==0)&&(bc[126][a]==0)&&(bc[131][a]==0)){
        if((bc[54][b]==0)&&(bc[68][b]==0)&&(bc[70][b]==0)&&(bc[147][b]==0)){
          if((bc[150][b]==0)&&(bc[165][b]==0)&&(bc[168][b]==0)){
            nm[8]=a; nm[118]=b;
            bc[4][a]=1; bc[8][a]=1; bc[24][a]=1; bc[104][a]=1;
            bc[109][a]=1; bc[126][a]=1; bc[131][a]=1;
            bc[54][b]=1; bc[68][b]=1; bc[70][b]=1; bc[147][b]=1;
            bc[150][b]=1; bc[165][b]=1; bc[168][b]=1;
            stp09();
            bc[54][b]=0; bc[68][b]=0; bc[70][b]=0; bc[147][b]=0;
            bc[150][b]=0; bc[165][b]=0; bc[168][b]=0;
            bc[4][a]=0; bc[8][a]=0; bc[24][a]=0; bc[104][a]=0;
            bc[109][a]=0; bc[126][a]=0; bc[131][a]=0;}}}}
    }
}
/* Set N9 & n117=CC-N9 */
void stp09(){
  short a,b;
  for(a=0;a<5;a++){b=CC-a;
    if((bc[4][a]==0)&&(bc[11][a]==0)&&(bc[27][a]==0)&&(bc[108][a]==0)){
      if((bc[113][a]==0)&&(bc[130][a]==0)&&(bc[135][a]==0)){
        if((bc[51][b]==0)&&(bc[65][b]==0)&&(bc[70][b]==0)&&(bc[143][b]==0)){
          if((bc[146][b]==0)&&(bc[161][b]==0)&&(bc[164][b]==0)){
            nm[9]=a; nm[117]=b;
            bc[4][a]=1; bc[11][a]=1; bc[27][a]=1; bc[108][a]=1;
            bc[113][a]=1; bc[130][a]=1; bc[135][a]=1;
            bc[51][b]=1; bc[65][b]=1; bc[70][b]=1; bc[143][b]=1;
            bc[146][b]=1; bc[161][b]=1; bc[164][b]=1;
            stp10();
            bc[51][b]=0; bc[65][b]=0; bc[70][b]=0; bc[143][b]=0;
            bc[146][b]=0; bc[161][b]=0; bc[164][b]=0;
            bc[4][a]=0; bc[11][a]=0; bc[27][a]=0; bc[108][a]=0;
            bc[113][a]=0; bc[130][a]=0; bc[135][a]=0;}}}}
    }
}
/* Set N10 & n116=CC-N10 */
void stp10(){
  short a,b;
  for(a=0;a<5;a++){b=CC-a;
    if((bc[4][a]==0)&&(bc[14][a]==0)&&(bc[30][a]==0)&&(bc[97][a]==0)){
      if((bc[112][a]==0)&&(bc[119][a]==0)&&(bc[134][a]==0)){
        if((bc[48][b]==0)&&(bc[62][b]==0)&&(bc[70][b]==0)&&(bc[139][b]==0)){
          if((bc[142][b]==0)&&(bc[157][b]==0)&&(bc[160][b]==0)){
            nm[10]=a; nm[116]=b;
            bc[4][a]=1; bc[14][a]=1; bc[30][a]=1; bc[97][a]=1;
            bc[112][a]=1; bc[119][a]=1; bc[134][a]=1;
            bc[48][b]=1; bc[62][b]=1; bc[70][b]=1; bc[139][b]=1;
            bc[142][b]=1; bc[157][b]=1; bc[160][b]=1;
            stp11();
            bc[48][b]=0; bc[62][b]=0; bc[70][b]=0; bc[139][b]=0;
            bc[142][b]=0; bc[157][b]=0; bc[160][b]=0;
            bc[4][a]=0; bc[14][a]=0; bc[30][a]=0; bc[97][a]=0;
            bc[112][a]=0; bc[119][a]=0; bc[134][a]=0;}}}}
    }
}
/* Set N11 & n115=CC-N11 */
void stp11(){
  short a,b;
  for(a=0;a<5;a++){b=CC-a;
    if((bc[2][a]==0)&&(bc[7][a]==0)&&(bc[33][a]==0)&&(bc[116][a]==0)){
      if((bc[121][a]==0)&&(bc[138][a]==0)&&(bc[143][a]==0)){
        if((bc[45][b]==0)&&(bc[67][b]==0)&&(bc[74][b]==0)&&(bc[118][b]==0)){

```

```

    if((bc[135][b]==0)&&(bc[136][b]==0)&&(bc[153][b]==0)){
        nm[11]=a; nm[115]=b;
        bc[2][a]=1; bc[7][a]=1; bc[33][a]=1; bc[116][a]=1;
        bc[121][a]=1; bc[138][a]=1; bc[143][a]=1;
        bc[45][b]=1; bc[67][b]=1; bc[74][b]=1; bc[118][b]=1;
        bc[135][b]=1; bc[136][b]=1; bc[153][b]=1;
        stp12();
        bc[45][b]=0; bc[67][b]=0; bc[74][b]=0; bc[118][b]=0;
        bc[135][b]=0; bc[136][b]=0; bc[153][b]=0;
        bc[2][a]=0; bc[7][a]=0; bc[33][a]=0; bc[116][a]=0;
        bc[121][a]=0; bc[138][a]=0; bc[143][a]=0;}}}}
    }
}
/* Set N12 & n114=CC-N12 */
void stp12(){
    short a,b;
    for(a=0;a<5;a++){b=CC-a;
        if((bc[5][a]==0)&&(bc[7][a]==0)&&(bc[36][a]==0)&&(bc[120][a]==0)){
            if((bc[125][a]==0)&&(bc[142][a]==0)&&(bc[147][a]==0)){
                if((bc[42][b]==0)&&(bc[67][b]==0)&&(bc[71][b]==0)&&(bc[131][b]==0)){
                    if((bc[134][b]==0)&&(bc[149][b]==0)&&(bc[152][b]==0)){
                        nm[12]=a; nm[114]=b;
                        bc[5][a]=1; bc[7][a]=1; bc[36][a]=1; bc[120][a]=1;
                        bc[125][a]=1; bc[142][a]=1; bc[147][a]=1;
                        bc[42][b]=1; bc[67][b]=1; bc[71][b]=1; bc[131][b]=1;
                        bc[134][b]=1; bc[149][b]=1; bc[152][b]=1;
                        stp13();
                        bc[42][b]=0; bc[67][b]=0; bc[71][b]=0; bc[131][b]=0;
                        bc[134][b]=0; bc[149][b]=0; bc[152][b]=0;
                        bc[5][a]=0; bc[7][a]=0; bc[36][a]=0; bc[120][a]=0;
                        bc[125][a]=0; bc[142][a]=0; bc[147][a]=0;}}}}
                    }
                }
            }
        }
    }
}
/* Set N13 & n113=CC-N13 */
void stp13(){
    short a,b;
    for(a=0;a<5;a++){b=CC-a;
        if((bc[7][a]==0)&&(bc[8][a]==0)&&(bc[39][a]==0)&&(bc[124][a]==0)){
            if((bc[129][a]==0)&&(bc[146][a]==0)&&(bc[151][a]==0)){
                if((bc[39][b]==0)&&(bc[67][b]==0)&&(bc[68][b]==0)&&(bc[127][b]==0)){
                    if((bc[130][b]==0)&&(bc[145][b]==0)&&(bc[148][b]==0)){
                        nm[13]=a; nm[113]=b;
                        bc[7][a]=1; bc[8][a]=1; bc[39][a]=1; bc[124][a]=1;
                        bc[129][a]=1; bc[146][a]=1; bc[151][a]=1;
                        bc[39][b]=1; bc[67][b]=1; bc[68][b]=1; bc[127][b]=1;
                        bc[130][b]=1; bc[145][b]=1; bc[148][b]=1;
                        stp14();
                        bc[39][b]=0; bc[67][b]=0; bc[68][b]=0; bc[127][b]=0;
                        bc[130][b]=0; bc[145][b]=0; bc[148][b]=0;
                        bc[7][a]=0; bc[8][a]=0; bc[39][a]=0; bc[124][a]=0;
                        bc[129][a]=0; bc[146][a]=0; bc[151][a]=0;}}}}
                    }
                }
            }
        }
    }
}
/* Set N14 & n112=CC-N14 */
void stp14(){
    short a,b;
    for(a=0;a<5;a++){b=CC-a;
        if((bc[7][a]==0)&&(bc[11][a]==0)&&(bc[42][a]==0)&&(bc[128][a]==0)){
            if((bc[133][a]==0)&&(bc[150][a]==0)&&(bc[155][a]==0)){
                if((bc[36][b]==0)&&(bc[65][b]==0)&&(bc[67][b]==0)&&(bc[123][b]==0)){
                    if((bc[126][b]==0)&&(bc[141][b]==0)&&(bc[144][b]==0)){
                        nm[14]=a; nm[112]=b;
                        bc[7][a]=1; bc[11][a]=1; bc[42][a]=1; bc[128][a]=1;
                        bc[133][a]=1; bc[150][a]=1; bc[155][a]=1;
                        bc[36][b]=1; bc[65][b]=1; bc[67][b]=1; bc[123][b]=1;
                        bc[126][b]=1; bc[141][b]=1; bc[144][b]=1;
                    }
                }
            }
        }
    }
}

```



```

    stp15();
    bc[36][b]=0; bc[65][b]=0; bc[67][b]=0; bc[123][b]=0;
    bc[126][b]=0; bc[141][b]=0; bc[144][b]=0;
    bc[7][a]=0; bc[11][a]=0; bc[42][a]=0; bc[128][a]=0;
    bc[133][a]=0; bc[150][a]=0; bc[155][a]=0;}}}}
}
}
/* Set N15 & n111=CC-N15 */
void stp15(){
    short a,b;
    for(a=0;a<5;a++){b=CC-a;
        if((bc[7][a]==0)&&(bc[14][a]==0)&&(bc[45][a]==0)&&(bc[117][a]==0)){
            if((bc[132][a]==0)&&(bc[139][a]==0)&&(bc[154][a]==0)){
                if((bc[33][b]==0)&&(bc[62][b]==0)&&(bc[67][b]==0)&&(bc[119][b]==0)){
                    if((bc[122][b]==0)&&(bc[137][b]==0)&&(bc[140][b]==0)){
                        nm[15]=a; nm[111]=b;
                        bc[7][a]=1; bc[14][a]=1; bc[45][a]=1; bc[117][a]=1;
                        bc[132][a]=1; bc[139][a]=1; bc[154][a]=1;
                        bc[33][b]=1; bc[62][b]=1; bc[67][b]=1; bc[119][b]=1;
                        bc[122][b]=1; bc[137][b]=1; bc[140][b]=1;
                        stp16();
                        bc[33][b]=0; bc[62][b]=0; bc[67][b]=0; bc[119][b]=0;
                        bc[122][b]=0; bc[137][b]=0; bc[140][b]=0;
                        bc[7][a]=0; bc[14][a]=0; bc[45][a]=0; bc[117][a]=0;
                        bc[132][a]=0; bc[139][a]=0; bc[154][a]=0;}}}}
                    }
                }
            }
        }
    }
}
/* Set N16 & n110=CC-N16 */
void stp16(){
    short a,b;
    for(a=0;a<5;a++){b=CC-a;
        if((bc[2][a]==0)&&(bc[10][a]==0)&&(bc[48][a]==0)&&(bc[136][a]==0)){
            if((bc[141][a]==0)&&(bc[158][a]==0)&&(bc[163][a]==0)){
                if((bc[30][b]==0)&&(bc[64][b]==0)&&(bc[74][b]==0)&&(bc[98][b]==0)){
                    if((bc[115][b]==0)&&(bc[116][b]==0)&&(bc[133][b]==0)){
                        nm[16]=a; nm[110]=b;
                        bc[2][a]=1; bc[10][a]=1; bc[48][a]=1; bc[136][a]=1;
                        bc[141][a]=1; bc[158][a]=1; bc[163][a]=1;
                        bc[30][b]=1; bc[64][b]=1; bc[74][b]=1; bc[98][b]=1;
                        bc[115][b]=1; bc[116][b]=1; bc[133][b]=1;
                        stp17();
                        bc[30][b]=0; bc[64][b]=0; bc[74][b]=0; bc[98][b]=0;
                        bc[115][b]=0; bc[116][b]=0; bc[133][b]=0;
                        bc[2][a]=0; bc[10][a]=0; bc[48][a]=0; bc[136][a]=0;
                        bc[141][a]=0; bc[158][a]=0; bc[163][a]=0;}}}}
                    }
                }
            }
        }
    }
}
/* Set N17 & n109=CC-N17 */
void stp17(){
    short a,b;
    for(a=0;a<5;a++){b=CC-a;
        if((bc[5][a]==0)&&(bc[10][a]==0)&&(bc[51][a]==0)&&(bc[140][a]==0)){
            if((bc[145][a]==0)&&(bc[162][a]==0)&&(bc[167][a]==0)){
                if((bc[27][b]==0)&&(bc[64][b]==0)&&(bc[71][b]==0)&&(bc[111][b]==0)){
                    if((bc[114][b]==0)&&(bc[129][b]==0)&&(bc[132][b]==0)){
                        nm[17]=a; nm[109]=b;
                        bc[5][a]=1; bc[10][a]=1; bc[51][a]=1; bc[140][a]=1;
                        bc[145][a]=1; bc[162][a]=1; bc[167][a]=1;
                        bc[27][b]=1; bc[64][b]=1; bc[71][b]=1; bc[111][b]=1;
                        bc[114][b]=1; bc[129][b]=1; bc[132][b]=1;
                        stp18();
                        bc[27][b]=0; bc[64][b]=0; bc[71][b]=0; bc[111][b]=0;
                        bc[114][b]=0; bc[129][b]=0; bc[132][b]=0;
                        bc[5][a]=0; bc[10][a]=0; bc[51][a]=0; bc[140][a]=0;
                        bc[145][a]=0; bc[162][a]=0; bc[167][a]=0;}}}}
                    }
                }
            }
        }
    }
}

```

```

}
/* Set N18 & n108=CC-N18 */
void stp18(){
  short a,b;
  for(a=0;a<5;a++){b=CC-a;
    if((bc[8][a]==0)&&(bc[10][a]==0)&&(bc[54][a]==0)&&(bc[144][a]==0)){
      if((bc[149][a]==0)&&(bc[166][a]==0)&&(bc[171][a]==0)){
        if((bc[24][b]==0)&&(bc[64][b]==0)&&(bc[68][b]==0)&&(bc[107][b]==0)){
          if((bc[110][b]==0)&&(bc[125][b]==0)&&(bc[128][b]==0)){
            nm[18]=a; nm[108]=b;
            bc[8][a]=1; bc[10][a]=1; bc[54][a]=1; bc[144][a]=1;
            bc[149][a]=1; bc[166][a]=1; bc[171][a]=1;
            bc[24][b]=1; bc[64][b]=1; bc[68][b]=1; bc[107][b]=1;
            bc[110][b]=1; bc[125][b]=1; bc[128][b]=1;
            stp19();
            bc[24][b]=0; bc[64][b]=0; bc[68][b]=0; bc[107][b]=0;
            bc[110][b]=0; bc[125][b]=0; bc[128][b]=0;
            bc[8][a]=0; bc[10][a]=0; bc[54][a]=0; bc[144][a]=0;
            bc[149][a]=0; bc[166][a]=0; bc[171][a]=0;}}}}
  }
}
/* Set N19 & n107=CC-N19 */
void stp19(){
  short a,b;
  for(a=0;a<5;a++){b=CC-a;
    if((bc[10][a]==0)&&(bc[11][a]==0)&&(bc[57][a]==0)&&(bc[148][a]==0)){
      if((bc[153][a]==0)&&(bc[170][a]==0)&&(bc[175][a]==0)){
        if((bc[21][b]==0)&&(bc[64][b]==0)&&(bc[65][b]==0)&&(bc[103][b]==0)){
          if((bc[106][b]==0)&&(bc[121][b]==0)&&(bc[124][b]==0)){
            nm[19]=a; nm[107]=b;
            bc[10][a]=1; bc[11][a]=1; bc[57][a]=1; bc[148][a]=1;
            bc[153][a]=1; bc[170][a]=1; bc[175][a]=1;
            bc[21][b]=1; bc[64][b]=1; bc[65][b]=1; bc[103][b]=1;
            bc[106][b]=1; bc[121][b]=1; bc[124][b]=1;
            stp20();
            bc[21][b]=0; bc[64][b]=0; bc[65][b]=0; bc[103][b]=0;
            bc[106][b]=0; bc[121][b]=0; bc[124][b]=0;
            bc[10][a]=0; bc[11][a]=0; bc[57][a]=0; bc[148][a]=0;
            bc[153][a]=0; bc[170][a]=0; bc[175][a]=0;}}}}
  }
}
/* Set N20 & n106=CC-N20 */
void stp20(){
  short a,b;
  for(a=0;a<5;a++){b=CC-a;
    if((bc[10][a]==0)&&(bc[14][a]==0)&&(bc[60][a]==0)&&(bc[137][a]==0)){
      if((bc[152][a]==0)&&(bc[159][a]==0)&&(bc[174][a]==0)){
        if((bc[18][b]==0)&&(bc[62][b]==0)&&(bc[64][b]==0)&&(bc[99][b]==0)){
          if((bc[102][b]==0)&&(bc[117][b]==0)&&(bc[120][b]==0)){
            nm[20]=a; nm[106]=b;
            bc[10][a]=1; bc[14][a]=1; bc[60][a]=1; bc[137][a]=1;
            bc[152][a]=1; bc[159][a]=1; bc[174][a]=1;
            bc[18][b]=1; bc[62][b]=1; bc[64][b]=1; bc[99][b]=1;
            bc[102][b]=1; bc[117][b]=1; bc[120][b]=1;
            stp21();
            bc[18][b]=0; bc[62][b]=0; bc[64][b]=0; bc[99][b]=0;
            bc[102][b]=0; bc[117][b]=0; bc[120][b]=0;
            bc[10][a]=0; bc[14][a]=0; bc[60][a]=0; bc[137][a]=0;
            bc[152][a]=0; bc[159][a]=0; bc[174][a]=0;}}}}
  }
}
/* Set N21 & n105=CC-N21 */
void stp21(){
  short a,b;
  for(a=0;a<5;a++){b=CC-a;
    if((bc[2][a]==0)&&(bc[13][a]==0)&&(bc[63][a]==0)&&(bc[78][a]==0)){

```

```

if((bc[83][a]==0)&&(bc[156][a]==0)&&(bc[161][a]==0)){
if((bc[15][b]==0)&&(bc[61][b]==0)&&(bc[74][b]==0)&&(bc[78][b]==0)){
if((bc[95][b]==0)&&(bc[96][b]==0)&&(bc[113][b]==0)){
nm[21]=a; nm[105]=b;
bc[2][a]=1; bc[13][a]=1; bc[63][a]=1; bc[78][a]=1;
bc[83][a]=1; bc[156][a]=1; bc[161][a]=1;
bc[15][b]=1; bc[61][b]=1; bc[74][b]=1; bc[78][b]=1;
bc[95][b]=1; bc[96][b]=1; bc[113][b]=1;
stp22();
bc[15][b]=0; bc[61][b]=0; bc[74][b]=0; bc[78][b]=0;
bc[95][b]=0; bc[96][b]=0; bc[113][b]=0;
bc[2][a]=0; bc[13][a]=0; bc[63][a]=0; bc[78][a]=0;
bc[83][a]=0; bc[156][a]=0; bc[161][a]=0;}}}}
}
}
/* Set N22 & n104=CC-N22 */
void stp22(){
short a,b;
for(a=0;a<5;a++){b=CC-a;
if((bc[5][a]==0)&&(bc[13][a]==0)&&(bc[66][a]==0)&&(bc[82][a]==0)){
if((bc[87][a]==0)&&(bc[160][a]==0)&&(bc[165][a]==0)){
if((bc[12][b]==0)&&(bc[61][b]==0)&&(bc[71][b]==0)&&(bc[91][b]==0)){
if((bc[94][b]==0)&&(bc[109][b]==0)&&(bc[112][b]==0)){
nm[22]=a; nm[104]=b;
bc[5][a]=1; bc[13][a]=1; bc[66][a]=1; bc[82][a]=1;
bc[87][a]=1; bc[160][a]=1; bc[165][a]=1;
bc[12][b]=1; bc[61][b]=1; bc[71][b]=1; bc[91][b]=1;
bc[94][b]=1; bc[109][b]=1; bc[112][b]=1;
stp23();
bc[12][b]=0; bc[61][b]=0; bc[71][b]=0; bc[91][b]=0;
bc[94][b]=0; bc[109][b]=0; bc[112][b]=0;
bc[5][a]=0; bc[13][a]=0; bc[66][a]=0; bc[82][a]=0;
bc[87][a]=0; bc[160][a]=0; bc[165][a]=0;}}}}}}
}
}
/* Set N23 & n103=CC-N23 */
void stp23(){
short a,b;
for(a=0;a<5;a++){b=CC-a;
if((bc[8][a]==0)&&(bc[13][a]==0)&&(bc[69][a]==0)&&(bc[86][a]==0)){
if((bc[91][a]==0)&&(bc[164][a]==0)&&(bc[169][a]==0)){
if((bc[9][b]==0)&&(bc[61][b]==0)&&(bc[68][b]==0)&&(bc[87][b]==0)){
if((bc[90][b]==0)&&(bc[105][b]==0)&&(bc[108][b]==0)){
nm[23]=a; nm[103]=b;
bc[8][a]=1; bc[13][a]=1; bc[69][a]=1; bc[86][a]=1;
bc[91][a]=1; bc[164][a]=1; bc[169][a]=1;
bc[9][b]=1; bc[61][b]=1; bc[68][b]=1; bc[87][b]=1;
bc[90][b]=1; bc[105][b]=1; bc[108][b]=1;
stp24();
bc[9][b]=0; bc[61][b]=0; bc[68][b]=0; bc[87][b]=0;
bc[90][b]=0; bc[105][b]=0; bc[108][b]=0;
bc[8][a]=0; bc[13][a]=0; bc[69][a]=0; bc[86][a]=0;
bc[91][a]=0; bc[164][a]=0; bc[169][a]=0;}}}}}}
}
}
/* Set N24 & n102=CC-N24 */
void stp24(){
short a,b;
for(a=0;a<5;a++){b=CC-a;
if((bc[11][a]==0)&&(bc[13][a]==0)&&(bc[72][a]==0)&&(bc[90][a]==0)){
if((bc[95][a]==0)&&(bc[168][a]==0)&&(bc[173][a]==0)){
if((bc[6][b]==0)&&(bc[61][b]==0)&&(bc[65][b]==0)&&(bc[83][b]==0)){
if((bc[86][b]==0)&&(bc[101][b]==0)&&(bc[104][b]==0)){
nm[24]=a; nm[102]=b;
bc[11][a]=1; bc[13][a]=1; bc[72][a]=1; bc[90][a]=1;
bc[95][a]=1; bc[168][a]=1; bc[173][a]=1;

```

```

bc[6][b]=1; bc[61][b]=1; bc[65][b]=1; bc[83][b]=1;
bc[86][b]=1; bc[101][b]=1; bc[104][b]=1;
stp25();
bc[6][b]=0; bc[61][b]=0; bc[65][b]=0; bc[83][b]=0;
bc[86][b]=0; bc[101][b]=0; bc[104][b]=0;
bc[11][a]=0; bc[13][a]=0; bc[72][a]=0; bc[90][a]=0;
bc[95][a]=0; bc[168][a]=0; bc[173][a]=0;}}}}
}
}
/* Set N25 & n101=CC-N25 */
void stp25(){
short a,b;
for(a=0;a<5;a++){b=CC-a;
if((bc[13][a]==0)&&(bc[14][a]==0)&&(bc[75][a]==0)&&(bc[79][a]==0)){
if((bc[94][a]==0)&&(bc[157][a]==0)&&(bc[172][a]==0)){
if((bc[3][b]==0)&&(bc[61][b]==0)&&(bc[62][b]==0)&&(bc[79][b]==0)){
if((bc[82][b]==0)&&(bc[97][b]==0)&&(bc[100][b]==0)){
nm[25]=a; nm[101]=b;
bc[13][a]=1; bc[14][a]=1; bc[75][a]=1; bc[79][a]=1;
bc[94][a]=1; bc[157][a]=1; bc[172][a]=1;
bc[3][b]=1; bc[61][b]=1; bc[62][b]=1; bc[79][b]=1;
bc[82][b]=1; bc[97][b]=1; bc[100][b]=1;
stp26();
bc[3][b]=0; bc[61][b]=0; bc[62][b]=0; bc[79][b]=0;
bc[82][b]=0; bc[97][b]=0; bc[100][b]=0;
bc[13][a]=0; bc[14][a]=0; bc[75][a]=0; bc[79][a]=0;
bc[94][a]=0; bc[157][a]=0; bc[172][a]=0;}}}}}}
}
}
/* Level 2: */
/* Set N26 & n100=CC-N26 */
void stp26(){
short a,b;
for(a=0;a<5;a++){b=CC-a;
if((bc[3][a]==0)&&(bc[16][a]==0)&&(bc[17][a]==0)&&(bc[127][a]==0)){
if((bc[134][a]==0)&&(bc[165][a]==0)&&(bc[172][a]==0)){
if((bc[58][b]==0)&&(bc[59][b]==0)&&(bc[75][b]==0)&&(bc[100][b]==0)){
if((bc[109][b]==0)&&(bc[142][b]==0)&&(bc[151][b]==0)){
nm[26]=a; nm[100]=b;
bc[3][a]=1; bc[16][a]=1; bc[17][a]=1; bc[127][a]=1;
bc[134][a]=1; bc[165][a]=1; bc[172][a]=1;
bc[58][b]=1; bc[59][b]=1; bc[75][b]=1; bc[100][b]=1;
bc[109][b]=1; bc[142][b]=1; bc[151][b]=1;
stp27();
bc[58][b]=0; bc[59][b]=0; bc[75][b]=0; bc[100][b]=0;
bc[109][b]=0; bc[142][b]=0; bc[151][b]=0;
bc[3][a]=0; bc[16][a]=0; bc[17][a]=0; bc[127][a]=0;
bc[134][a]=0; bc[165][a]=0; bc[172][a]=0;}}}}}}
}
}
/* ...(skip)... */
/* Set N62 & n64=CC-N62 */
void stp62(){
short a,b;
for(a=0;a<5;a++){b=CC-a;
if((bc[35][a]==0)&&(bc[36][a]==0)&&(bc[37][a]==0)&&(bc[92][a]==0)){
if((bc[93][a]==0)&&(bc[94][a]==0)&&(bc[95][a]==0)){
if((bc[37][b]==0)&&(bc[41][b]==0)&&(bc[42][b]==0)&&(bc[80][b]==0)){
if((bc[81][b]==0)&&(bc[82][b]==0)&&(bc[83][b]==0)){
nm[62]=a; nm[64]=b;
bc[35][a]=1; bc[36][a]=1; bc[37][a]=1; bc[92][a]=1;
bc[93][a]=1; bc[94][a]=1; bc[95][a]=1;
bc[37][b]=1; bc[41][b]=1; bc[42][b]=1; bc[80][b]=1;
bc[81][b]=1; bc[82][b]=1; bc[83][b]=1;
recordans();
bc[37][b]=0; bc[41][b]=0; bc[42][b]=0; bc[80][b]=0;

```

```

        bc[81][b]=0; bc[82][b]=0; bc[83][b]=0;
        bc[35][a]=0; bc[36][a]=0; bc[37][a]=0; bc[92][a]=0;
        bc[93][a]=0; bc[94][a]=0; bc[95][a]=0;}}}}
    }
}
/**/
/* Record the Latin Units */
void recordans(){
    short n;
    tlu[cnt][0]=cnt+1;
    for(n=1;n<126;n++){tlu[cnt][n]=nm[n];}
    cnt++;
    d[cnt3][0]=cnt; d[cnt3+1][0]=0;
    for(n=1;n<126;n++){d[cnt3][n]=nm[n];}
    cnt3++; if(cnt3==3){prdlu(cnt3); cnt3=0;}
}
/**/
/* Make the Reference Table */
void mkreftbl(){
    short m,l,n,t,d;
/* Measure How Similar each Pair is. */
    for(m=0;m<cnt;m++){
        for(l=0;l<cnt;l++){
            t=0;
            for(n=1;n<126;n++){if(tlu[m][n]==tlu[l][n]){t++;}}
            mtc[m][l]=t;
            t=0;
            for(n=1;n<126;n++){if(tlu[m][n]+tlu[l][n]==4){t++;}}
            if(mtc[m][l]<t){mtc[m][l]=t;}
        }
    }
}
/**/
/* ..... */

```

In every step we always watch each pair of variables and check if they are 'Used' or 'Un-used' on the 14 lines: 2 rows, 2 columns, 2 pillars and 8 pan-triagonals.

If they are 'Used', then we must have another value pair to be checked again. But, if they are all 'Un-used' in the 14 lines, then we can take them for the pair of variables and go to the next step, after we set the 14 flags for the values as 'Used'.

We have to repeat the same kind of examination 62 times in all.

## 6. How many Latin Units we could have in all:

The compact list below shows the recent result of my research.

The total count of Latin Units is 32.

What a small count it is! But it means we have to check every combination of three Latin Units in over 32 thousand ways (=32 x 32 x 32)! Why!

**\*\* 'Latin Cubes' 5x5x5 for the Simultaneous Type: Self-complimentary and Pan-triagonal \*\***  
**\*\* Complete List made by Kanji Setsuda on June 7, 2010 with MacOSX & Xcode3.1.2 \*\***

1/					2/					3/				
0	4	3	2	1	0	1	2	3	4	0	2	4	1	3
1	0	4	3	2	1	2	3	4	0	2	4	1	3	0
1	2	0	1	4	4	2	0	3	1	2	4	4	1	3
2	3	1	2	0	0	3	1	4	2	4	1	1	3	0
2	3	4	1	2	3	1	4	2	0	4	1	3	1	3
0	1	2	3	0	1	4	2	0	3	0	2	0	2	4
1	3	4	0	1	2	0	3	1	4	2	4	2	4	1
3	4	2	3	1	4	2	0	3	1	1	3	3	0	2
3	4	0	2	3	2	0	3	1	4	1	3	0	2	4
0	1	2	3	0	3	1	4	2	0	2	4	2	4	1
4	0	3	4	2	3	1	4	2	0	3	0	0	2	4
4	0	1	3	4	1	4	2	0	3	3	0	2	4	1
0	1	2	3	0	1	4	2	0	3	3	0	2	4	1
1	2	0	1	4	2	0	3	1	4	0	2	2	4	1
4	0	3	4	2	3	1	4	2	0	2	4	4	1	3
0	1	2	3	0	1	4	2	0	3	0	2	2	4	1
1	2	0	1	4	2	0	3	1	4	2	4	4	1	3
4	0	3	4	2	3	1	4	2	0	0	2	2	4	1
3	2	1	0	4	0	1	2	3	4	1	3	0	2	4

4/  
 0—2—4—3—1  
 |2 4 3 1 0|  
 2 4 4 3 3 1 1 0 0 2  
 |4 3 3 1 1 0 0 2| 2 4  
 4 3 1-3-1-0-1-0-2-0-2-4-2-4-3  
 |3 1| 1 0 0 2 2 4 |4 3|  
 3 1 0 1 0 2 0 2 4 2 4 3 4 3 1  
 |1 0| 0 2 2 4 4 3 |3 1|  
 1-0-2-0-2-4-2-4-3-4-3-1-3-1-0  
 0 2| 2 4 4 3 3 1 1 0|  
 2 4 4 3 3 1 1 0 0 2  
 4| 3 1 0 2|  
 3—1—0—2—4

5/  
 0—4—1—2—3  
 |3 0 4 1 2|  
 3 2 0 3 4 0 1 4 2 1  
 |2 1 3 2 0 3 4 0| 1 4  
 2 1 4-3-2-1-0-3-2-4-0-3-1-4-0  
 |1 4| 2 1 3 2 0 3 |4 0|  
 1 4 0 2 1 4 3 2 1 0 3 2 4 0 3  
 |4 0| 1 4 2 1 3 2 |0 3|  
 4-0-3-1-4-0-2-1-4-3-2-1-0-3-2  
 0 3| 4 0 1 4 2 1 3 2|  
 3 2 0 3 4 0 1 4 2 1  
 2| 3 0 4 1|  
 1—2—3—0—4

6/  
 0—3—2—1—4  
 |3 2 1 4 0|  
 4 2 0 1 3 4 2 0 1 3  
 |0 1 3 4 2 0 1 3| 4 2  
 1 3 4-4-2-0-0-1-3-3-4-2-2-0-1  
 |4 2| 0 1 3 4 2 0 |1 3|  
 2 0 1 1 3 4 4 2 0 0 1 3 3 4 2  
 |1 3| 4 2 0 1 3 4 |2 0|  
 3-4-2-2-0-1-1-3-4-4-2-0-0-1-3  
 2 0| 1 3 4 2 0 1 3 4|  
 1 3 4 2 0 1 3 4 2 0  
 4| 0 3 2 1|  
 0—3—2—1—4

7/  
 0—3—2—1—4  
 |4 0 3 2 1|  
 3 1 2 4 1 0 4 3 0 2  
 |0 2 3 1 2 4 1 0| 4 3  
 2 4 3-1-0-2-4-3-1-0-2-4-3-1-0  
 |3 1| 2 4 1 0 4 3 |0 2|  
 1 0 2 4 3 1 0 2 4 3 1 0 2 4 3  
 |2 4| 1 0 4 3 0 2 |3 1|  
 4-3-1-0-2-4-3-1-0-2-4-3-1-0-2  
 1 0| 4 3 0 2 3 1 2 4|  
 2 4 1 0 4 3 0 2 3 1  
 3| 2 1 4 0|  
 0—3—2—1—4

8/  
 0—1—2—3—4  
 |4 0 1 2 3|  
 1 3 2 4 3 0 4 1 0 2  
 |0 2 1 3 2 4 3 0| 4 1  
 2 4 1-3-0-2-4-1-3-0-2-4-1-3-0  
 |1 3| 2 4 3 0 4 1 |0 2|  
 3 0 2 4 1 3 0 2 4 1 3 0 2 4 1  
 |2 4| 3 0 4 1 0 2 |1 3|  
 4-1-3-0-2-4-1-3-0-2-4-1-3-0-2  
 3 0| 4 1 0 2 1 3 2 4|  
 2 4 3 0 4 1 0 2 1 3  
 1| 2 3 4 0|  
 0—1—2—3—4

9/  
 1—3—4—2—0  
 |0 1 3 4 2|  
 0 2 1 0 3 1 4 3 2 4  
 |2 4 0 2 1 0 3 1| 4 3  
 2 4 3-0-2-4-1-0-2-3-1-0-4-3-1  
 |4 3| 2 4 0 2 1 0 |3 1|  
 4 3 1 2 4 3 0 2 4 1 0 2 3 1 0  
 |3 1| 4 3 2 4 0 2 |1 0|  
 3-1-0-4-3-1-2-4-3-0-2-4-1-0-2  
 1 0| 3 1 4 3 2 4 0 2|  
 0 2 1 0 3 1 4 3 2 4  
 2| 0 1 3 4|  
 4—2—0—1—3

10/  
 1—0—2—4—3  
 |0 2 4 3 1|  
 3 2 1 4 0 3 2 1 4 0  
 |1 4 0 3 2 1 4 0| 3 2  
 4 0 3-3-2-1-1-4-0-0-3-2-2-1-4  
 |3 2| 1 4 0 3 2 1 |4 0|  
 2 1 4 4 0 3 3 2 1 1 4 0 0 3 2  
 |4 0| 3 2 1 4 0 3 |2 1|  
 0-3-2-2-1-4-4-0-3-3-2-1-1-4-0  
 2 1| 4 0 3 2 1 4 0 3|  
 4 0 3 2 1 4 0 3 2 1  
 3| 1 0 2 4|  
 1—0—2—4—3

11/  
 1—2—3—0—4  
 |2 3 0 4 1 2 3|  
 2 3 3 0 0 4 4 1 1 2  
 |3 0 0 4 4 1 1 2| 2 3  
 3 0 4-0-4-1-4-1-2-1-2-3-2-3-0  
 |0 4| 4 1 1 2 2 3 |3 0|  
 0 4 1 4 1 2 1 2 3 2 3 0 3 0 4  
 |4 1| 1 2 2 3 3 0 |0 4|  
 4-1-2-1-2-3-2-3-0-3-0-4-0-4-1  
 1 2| 2 3 3 0 0 4 4 1|  
 2 3 3 0 0 4 4 1 1 2  
 3| 0 4 1 2|  
 0—4—1—2—3

12/  
 1—2—3—4—0  
 |2 3 4 0 1 2 3|  
 2 3 3 4 4 0 0 1 1 2  
 |3 4 4 0 0 1 1 2| 2 3  
 3 4 0-4-0-1-0-1-2-1-2-3-2-3-4  
 |4 0| 0 1 1 2 2 3 |3 4|  
 4 0 1 0 1 2 1 2 3 2 3 4 3 4 0  
 |0 1| 1 2 2 3 3 4 |4 0|  
 0-1-2-1-2-3-2-3-4-3-4-0-4-0-1  
 1 2| 2 3 3 4 4 0 0 1|  
 2 3 3 4 4 0 0 1 1 2  
 3| 4 0 1 2|  
 4—0—1—2—3

13/  
 1—4—2—0—3  
 |3 1 4 2 0|  
 4 0 2 3 0 1 3 4 1 2  
 |1 2 4 0 2 3 0 1| 3 4  
 2 3 4-0-1-2-3-4-0-1-2-3-4-0-1  
 |4 0| 2 3 0 1 3 4 |1 2|  
 0 1 2 3 4 0 1 2 3 4 0 1 2 3 4  
 |2 3| 0 1 3 4 1 2 |4 0|  
 3-4-0-1-2-3-4-0-1-2-3-4-0-1-2  
 0 1| 3 4 1 2 4 0 2 3|  
 2 3 0 1 3 4 1 2 4 0  
 4| 2 0 3 1|  
 1—4—2—0—3

14/  
 1—0—2—4—3  
 |3 1 0 2 4|  
 0 4 2 3 4 1 3 0 1 2  
 |1 2 0 4 2 3 4 1| 3 0  
 2 3 0-4-1-2-3-0-4-1-2-3-0-4-1  
 |0 4| 2 3 4 1 3 0 |1 2|  
 4 1 2 3 0 4 1 2 3 0 4 1 2 3 0  
 |2 3| 4 1 3 0 1 2 |0 4|  
 3-0-4-1-2-3-0-4-1-2-3-0-4-1-2  
 4 1| 3 0 1 2 0 4 2 3|  
 2 3 4 1 3 0 1 2 0 4  
 0| 2 4 3 1|  
 1—0—2—4—3

15/  
 1—4—2—0—3  
 |4 2 0 3 1|  
 3 2 1 0 4 3 2 1 0 4  
 |1 0 4 3 2 1 0 4| 3 2  
 0 4 3-3-2-1-1-0-4-4-3-2-2-1-0  
 |3 2| 1 0 4 3 2 1 |0 4|  
 2 1 0 0 4 3 3 2 1 1 0 4 4 3 2  
 |0 4| 3 2 1 0 4 3 |2 1|  
 4-3-2-2-1-0-0-4-3-3-2-1-1-0-4  
 2 1| 0 4 3 2 1 0 4 3|  
 0 4 3 2 1 0 4 3 2 1  
 3| 1 4 2 0|  
 1—4—2—0—3

16/  
 1—3—0—2—4  
 |4 1 3 0 2|  
 4 2 1 4 3 1 0 3 2 0  
 |2 0 4 2 1 4 3 1| 0 3  
 2 0 3-4-2-0-1-4-2-3-1-4-0-3-1  
 |0 3| 2 0 4 2 1 4 |3 1|  
 0 3 1 2 0 3 4 2 0 1 4 2 3 1 4  
 |3 1| 0 3 2 0 4 2 |1 4|  
 3-1-4-0-3-1-2-0-3-4-2-0-1-4-2  
 1 4| 3 1 0 3 2 0 4 2|  
 4 2 1 4 3 1 0 3 2 0  
 2| 4 1 3 0|  
 0—2—4—1—3

17/  
 3—1—4—2—0  
 |0 3 1 4 2|  
 0 2 3 0 1 3 4 1 2 4  
 |2 4 0 2 3 0 1 3| 4 1  
 2 4 1-0-2-4-3-0-2-1-3-0-4-1-3  
 |4 1| 2 4 0 2 3 0 |1 3|  
 4 1 3 2 4 1 0 2 4 3 0 2 1 3 0  
 |1 3| 4 1 2 4 0 2 |3 0|  
 1-3-0-4-1-3-2-4-1-0-2-4-3-0-2  
 3 0| 1 3 4 1 2 4 0 2|  
 0 2 3 0 1 3 4 1 2 4  
 2| 0 3 1 4|  
 4—2—0—3—1

18/  
 3—0—2—4—1  
 |0 2 4 1 3|  
 1 2 3 4 0 1 2 3 4 0  
 |3 4 0 1 2 3 4 0| 1 2  
 4 0 1-1-2-3-3-4-0-0-1-2-2-3-4  
 |1 2| 3 4 0 1 2 3 |4 0|  
 2 3 4 4 0 1 1 2 3 3 4 0 0 1 2  
 |4 0| 1 2 3 4 0 1 |2 3|  
 0-1-2-2-3-4-4-0-1-1-2-3-3-4-0  
 2 3| 4 0 1 2 3 4 0 1|  
 4 0 1 2 3 4 0 1 2 3  
 1| 3 0 2 4|  
 3—0—2—4—1

19/  
 3—4—2—0—1  
 |1 3 4 2 |0  
 4 0 2 1 0 3 1 4 3 2  
 |3 2 4 0 2 1 0 3 |1 4  
 2 1 4 0 3 2 1 4 0 3 2 1 4 0 3  
 |4 0 | 2 1 0 3 1 4 |3 2|  
 0 3 2 1 4 0 3 2 1 4 0 3 2 1 4  
 |2 1 | 0 3 1 4 3 2 |4 0|  
 1-4-0-3-2-1-4-0-3-2-1-4-0-3-2  
 0 3 | 1 4 3 2 4 0 2 1|  
 2 1 0 3 1 4 3 2 4 0  
 4 | 2 0 1 3 |  
 3—4—2—0—1

20/  
 3—0—2—4—1  
 |1 3 0 2 |4  
 0 4 2 1 4 3 1 0 3 2  
 |3 2 0 4 2 1 4 3 |1 0  
 2 1 0 4 3 2 1 0 4 3 2 1 0 4 3  
 |0 4 | 2 1 4 3 1 0 |3 2|  
 0 4 3 2 1 0 4 3 2 1 0 4 3 2 1 0  
 |2 1 | 4 3 1 0 3 2 |0 4|  
 1-0-4-3-2-1-0-4-3-2-1-0-4-3-2  
 4 3 | 1 0 3 2 0 4 2 1|  
 2 1 4 3 1 0 3 2 0 4  
 0 | 2 4 1 3 |  
 3—0—2—4—1

21/  
 3—2—1—0—4  
 |2 1 0 4 |3  
 2 1 1 0 0 4 4 3 3 2  
 |1 0 0 4 4 3 3 2 |2 1  
 1 0 4 0 4 3 4 3 2 3 2 1 2 1 0  
 |0 4 | 4 3 3 2 2 1 |1 0|  
 0 4 3 4 3 2 3 2 1 2 1 0 1 0 4  
 |4 3 | 3 2 2 1 1 0 |0 4|  
 4-3-2-3-2-1-2-1-0-1-0-4-0-4-3  
 3 2 | 2 1 1 0 0 4 4 3|  
 2 1 1 0 0 4 4 3 3 2  
 1 | 0 4 3 2 |  
 0—4—3—2—1

22/  
 3—2—1—4—0  
 |2 1 4 0 |3  
 2 1 1 4 4 0 0 3 3 2  
 |1 4 4 0 0 3 3 2 |2 1  
 1 4 0 4 0 3 0 3 2 3 2 1 2 1 4  
 |4 0 | 0 3 3 2 2 1 |1 4|  
 4 0 3 0 3 2 3 2 1 2 1 4 1 4 0  
 |0 3 | 3 2 2 1 1 4 |4 0|  
 0-3-2-3-2-1-2-1-4-1-4-0-4-0-3  
 3 2 | 2 1 1 4 4 0 0 3|  
 2 1 1 4 4 0 0 3 3 2  
 1 | 4 0 3 2 |  
 4—0—3—2—1

23/  
 3—4—2—0—1  
 |4 2 0 1 |3  
 1 2 3 0 4 1 2 3 0 4  
 |3 0 4 1 2 3 0 4 |1 2  
 0 4 1 1 2 3 3 0 4 4 1 2 2 3 0  
 |1 2 | 3 0 4 1 2 3 |0 4|  
 2 3 0 0 4 1 1 2 3 3 0 4 4 1 2  
 |0 4 | 1 2 3 0 4 1 |2 3|  
 4-1-2-2-3-0-0-4-1-1-2-3-3-0-4  
 2 3 | 0 4 1 2 3 0 4 1|  
 0 4 1 2 3 0 4 1 2 3  
 1 | 3 4 2 0 |  
 3—4—2—0—1

24/  
 3—1—0—2—4  
 |4 3 1 0 |2  
 4 2 3 4 1 3 0 1 2 0  
 |2 0 4 2 3 4 1 3 |0 1  
 2 0 1 4 2 0 3 4 2 1 3 4 0 1 3  
 |0 1 | 2 0 4 2 3 4 |1 3|  
 0 1 3 2 0 1 4 2 0 3 4 2 1 3 4  
 |1 3 | 0 1 2 0 4 2 |3 4|  
 1-3-4-0-1-3-2-0-1-4-2-0-3-4-2  
 3 4 | 1 3 0 1 2 0 4 2|  
 4 2 3 4 1 3 0 1 2 0  
 2 | 4 3 1 0 |  
 0—2—4—3—1

25/  
 4—3—2—1—0  
 |0 4 3 2 |1  
 3 1 2 0 1 4 0 3 4 2  
 |4 2 3 1 2 0 1 4 |0 3  
 2 0 3 1 4 2 0 3 1 4 2 0 3 1 4  
 |3 1 | 2 0 1 4 0 3 |4 2|  
 1 4 2 0 3 1 4 2 0 3 1 4 2 0 3  
 |2 0 | 1 4 0 3 4 2 |3 1|  
 0-3-1-4-2-0-3-1-4-2-0-3-1-4-2  
 1 4 | 0 3 4 2 3 1 2 0|  
 2 0 1 4 0 3 4 2 3 1  
 3 | 2 1 0 4 |  
 4—3—2—1—0

26/  
 4—1—2—3—0  
 |0 4 1 2 |3  
 1 3 2 0 3 4 0 1 4 2  
 |4 2 1 3 2 0 3 4 |0 1  
 2 0 1 3 4 2 0 1 3 4 2 0 1 3 4  
 |1 3 | 2 0 3 4 0 1 |4 2|  
 3 4 2 0 1 3 4 2 0 1 3 4 2 0 1  
 |2 0 | 3 4 0 1 4 2 |1 3|  
 0-1-3-4-2-0-1-3-4-2-0-1-3-4-2  
 3 4 | 0 1 4 2 1 3 2 0|  
 2 0 3 4 0 1 4 2 1 3  
 1 | 2 3 0 4 |  
 4—1—2—3—0

27/  
 4—1—2—3—0  
 |1 2 3 0 |4  
 0 2 4 3 1 0 2 4 3 1  
 |4 3 1 0 2 4 3 1 |0 2  
 3 1 0 0 2 4 4 3 1 1 0 2 2 4 3  
 |0 2 | 4 3 1 0 2 4 |3 1|  
 2 4 3 3 1 0 0 2 4 4 3 1 1 0 2  
 |3 1 | 0 2 4 3 1 0 |2 4|  
 1-0-2-2-4-3-3-1-0-0-2-4-4-3-1  
 2 4 | 3 1 0 2 4 3 1 0|  
 3 1 0 2 4 3 1 0 2 4  
 0 | 4 1 2 3 |  
 4—1—2—3—0

28/  
 4—0—3—2—1  
 |1 4 0 3 |2  
 1 2 4 1 0 4 3 0 2 3  
 |2 3 1 2 4 1 0 4 |3 0  
 2 3 0 1 2 3 4 1 2 0 4 1 3 0 4  
 |3 0 | 2 3 1 2 4 1 |0 4|  
 3 0 4 2 3 0 1 2 3 4 1 2 0 4 1  
 |0 4 | 3 0 2 3 1 2 |4 1|  
 0-4-1-3-0-4-2-3-0-1-2-3-4-1-2  
 4 1 | 0 4 3 0 2 3 1 2|  
 1 2 4 1 0 4 3 0 2 3  
 2 | 1 4 0 3 |  
 3—2—1—4—0

29/  
 4—2—0—1—3  
 |2 0 1 3 |4  
 2 0 0 1 1 3 3 4 4 2  
 |0 1 1 3 3 4 4 2 |2 0  
 0 1 3 1 3 4 3 4 2 4 2 0 2 0 1  
 |1 3 | 3 4 4 2 2 0 |0 1|  
 1 3 4 3 4 2 4 2 0 2 0 1 0 1 3  
 |3 4 | 4 2 2 0 0 1 |1 3|  
 3-4-2-4-2-0-2-0-1-0-1-3-1-3-4  
 4 2 | 2 0 0 1 1 3 3 4|  
 2 0 0 1 1 3 3 4 4 2  
 0 | 1 3 4 2 |  
 1—3—4—2—0

30/  
 4—2—0—3—1  
 |2 0 3 1 |4  
 2 0 0 3 3 1 1 4 4 2  
 |0 3 3 1 1 4 4 2 |2 0  
 0 3 1 3 1 4 1 4 2 4 2 0 2 0 3  
 |3 1 | 1 4 4 2 2 0 |0 3|  
 3 1 4 1 4 2 4 2 0 2 0 3 0 3 1  
 |1 4 | 4 2 2 0 0 3 |3 1|  
 1-4-2-4-2-0-2-0-3-0-3-1-3-1-4  
 4 2 | 2 0 0 3 3 1 1 4|  
 2 0 0 3 3 1 1 4 4 2  
 0 | 3 1 4 2 |  
 3—1—4—2—0

31/  
 4—3—2—1—0  
 |3 2 1 0 |4  
 0 2 4 1 3 0 2 4 1 3  
 |4 1 3 0 2 4 1 3 |0 2  
 1 3 0 0 2 4 4 1 3 3 0 2 2 4 1  
 |0 2 | 4 1 3 0 2 4 |1 3|  
 2 4 1 1 3 0 0 2 4 4 1 3 3 0 2  
 |1 3 | 0 2 4 1 3 0 |2 4|  
 3-0-2-2-4-1-1-3-0-0-2-4-4-1-3  
 2 4 | 1 3 0 2 4 1 3 0|  
 1 3 0 2 4 1 3 0 2 4  
 0 | 4 3 2 1 |  
 4—3—2—1—0

32/  
 4—0—1—2—3  
 |3 4 0 1 |2  
 3 2 4 3 0 4 1 0 2 1  
 |2 1 3 2 4 3 0 4 |1 0  
 2 1 0 3 2 1 4 3 2 0 4 3 1 0 4  
 |1 0 | 2 1 3 2 4 3 |0 4|  
 1 0 4 2 1 0 3 2 1 4 3 2 0 4 3  
 |0 4 | 1 0 2 1 3 2 |4 3|  
 0-4-3-1-0-4-2-1-0-3-2-1-4-3-2  
 4 3 | 0 4 1 0 2 1 3 2|  
 3 2 4 3 0 4 1 0 2 1  
 2 | 3 4 0 1 |  
 1—2—3—4—0

[Count of Layer Units = 32]





definition. And find whether it has any duplication or lack of any number in it.

Every combination of Latin Units does not always make the correct solution, you must remember. It often makes a wrong solution.

When you combine any two units which have too little difference or too much difference between them, you will surely make a wrong solution. You will have to find a lot of duplication or lack of same numbers in it.

Only selected combinations could compose correct solutions.

Always examine it two times: both before the calculation and after that.

We had better make a reference table before all, that has 'resemblance' data between any two units, by measuring how many variables have the same values or the complimentary ones. You can come to this table before you combine any two units, and refer to it about their compatibility.

After the calculation, you also have to examine your composition to find whether it really has duplication or lack of any number or doesn't.

Since we must have the set of Standard Solutions of our object cubes, we have to remove all the reflected patterns or the rotated ones from the Primitive Solutions. Standard solutions are found among primitive in 1/48 necessarily in any case of magic cubes. What can we do before or after our calculations?

Then we have to prepare two sets of Inequality Conditions defining the relations among these critical variables:

```
n1<n5; n1<n21; n1<n25; n1<n101; n1<n105; n1<n121; n1<n125; n2<n6; n6<n26;
```

## 8. How to dictate our program for our 'C.E.C.' of Order5

Let me list out the second part of my Program, only a part.

```
/**/  
short int cnt;  
short cnt2, cnt3;  
short u1, u2, u3;  
short CC;  
short nm[126], flg[126];  
short bc[176][5];  
short d[4][129];  
short tlu[33][126];  
short mtc[33][33];  
short ts[481][129]; //Data Table for the Final Solutions  
/**/  
/* Main Program */  
int main(){  
short m,n;  
printf("\n** 'Latin Cubes' 5x5x5 for the Simultaneous Type: Self-complimentary and Pan-triagonal **\n");  
printf("** Made by Kanji Setsuda on June 13, 2010 with MacOSX & Xcode3.1.2 **\n");  
CC=4; cnt=0; cnt3=0;  
for(n=0;n<126;n++){nm[n]=0;}  
for(m=1;m<176;m++){for(n=0;n<5;n++){bc[m][n]=0;}}  
nm[63]=2; //Set n63=2;  
bc[37][2]=1; bc[38][2]=1; bc[39][2]=1; bc[76][2]=1; bc[77][2]=1; bc[78][2]=1; bc[79][2]=1;  
stp01(); //Make Latin Cubes by 'New Euler's Method'  
if(cnt3>0){prdlu(cnt3);}  
mkreftbl(); //Make Reference Table for the Compatibilities of any two Latin Units  
printf("\n** Complete Euler Cubes 5x5x5 for the Simultaneous Type: Self-complimentary and Pan-triagonal **\n");  
printf("** Sorted List of Standard Solutions by Kanji Setsuda on June 13, 2010 with MacOSX & Xcode 3.1 **\n");  
cnt=0; cnt3=0;  
cmbcmp(); //Combine and Compose 'Complete Euler Cubes' 5x5x5  
solsort(); //Sort the Result Data for Listing them in smarter style  
prsol1(); //Print out the List of Solutions  
printf(" [Total Count of Standard Solutions = %d] OK!\n",cnt);  
return 0;
```

```

}
/* ..... */
/**/
/* Combine and Compose */
void cmbcmp(){
    short n,d,lu,md,fc;
    lu=32; md=25;
    for(u1=0;u1<lu;u1++){
        for(u2=0;u2<lu;u2++){
            if(mtc[u2][u1]==md){
                for(u3=0;u3<lu;u3++){
                    if((mtc[u3][u1]==md)&&(mtc[u3][u2]==md)){cnt2=0;
                        for(n=1;n<126;n++){flg[n]=0;}
                        for(n=1;n<126;n++){
                            d=tlu[u1][n]*25+tlu[u2][n]*5+tlu[u3][n]+1;
                            nm[n]=d; flg[d]++;
                        }
                        fc=0;
                        for(n=1;n<126;n++){
                            if(flg[n]==1){fc++;}else{break;}
                        }
                        if(fc==125){d=nm[1];
                            if((d<nm[5])&&(d<nm[21])&&(d<nm[25])&&(d<nm[101])&&(d<nm[105])){
                                if((d<nm[121])&&(d<nm[125])&&(nm[2]<nm[6])&&(nm[6]<nm[26])){
                                    recsol();
                                }
                            }
                        }
                    }
                }
            }
        }
    }
}
/**/
/* Record the Solution Data to the Table */
void recsol(void){
    short n;
    for(n=1;n<126;n++){ts[cnt][n]=nm[n];}
    ts[cnt][0]=cnt+1; ts[cnt][126]=u1+1; ts[cnt][127]=u2+1; ts[cnt][128]=u3+1;
    cnt++;
}
/**/
/* Print the Solution List in Full Forms */
void prsoll(void){
    short l,ln,m,n,d1;
    d1=0;
    for(m=0;m<cnt;m++){
        if(ts[m][1]>d1){d1=ts[m][1];
            d[0][0]=ts[m][126]; d[1][0]=ts[m][127]; d[2][0]=ts[m][128];
            for(l=0;l<3;l++){ln=d[l][0]-1;
                for(n=1;n<126;n++){d[l][n]=tlu[ln][n];}
            }
            prdlu(4);
            for(n=1;n<126;n++){d[0][n]=ts[m][n];}
            d[0][0]=m+1; d[0][126]=ts[m][126]; d[0][127]=ts[m][127]; d[0][128]=ts[m][128];
            prptl();
        }
    }
}
/**/
/* Print the Object Solution */
void prptl(void){
    printf("%10d/H%5d/M%5d/L%45d#\n",d[0][126],d[0][127],d[0][128],d[0][0]);
    printf("-----%3d-----%3d-----%3d-----%3d\n",
        d[0][1],d[0][26],d[0][51],d[0][76],d[0][101]);
    printf("    !%3d%17d%17d%17d    !%3d#\n",
        d[0][2],d[0][27],d[0][52],d[0][77],d[0][102]);
}

```

```

printf("%4d%6d      %3d%6d      %3d%6d      %3d%6d      %3d%6d\n",
    d[0][6],d[0][3],d[0][31],d[0][28],d[0][56],d[0][53],d[0][81],d[0][78],d[0][106],d[0][103]);
printf("      |%3d%6d      %3d%6d      %3d%6d      %3d%6d      |%3d%6d\n",
    d[0][7],d[0][4],d[0][32],d[0][29],d[0][57],d[0][54],d[0][82],d[0][79],d[0][107],d[0][104]);
printf("%4d%6d%6d--%3d---%3d---%3d---%3d---%3d---%3d---%3d---%3d---%3d---%3d\n",
    d[0][11],d[0][8],d[0][5],d[0][36],d[0][33],d[0][30],d[0][61],d[0][58],d[0][55],
    d[0][86],d[0][83],d[0][80],d[0][111],d[0][108],d[0][105]);
printf("      |%3d%6d |      %3d%6d      %3d%6d      %3d%6d      |%3d%6d | \n",
    d[0][12],d[0][9],d[0][37],d[0][34],d[0][62],d[0][59],d[0][87],d[0][84],d[0][112],d[0][109]);
printf("%4d%6d%6d%5d%6d%6d%5d%6d%6d%5d%6d%6d%5d%6d%6d%5d%6d%6d\n",
    d[0][16],d[0][13],d[0][10],d[0][41],d[0][38],d[0][35],d[0][66],d[0][63],d[0][60],
    d[0][91],d[0][88],d[0][85],d[0][116],d[0][113],d[0][110]);
printf("      |%3d%6d |      %3d%6d      %3d%6d      %3d%6d      |%3d%6d | \n",
    d[0][17],d[0][14],d[0][42],d[0][39],d[0][67],d[0][64],d[0][92],d[0][89],d[0][117],d[0][114]);
printf("%4d--%3d---%3d---%3d---%3d---%3d---%3d---%3d---%3d---%3d---%3d---%3d---%3d%6d%6d\n",
    d[0][21],d[0][18],d[0][15],d[0][46],d[0][43],d[0][40],d[0][71],d[0][68],d[0][65],
    d[0][96],d[0][93],d[0][90],d[0][121],d[0][118],d[0][115]);
printf("%7d%6d |      %3d%6d      %3d%6d      %3d%6d      %3d%6d | \n",
    d[0][22],d[0][19],d[0][47],d[0][44],d[0][72],d[0][69],d[0][97],d[0][94],d[0][122],d[0][119]);
printf("%10d%6d      %3d%6d      %3d%6d      %3d%6d      %3d%6d\n",
    d[0][23],d[0][20],d[0][48],d[0][45],d[0][73],d[0][70],d[0][98],d[0][95],d[0][123],d[0][120]);
printf("%13d |%14d%17d%17d%17d | \n",
    d[0][24],d[0][49],d[0][74],d[0][99],d[0][124]);
printf("%16d-----%3d-----%3d-----%3d-----%3d-----%3d\n",
    d[0][25],d[0][50],d[0][75],d[0][100],d[0][125]);
}
/**/
/* Print the Latin Units */
void prdlu(short x){
    short i;
    if(x>3){printf("%24d/H%29d/M%29d/L\n",d[0][0],d[1][0],d[2][0]); x=3;}
    else if(x==3){printf("%25d/%30d/%30d/\n",d[0][0],d[1][0],d[2][0]);}
    else if(x==2){printf("%25d/%30d/\n",d[0][0],d[1][0]);}
    else{printf("%25d/\n",d[0][0]);}
    for(i=0;i<x;i++){
        printf("%2d-----%d-----%d-----%d-----%d",d[i][1],d[i][26],d[i][51],d[i][76],d[i][101]);
        if(i<2){printf(" ");}}
    printf("\n");
    for(i=0;i<x;i++){
        printf(" |%d%6d%6d%6d |%d",d[i][2],d[i][27],d[i][52],d[i][77],d[i][102]);
        if(i<2){printf(" ");}}
    printf("\n");
    for(i=0;i<x;i++){
        printf("%2d%2d%4d%2d%4d%2d%4d%2d%4d%2d",
            d[i][6],d[i][3],d[i][31],d[i][28],d[i][56],d[i][53],d[i][81],d[i][78],d[i][106],d[i][103]);
        if(i<2){printf(" ");}}
    printf("\n");
    for(i=0;i<x;i++){
        printf(" |%d%2d%4d%2d%4d%2d%4d%2d |%d%2d",
            d[i][7],d[i][4],d[i][32],d[i][29],d[i][57],d[i][54],d[i][82],d[i][79],d[i][107],d[i][104]);
        if(i<2){printf(" ");}}
    printf("\n");
    for(i=0;i<x;i++){
        printf("%2d%2d%2d-%d-%d-%d-%d-%d-%d-%d-%d-%d-%d",
            d[i][11],d[i][8],d[i][5],d[i][36],d[i][33],d[i][30],d[i][61],d[i][58],d[i][55],
            d[i][86],d[i][83],d[i][80],d[i][111],d[i][108],d[i][105]);
        if(i<2){printf(" ");}}
    printf("\n");
    for(i=0;i<x;i++){
        printf(" |%d%2d|%3d%2d%4d%2d%4d%2d |%d%2d|",
            d[i][12],d[i][9],d[i][37],d[i][34],d[i][62],d[i][59],d[i][87],d[i][84],d[i][112],d[i][109]);
        if(i<2){printf(" ");}}
    printf("\n");
    for(i=0;i<x;i++){
        printf("%2d%2d%2d%2d%2d%2d%2d%2d%2d%2d%2d%2d%2d%2d%2d%2d%2d%2d",
            d[i][16],d[i][13],d[i][10],d[i][41],d[i][38],d[i][35],d[i][66],d[i][63],d[i][60],

```

```

        d[i][91],d[i][88],d[i][85],d[i][116],d[i][113],d[i][110]);
    if(i<2){printf(" ");}}
printf("\n");
for(i=0;i<x;i++){
    printf(" %d%d|%3d%2d%4d%2d%4d%2d %d%d|",
        d[i][17],d[i][14],d[i][42],d[i][39],d[i][67],d[i][64],d[i][92],d[i][89],d[i][117],d[i][114]);
    if(i<2){printf(" ");}}
printf("\n");
for(i=0;i<x;i++){
    printf("%2d-%d-%d-%d-%d-%d-%d-%d-%d-%d-%d-%d%2d%2d",
        d[i][21],d[i][18],d[i][15],d[i][46],d[i][43],d[i][40],d[i][71],d[i][68],d[i][65],
        d[i][96],d[i][93],d[i][90],d[i][121],d[i][118],d[i][115]);
    if(i<2){printf(" ");}}
printf("\n");
for(i=0;i<x;i++){
    printf(" %d%d|%3d%2d%4d%2d%4d%2d%4d%2d|",
        d[i][22],d[i][19],d[i][47],d[i][44],d[i][72],d[i][69],d[i][97],d[i][94],d[i][122],d[i][119]);
    if(i<2){printf(" ");}}
printf("\n");
for(i=0;i<x;i++){
    printf(" %2d%2d%4d%2d%4d%2d%4d%2d%4d%2d",
        d[i][23],d[i][20],d[i][48],d[i][45],d[i][73],d[i][70],d[i][98],d[i][95],d[i][123],d[i][120]);
    if(i<2){printf(" ");}}
printf("\n");
for(i=0;i<x;i++){
    printf("%5d|%5d%6d%6d%6d|",
        d[i][24],d[i][49],d[i][74],d[i][99],d[i][124]);
    if(i<2){printf(" ");}}
printf("\n");
for(i=0;i<x;i++){
    printf("%6d-----%d-----%d-----%d-----%d",
        d[i][25],d[i][50],d[i][75],d[i][100],d[i][125]);
    if(i<2){printf(" ");}}
printf("\n");
}
/**/
/* Sort the Solution Data */
void solsort(){
    short mx,n,f;
    short d1,d2;
    long d3,d4,d5,d6;
    mx=cnt-1;
    do{f=0;
        for(n=0;n<mx;n++){
            d1=ts[n][1]*126+ts[n][32]; d2=ts[n+1][1]*126+ts[n+1][32];
            d3=(ts[n][2]*126+ts[n][3])*126+ts[n][4];
            d4=(ts[n+1][2]*126+ts[n+1][3])*126+ts[n+1][4];
            d5=(ts[n][6]*126+ts[n][11])*126+ts[n][16];
            d6=(ts[n+1][6]*126+ts[n+1][11])*126+ts[n+1][16];
            if(d1>d2){excsol(n); f=1;}
            else if((d1==d2)&&(d3>d4)){excsol(n); f=1;}
            else if((d1==d2)&&(d3==d4)&&(d5<d6)){excsol(n); f=1;}
        }
        mx--;
    }while(f>0);
}
/**/
void excsol(short x){
    short n,d;
    for(n=0;n<129;n++){d=ts[x][n]; ts[x][n]=ts[x+1][n]; ts[x+1][n]=d;}
}
/**/

```

## 9. What I have got as for the result of my computation:

Let me list out the recent result of my compositions, though only a part.

I have sorted the result to rearrange the listing order of them to the smarter style.

\*\* Complete Euler Cubes 5x5x5 for the Simultaneous Type: Self-complimentary and Pan-triagonal \*\*  
 \*\* Sorted List of Standard Solutions by Kanji Setsuda on June 7, 2010 with MacOSX & Xcode 3.1 \*\*

1/H					2/M					3/L				
0	4	3	2	1	0	1	2	3	4	0	2	4	1	3
1	0	4	3	2	1	2	3	4	0	2	4	1	3	0
1 2	0 1	4 0	3 4	2 3	4 2	0 3	1 4	2 0	3 1	2 4	4 1	1 3	3 0	0 2
2 3	1 2	0 1	4 0	3 4	0 3	1 4	2 0	3 1	4 2	4 1	1 3	3 0	0 2	2 4
2 3 4	1 2 3	0 1 2	4 0 1	3 4 0	3 1 4	4 2 0	0 3 1	1 4 2	2 0 3	4 1 3	1 3 0	3 0 2	0 2 4	2 4 1
3 4	2 3	1 2	0 1	4 0	4 2	0 3	1 4	2 0	3 1	1 3	3 0	0 2	2 4	4 1
3 4 0	2 3 4	1 2 3	0 1 2	4 0 1	2 0 3	3 1 4	4 2 0	0 3 1	1 4 2	1 3 0	3 0 2	0 2 4	2 4 1	4 1 3
4 0	3 4	2 3	1 2	0 1	3 1	4 2	0 3	1 4	2 0	3 0	0 2	2 4	4 1	1 3
4 0 1	3 4 0	2 3 4	1 2 3	0 1 2	1 4 2	2 0 3	3 1 4	4 2 0	0 3 1	3 0 2	0 2 4	2 4 1	4 1 3	1 3 0
0 1	4 0	3 4	2 3	1 2	2 0	3 1	4 2	0 3	1 4	0 2	2 4	4 1	1 3	3 0
1 2	0 1	4 0	3 4	2 3	3 1	4 2	0 3	1 4	2 0	2 4	4 1	1 3	3 0	0 2
2	1	0	4	3	4	0	1	2	3	4	1	3	0	2
3	2	1	0	4	0	1	2	3	4	1	3	0	2	4

1/H	2/M	3/L	1#				
1	108	90	67	49			
33	15	117	99	51			
48	65	5	42	107	24	89	101
55	92	32	74	14	26	116	8
70	82	124	47	64	76	4	41
97	114	54	91	31	73	13	30
87	104	16	69	81	123	46	63
119	6	96	113	53	95	35	72
109	21	38	86	103	20	68	85
11	28	118	10	100	112	52	94
43	60	25	37	102	19	84	121
75	27	9	111	93			
77	59	36	18	125			

1/H					2/M					8/L				
0	4	3	2	1	0	1	2	3	4	0	1	2	3	4
1	0	4	3	2	1	2	3	4	0	4	0	1	2	3
1 2	0 1	4 0	3 4	2 3	4 2	0 3	1 4	2 0	3 1	1 3	2 4	3 0	4 1	0 2
2 3	1 2	0 1	4 0	3 4	0 3	1 4	2 0	3 1	4 2	0 2	1 3	2 4	3 0	4 1
2 3 4	1 2 3	0 1 2	4 0 1	3 4 0	3 1 4	4 2 0	0 3 1	1 4 2	2 0 3	2 4 1	3 0 2	4 1 3	0 2 4	1 3 0
3 4	2 3	1 2	0 1	4 0	4 2	0 3	1 4	2 0	3 1	1 3	2 4	3 0	4 1	0 2
3 4 0	2 3 4	1 2 3	0 1 2	4 0 1	2 0 3	3 1 4	4 2 0	0 3 1	1 4 2	3 0 2	4 1 3	0 2 4	1 3 0	2 4 1
4 0	3 4	2 3	1 2	0 1	3 1	4 2	0 3	1 4	2 0	2 4	3 0	4 1	0 2	1 3
4 0 1	3 4 0	2 3 4	1 2 3	0 1 2	1 4 2	2 0 3	3 1 4	4 2 0	0 3 1	4 1 3	0 2 4	1 3 0	2 4 1	3 0 2
0 1	4 0	3 4	2 3	1 2	2 0	3 1	4 2	0 3	1 4	3 0	4 1	0 2	1 3	2 4
1 2	0 1	4 0	3 4	2 3	3 1	4 2	0 3	1 4	2 0	2 4	3 0	4 1	0 2	1 3
2	1	0	4	3	4	0	1	2	3	1	2	3	4	0
3	2	1	0	4	0	1	2	3	4	0	1	2	3	4

1/H	2/M	8/L	2#				
1	107	88	69	50			
35	11	117	98	54			
47	64	3	45	109	21	90	102
51	93	32	74	13	30	119	6
68	85	122	49	61	78	5	42
97	114	53	95	34	71	15	27
89	101	18	70	82	124	46	63
118	10	99	111	55	92	31	73
110	22	39	86	103	20	67	84
14	26	120	7	96	113	52	94
43	60	24	36	105	17	81	123
72	28	9	115	91			
76	57	38	19	125			

1/H					3/M					2/L				
0	4	3	2	1	0	2	4	1	3	0	1	2	3	4
1	0	4	3	2	2	4	1	3	0	1	2	3	4	0
1 2	0 1	4 0	3 4	2 3	2 4	4 1	1 3	3 0	0 2	4 2	0 3	1 4	2 0	3 1
2 3	1 2	0 1	4 0	3 4	4 1	1 3	3 0	0 2	2 4	0 3	1 4	2 0	3 1	4 2

2 3 4-1-2-3-0-1-2-4-0-1-3-4-0 4 1 3-1-3-0-3-0-2-0-2-4-2-4-1 3 1 4-4-2-0-0-3-1-1-4-2-2-0-3  
 |3 4| 2 3 1 2 0 1 |4 0| |1 3| 3 0 0 2 2 4 |4 1| |4 2| 0 3 1 4 2 0 |3 1|  
 3 4 0 2 3 4 1 2 3 0 1 2 4 0 1 1 3 0 3 0 2 0 2 4 2 4 1 4 1 3 2 0 3 3 1 4 4 2 0 0 3 1 1 4 2  
 |4 0| 3 4 2 3 1 2 |0 1| |3 0| 0 2 2 4 4 1 |1 3| |3 1| 4 2 0 3 1 4 |2 0|  
 4-0-1-3-4-0-2-3-4-1-2-3-0 1 2 3-0-2-0-2-4-2-4-1-4-1-3-1 3 0 1-4-2-2-0-3-3-1-4-4-2-0-0 3 1  
 0 1| 4 0 3 4 2 3 1 2| 0 2| 2 4 4 1 1 3 3 0| 2 0| 3 1 4 2 0 3 1 4 |2 0  
 1 2 0 1 4 0 3 4 2 3 2 4 4 1 1 3 3 0 0 2 3 1 4 2 0 3 1 4 2 0  
 2| 1 0 4 3| 4| 1 3 0 2| 4| 0 1 2 3|  
 3-----2-----1-----0-----4 1-----3-----0-----2-----4 0-----1-----2-----3-----4

1/H 3/M 2/L 3#  
 1-----112-----98-----59-----45  
 | 37 23 109 95 | 51  
 40 73 21 34 107 20 93 101 54 87  
 | 71 84 32 70 18 26 104 12 | 90 123  
 74 82 120 35 68 76 16 29 62 102 15 48 88 121 9  
 | 85 118 | 66 79 27 65 13 46 | 124 7 |  
 83 116 4 69 77 115 30 63 96 11 49 57 122 10 43  
 | 119 2 | 80 113 61 99 47 60 | 8 41 |  
 117 5 38 78 111 24 64 97 110 50 58 91 6 44 52  
 3 36 | 114 22 100 108 56 94 42 55 |  
 39 72 25 33 106 19 92 105 53 86  
 75 | 31 17 103 89 |  
 81-----67-----28-----14-----125

0-----2-----4-----1-----3 0-----4-----3-----2-----1 0-----1-----2-----3-----4  
 |2 4 1 3 0| |1 0 4 3 2| |1 2 3 4 0|  
 2 4 4 1 1 3 3 0 0 2 |2 3 1 2 0 1 4 0 |3 4 |0 3 1 4 2 0 3 1 |4 2  
 |4 1 1 3 3 0 0 2 |2 4 |2 3 1 2 0 1 4 0 |3 4 |0 3 1 4 2 0 3 1 |4 2  
 4 1 3-1-3-0-3-0-2-0-2-4-2-4-1 2 3 4-1-2-3-0-1-2-4-0-1-3-4-0 3 1 4-4-2-0-0-3-1-1-4-2-2-0-3  
 |1 3| 3 0 0 2 2 4 |4 1| |3 4| 2 3 1 2 0 1 |4 0| |4 2| 0 3 1 4 2 0 |3 1|  
 1 3 0 3 0 2 0 2 4 2 4 1 4 1 3 3 4 0 2 3 4 1 2 3 0 1 2 4 0 1 2 0 3 3 1 4 4 2 0 0 3 1 1 4 2  
 |3 0| 0 2 2 4 4 1 |1 3| |4 0| 3 4 2 3 1 2 |0 1| |3 1| 4 2 0 3 1 4 |2 0|  
 3-0-2-0-2-4-2-4-1-4-1-3-1 3 0 4-0-1-3-4-0-2-3-4-1-2-3-0 1 2 1-4-2-2-0-3-3-1-4-4-2-0-0 3 1  
 0 2| 2 4 4 1 1 3 3 0| 0 1| 4 0 3 4 2 3 1 2| 2 0| 3 1 4 2 0 3 1 4 |2 0  
 2 4 4 1 1 3 3 0 0 2 1 2 0 1 4 0 3 4 2 3 3 1 4 2 0 3 1 4 2 0  
 4| 1 3 0 2| 2| 1 0 4 3| 4| 0 1 2 3|  
 1-----3-----0-----2-----4 3-----2-----1-----0-----4 0-----1-----2-----3-----4

3/H 1/M 2/L 4#  
 1-----72-----118-----39-----85  
 | 57 103 49 95 | 11  
 60 113 101 34 47 80 93 21 14 67  
 | 111 44 32 90 78 6 24 52 | 70 123  
 114 42 100 35 88 16 76 9 62 22 55 108 68 121 29  
 | 45 98 | 86 19 7 65 53 106 | 124 27 |  
 43 96 4 89 17 75 10 63 116 51 109 37 122 30 83  
 | 99 2 | 20 73 61 119 107 40 | 28 81 |  
 97 5 58 18 71 104 64 117 50 110 38 91 26 84 12  
 3 56 | 74 102 120 48 36 94 82 15 |  
 59 112 105 33 46 79 92 25 13 66  
 115 | 31 77 23 69 |  
 41-----87-----8-----54-----125

1/H 2/M 4/L 5# 1/H 2/M 7/L 6#  
 1-----108-----90-----69-----47 1-----109-----88-----67-----50  
 |33 15 119 97 |51 |35 11 119 98 |52  
 48 65 5 44 109 22 87 101 66 83 49 62 3 45 107 21 90 104 66 83  
 |55 94 34 72 12 26 116 8 |98 115 |51 93 34 72 13 30 117 6 100 114  
 70 84 122 49 62 76 2 41 58 106 23 40 88 105 19 68 85 124 47 61 78 5 44 57 106 23 40 89 102 16  
 |99 112 | 52 91 31 73 13 30 120 9 | |99 112 | 53 95 32 71 15 29 116 8 |  
 89 102 16 67 81 123 46 63 80 3 45 59 110 24 37 87 101 18 70 84 122 46 63 80 4 42 56 108 25 39  
 117 6 | 96 113 53 95 35 74 |14 27 | |118 10 | 97 111 55 94 31 73 |14 27 |  
 107 21 38 86 103 20 68 85 124 50 64 77 4 42 56 110 24 37 86 103 20 69 82 121 48 65 79 2 41 58  
 11 28 | 118 10 100 114 54 92 32 71 | |12 26 | 120 9 96 113 54 92 33 75 |  
 43 60 25 39 104 17 82 121 61 78 43 60 22 36 105 19 81 123 64 77  
 75 | 29 7 111 93 | 74 | 28 7 115 91 |  
 79-----57-----36-----18-----125 76-----59-----38-----17-----125



1/H	3/M	6/L		7#		3/H	1/M	6/L		8#
1	114	98	57	45		1	74	118	37	85
39	23	107	95	51		59	103	47	95	11
40 73	21 32	109 20	93 101	52 89		60 113	101 32	49 80	93 21	12 69
71 82	34 70	18 26	102 14	90 123		111 42	34 90	78 6	22 54	70 123
72 84 120	35 68 76	16 27 64	104 15 48	88 121 7		112 44 100	35 88 16	76 7 64	24 55 108	68 121 27
85 118	66 77	29 65	13 46	122 9		45 98	86 17	9 65	53 106	122 29
83 116 2	67 79 115	30 63 96	11 47 59	124 10 43		43 96 2	87 19 75	10 63 116	51 107 39	124 30 83
117 4	80 113	61 97	49 60	8 41		97 4	20 73	61 117	109 40	28 81
119 5 38	78 111 22	62 99 110	50 58 91	6 42 54		99 5 58	18 71 102	62 119 50	110 38 91	26 82 14
3 36	112 24	100 108	56 92	44 55		3 56	72 104	120 48	36 92	84 15
37 74	25 33	106 17	94 105	53 86		57 114	105 33	46 77	94 25	13 66
75	31	19	103	87		115	31	79	23	67
81	69	28	12	125		41	89	8	52	125

1/H	4/M	2/L		9#		1/H	6/M	3/L		10#
1	112	98	69	35		1	118	90	57	49
37	23	119	85	51		43	15	107	99	51
40 73	21 44	117 10	83 101	54 87		48 65	5 32	117 24	89 101	56 93
71 94	42 60	8 26	104 12	90 123		55 82	42 74	14 26	106 18	98 115
74 92 110	45 58 76	6 29 62	102 15 48	88 121 19		60 92 124	47 64 76	4 31 68	116 23 40	88 105 7
95 108	56 79	27 65	13 46	124 17		97 114	54 81	41 73	13 30	110 17
93 106 4	59 77 115	30 63 96	11 49 67	122 20 33		87 104 6	59 91 123	46 63 80	3 35 67	120 22 39
109 2	80 113	61 99	47 70	18 31		109 16	96 113	53 85	45 72	12 29
107 5 38	78 111 24	64 97 120	50 68 81	16 34 52		119 21 38	86 103 10	58 95 122	50 62 79	2 34 66
3 36	114 22	100 118	66 84	32 55		11 28	108 20	100 112	52 84	44 71
39 72	25 43	116 9	82 105	53 86		33 70	25 37	102 9	94 121	61 78
75	41	7	103	89		75	27	19	111	83
91	57	28	14	125		77	69	36	8	125

1/H	6/M	8/L		11#		3/H	5/M	2/L		12#
1	117	88	59	50		1	72	108	39	95
45	11	107	98	54		67	103	49	85	11
47 64	3 35	119 21	90 102	56 93		70 113	101 44	47 80	83 21	14 57
51 83	42 74	13 30	109 16	100 112		111 34	42 90	78 16	24 52	60 123
58 95 122	49 61 78	5 32 69	116 23 40	87 104 6		114 32 100	45 88 6	76 19 62	22 55 118	58 121 29
97 114	53 85	44 71	15 27	106 18		35 98	86 9	17 65	53 116	124 27
89 101 8	60 92 124	46 63 80	2 34 66	118 25 37		33 96 4	89 7 75	20 63 106	51 119 37	122 30 93
108 20	99 111	55 82	41 73	12 29		99 2	10 73	61 109	117 40	28 91
120 22 39	86 103 10	57 94 121	48 65 77	4 31 68		97 5 68	8 71 104	64 107 50	120 38 81	26 94 12
14 26	110 17	96 113	52 84	43 75		3 66	74 102	110 48	36 84	92 15
33 70	24 36	105 7	91 123	62 79		69 112	105 43	46 79	82 25	13 56
72	28	19	115	81		115	41	77	23	59
76	67	38	9	125		31	87	18	54	125

1/H	4/M	6/L		13#		1/H	6/M	4/L		14#
1	114	98	67	35		1	118	90	59	47
39	23	117	85	51		43	15	109	97	51
40 73	21 42	119 10	83 101	52 89		48 65	5 34	119 22	87 101	56 93
71 92	44 60	8 26	102 14	90 123		55 84	44 72	12 26	106 18	98 115
72 94 110	45 58 76	6 27 64	104 15 48	88 121 17		60 94 122	49 62 76	2 31 68	116 23 40	88 105 9
95 108	56 77	29 65	13 46	122 19		99 112	52 81	41 73	13 30	110 19
93 106 2	57 79 115	30 63 96	11 47 69	124 20 33		89 102 6	57 91 123	46 63 80	3 35 69	120 24 37
107 4	80 113	61 97	49 70	18 31		107 16	96 113	53 85	45 74	14 27
109 5 38	78 111 22	62 99 120	50 68 81	16 32 54		117 21 38	86 103 10	58 95 124	50 64 77	4 32 66
3 36	112 24	100 118	66 82	34 55		11 28	108 20	100 114	54 82	42 71
37 74	25 43	116 7	84 105	53 86		33 70	25 39	104 7	92 121	61 78
75	41	9	103	87		75	29	17	111	83
91	59	28	12	125		79	67	36	8	125

1/H	6/M	7/L		15#		3/H	5/M	6/L		16#
1	119	88	57	50		1	74	108	37	95
45	11	109	98	52		69	103	47	85	11
49 62	3 35	117 21	90 104	56 93		70 113	101 42	49 80	83 21	12 59
51 83	44 72	13 30	107 16	100 114		111 32	44 90	78 16	22 54	60 123
58 95 124	47 61 78	5 34 67	116 23 40	89 102 6		112 34 100	45 88 6	76 17 64	24 55 118	58 121 27
99 112	53 85	42 71	15 29	106 18		35 98	86 7	19 65	53 116	122 29
87 101 8	60 94 122	46 63 80	4 32 66	118 25 39		33 96 2	87 9 75	20 63 106	51 117 39	124 30 93
108 20	97 111	55 84	41 73	14 27		97 4	10 73	61 107	119 40	28 91
120 24 37	86 103 10	59 92 121	48 65 79	2 31 68		99 5 68	8 71 102	62 109 50	120 38 81	26 92 14
12 26	110 19	96 113	54 82	43 75		3 66	72 104	110 48	36 82	94 15
33 70	22 36	105 9	91 123	64 77		67 114	105 43	46 77	84 25	13 56
74	28	17	115	81		115	41	79	23	57
76	69	38	7	125		31	89	18	52	125

4/H 1/M 2/L 17#

1-----72-----118-----89-----35  
 |57 103 99 45 |11  
 60 113 101 84 97 30 43 21 14 67  
 111 94 82 40 28 6 24 52 |70 123  
 114 92 50-85-38-16-26-9-62-22-55-108-68-121-79  
 |95 48 | 36 19 7 65 53 106 124 77 |  
 93 46 4 39 17 75 10 63 116 51 109 87 122 80 33  
 |49 2 | 20 73 61 119 107 90 |78 31 |  
 47-5-58-18-71-104-64-117-100-110-88-41-76 34 12  
 3 56 | 74 102 120 98 86 44 32 15 |  
 59 112 105 83 96 29 42 25 13 66  
 115 | 81 27 23 69 |  
 91-----37-----8-----54-----125

5/H 2/M 3/L 18#

1-----108-----40-----67-----99  
 |83 15 117 49 |51  
 98 65 5 92 107 24 39 101 66 33  
 |55 42 82 74 14 76 116 8 |48 115  
 70 32 124-97-64-26-4-91-58-106-23-90-38-105-17  
 |47 114 | 54 41 81 73 13 80 120 7 |  
 37 104 16 69 31 123 96 63 30 3 95 57 110 22 89  
 |119 6 | 46 113 53 45 85 72 |12 79 |  
 109-21-88-36-103-20-68-35-122-100-62-29-2 94 56  
 11 78 | 118 10 50 112 52 44 84 71 |  
 93 60 25 87 102 19 34 121 61 28  
 75 | 77 9 111 43 |  
 27-----59-----86-----18-----125

5/H 2/M 8/L 19#

1-----107-----38-----69-----100  
 |85 11 117 48 |54  
 97 64 3 95 109 21 40 102 66 33  
 |51 43 82 74 13 80 119 6 |50 112  
 68 35 122-99-61-28-5-92-59-106-23-90-37-104-16  
 |47 114 | 53 45 84 71 15 77 116 8 |  
 39 101 18 70 32 124 96 63 30 2 94 56 108 25 87  
 |118 10 | 49 111 55 42 81 73 |12 79 |  
 110-22-89-36-103-20-67-34-121-98-65-27-4 91 58  
 14 76 | 120 7 46 113 52 44 83 75 |  
 93 60 24 86 105 17 31 123 62 29  
 72 | 78 9 115 41 |  
 26-----57-----88-----19-----125

5/H 3/M 2/L 20#

1-----112-----48-----59-----95  
 |87 23 109 45 |51  
 90 73 21 84 107 20 43 101 54 37  
 |71 34 82 70 18 76 104 12 |40 123  
 74 32 120-85-68-26-16-79-62-102-15-98-38-121-9  
 |35 118 | 66 29 77 65 13 96 124 7 |  
 33 116 4 69 27 115 80 63 46 11 99 57 122 10 93  
 |119 2 | 30 113 61 49 97 60 |8 91 |  
 117-5-88-28-111-24-64-47-110-100-58-41-6 94 52  
 3 86 | 114 22 50 108 56 44 92 55 |  
 89 72 25 83 106 19 42 105 53 36  
 75 | 81 17 103 39 |  
 31-----67-----78-----14-----125

4/H 1/M 6/L 21#

1-----74-----118-----87-----35  
 |59 103 97 45 |11  
 60 113 101 82 99 30 43 21 12 69  
 111 92 84 40 28 6 22 54 |70 123  
 112 94 50-85-38-16-26-7-64-24-55-108-68-121-77  
 |95 48 | 36 17 9 65 53 106 122 79 |  
 93 46 2 37 19 75 10 63 116 51 107 89 124 80 33  
 |47 4 | 20 73 61 117 109 90 |78 31 |  
 49-5-58-18-71-102-62-119-100-110-88-41-76 32 14  
 3 56 | 72 104 120 98 86 42 34 15 |  
 57 114 105 83 96 27 44 25 13 66  
 115 | 81 29 23 67 |  
 91-----39-----8-----52-----125

5/H 2/M 4/L 22#

1-----108-----40-----69-----97  
 |83 15 119 47 |51  
 98 65 5 94 109 22 37 101 66 33  
 |55 44 84 72 12 76 116 8 |48 115  
 70 34 122-99-62-26-2-91-58-106-23-90-38-105-19  
 |49 112 | 52 41 81 73 13 80 120 9 |  
 39 102 16 67 31 123 96 63 30 3 95 59 110 24 87  
 |117 6 | 46 113 53 45 85 74 |14 77 |  
 107-21-88-36-103-20-68-35-124-100-64-27-4 92 56  
 11 78 | 118 10 50 114 54 42 82 71 |  
 93 60 25 89 104 17 32 121 61 28  
 75 | 79 7 111 43 |  
 29-----57-----86-----18-----125

5/H 2/M 7/L 23#

1-----109-----38-----67-----100  
 |85 11 119 48 |52  
 99 62 3 95 107 21 40 104 66 33  
 |51 43 84 72 13 80 117 6 |50 114  
 68 35 124-97-61-28-5-94-57-106-23-90-39-102-16  
 |49 112 | 53 45 82 71 15 79 116 8 |  
 37 101 18 70 34 122 96 63 30 4 92 56 108 25 89  
 |118 10 | 47 111 55 44 81 73 |14 77 |  
 110-24-87-36-103-20-69-32-121-98-65-29-2 91 58  
 12 76 | 120 9 46 113 54 42 83 75 |  
 93 60 22 86 105 19 31 123 64 27  
 74 | 78 7 115 41 |  
 26-----59-----88-----17-----125

5/H 3/M 6/L 24#

1-----114-----48-----57-----95  
 |89 23 107 45 |51  
 90 73 21 82 109 20 43 101 52 39  
 |71 32 84 70 18 76 102 14 |40 123  
 72 34 120-85-68-26-16-77-64-104-15-98-38-121-7  
 |35 118 | 66 27 79 65 13 96 122 9 |  
 33 116 2 67 29 115 80 63 46 11 97 59 124 10 93  
 |117 4 | 30 113 61 47 99 60 |8 91 |  
 119-5-88-28-111-22-62-49-110-100-58-41-6 92 54  
 3 86 | 112 24 50 108 56 42 94 55 |  
 87 74 25 83 106 17 44 105 53 36  
 75 | 81 19 103 37 |  
 31-----69-----78-----12-----125

4/H 5/M 2/L 25#

1-----72-----108-----89-----45  
 |67 103 99 35 |11  
 70 113 101 94 97 30 33 21 14 57  
 111 84 92 40 28 16 24 52 |60 123  
 114 82 50-95-38-6-26-19-62-22-55-118-58-121-79  
 |85 48 | 36 9 17 65 53 116 124 77 |  
 83 46 4 39 7 75 20 63 106 51 119 87 122 80 43  
 |49 2 | 10 73 61 109 117 90 |78 41 |  
 47-5-68-8-71-104-64-107-100-120-88-31-76 44 12  
 3 66 | 74 102 110 98 86 34 42 15 |  
 69 112 105 93 96 29 32 25 13 56  
 115 | 91 27 23 59 |  
 81-----37-----18-----54-----125

5/H 4/M 2/L 26#

1-----112-----48-----69-----85  
 |87 23 119 35 |51  
 90 73 21 94 117 10 33 101 54 37  
 |71 44 92 60 8 76 104 12 |40 123  
 74 42 110-95-58-26-6-79-62-102-15-98-38-121-19  
 |45 108 | 56 29 77 65 13 96 124 17 |  
 43 106 4 59 27 115 80 63 46 11 99 67 122 20 83  
 |109 2 | 30 113 61 49 97 70 |18 81 |  
 107-5-88-28-111-24-64-47-120-100-68-31-16 84 52  
 3 86 | 114 22 50 118 66 34 82 55 |  
 89 72 25 93 116 9 32 105 53 36  
 75 | 91 7 103 39 |  
 41-----57-----78-----14-----125

5/H 6/M 3/L 27#

1-----118-----40-----57-----99  
 |93 15 107 49 |51  
 98 65 5 82 117 24 39 101 56 43  
 |55 32 92 74 14 76 106 18 |48 115  
 60 42 124-97-64-26- 4-81-68-116-23-90-38-105- 7  
 |47 114 | 54 31 91 73 13 80 110 17 |  
 37 104 6 59 41 123 96 63 30 3 85 67 120 22 89  
 109 16 | 46 113 53 35 95 72 |12 79 |  
 119-21-88-36-103-10-58-45-122-100-62-29- 2 84 66  
 11 78 | 108 20 50 112 52 34 94 71 |  
 83 70 25 87 102 9 44 121 61 28  
 75 | 77 19 111 33 |  
 27-----69-----86-----8-----125

5/H 6/M 8/L 28#

1-----117-----38-----59-----100  
 |95 11 107 48 |54  
 97 64 3 85 119 21 40 102 56 43  
 |51 33 92 74 13 80 109 16 |50 112  
 58 45 122-99-61-28- 5-82-69-116-23-90-37-104- 6  
 |47 114 | 53 35 94 71 15 77 106 18 |  
 39 101 8 60 42 124 96 63 30 2 84 66 118 25 87  
 108 20 | 49 111 55 32 91 73 |12 79 |  
 120-22-89-36-103-10-57-44-121-98-65-27- 4 81 68  
 14 76 | 110 17 46 113 52 34 93 75 |  
 83 70 24 86 105 7 41 123 62 29  
 72 | 78 19 115 31 |  
 26-----67-----88-----9-----125

4/H 5/M 6/L 29#

1-----74-----108-----87-----45  
 |69 103 97 35 |11  
 70 113 101 92 99 30 33 21 12 59  
 111 82 94 40 28 16 22 54 |60 123  
 112 84 50-95-38- 6-26-17-64-24-55-118-58-121-77  
 |85 48 | 36 7 19 65 53 116 122 79 |  
 83 46 2 37 9 75 20 63 106 51 117 89 124 80 43  
 |47 4 | 10 73 61 107 119 90 |78 41 |  
 49- 5-68- 8-71-102-62-109-100-120-88-31-76 42 14  
 3 66 | 72 104 110 98 86 32 44 15 |  
 67 114 105 93 96 27 34 25 13 56  
 115 | 91 29 23 57 |  
 81-----39-----18-----52-----125

5/H 4/M 6/L 30#

1-----114-----48-----67-----85  
 |89 23 117 35 |51  
 90 73 21 92 119 10 33 101 52 39  
 |71 42 94 60 8 76 102 14 |40 123  
 72 44 110-95-58-26- 6-77-64-104-15-98-38-121-17  
 |45 108 | 56 27 79 65 13 96 122 19 |  
 43 106 2 57 29 115 80 63 46 11 97 69 124 20 83  
 107 4 | 30 113 61 47 99 70 |18 81 |  
 109- 5-88-28-111-22-62-49-120-100-68-31-16 82 54  
 3 86 | 112 24 50 118 66 32 84 55 |  
 87 74 25 93 116 7 34 105 53 36  
 75 | 91 9 103 37 |  
 41-----59-----78-----12-----125

5/H 6/M 4/L 31#

1-----118-----40-----59-----97  
 |93 15 109 47 |51  
 98 65 5 84 119 22 37 101 56 43  
 |55 34 94 72 12 76 106 18 |48 115  
 60 44 122-99-62-26- 2-81-68-116-23-90-38-105- 9  
 |49 112 | 52 31 91 73 13 80 110 19 |  
 39 102 6 57 41 123 96 63 30 3 85 69 120 24 87  
 107 16 | 46 113 53 35 95 74 |14 77 |  
 117-21-88-36-103-10-58-45-124-100-64-27- 4 82 66  
 11 78 | 108 20 50 114 54 32 92 71 |  
 83 70 25 89 104 7 42 121 61 28  
 75 | 79 17 111 33 |  
 29-----67-----86-----8-----125

5/H 6/M 7/L 32#

1-----119-----38-----57-----100  
 |95 11 109 48 |52  
 99 62 3 85 117 21 40 104 56 43  
 |51 33 94 72 13 80 107 16 |50 114  
 58 45 124-97-61-28- 5-84-67-116-23-90-39-102- 6  
 |49 112 | 53 35 92 71 15 79 106 18 |  
 37 101 8 60 44 122 96 63 30 4 82 66 118 25 89  
 108 20 | 47 111 55 34 91 73 |14 77 |  
 120-24-87-36-103-10-59-42-121-98-65-29- 2 81 68  
 12 76 | 110 19 46 113 54 32 93 75 |  
 83 70 22 86 105 9 41 123 64 27  
 74 | 78 17 115 31 |  
 26-----69-----88-----7-----125

1/H 2/M 14/L 33#

2-----106-----88-----70-----49  
 |34 12 116 98 |55  
 46 65 3 44 110 22 89 101 67 83  
 |52 93 31 75 13 29 120 7 |99 111  
 68 84 121-50-62-78- 4-41-60-107-23-39-86-105-17  
 |96 115 | 53 94 35 72 14 26 117 8 |  
 90 102 18 69 81 125 47 63 79 1 45 57 108 24 36  
 118 9 | 100 112 54 91 32 73 |11 30 |  
 109-21-40-87-103-19-66-85-122-48-64-76- 5 42 58  
 15 27 | 119 6 97 113 51 95 33 74 |  
 43 59 25 37 104 16 82 123 61 80  
 71 | 28 10 114 92 |  
 77-----56-----38-----20-----124

1/H 2/M 12/L 36#

2-----108-----89-----70-----46  
 |33 14 120 96 |52  
 48 64 4 45 110 21 86 102 67 83  
 |54 95 35 71 11 27 117 8 |98 114  
 69 85 121-50-61-77- 1-42-58-107-23-39-88-104-20  
 |100 111 | 51 92 32 73 13 29 119 10 |  
 90 101 17 66 82 123 47 63 79 3 44 60 109 25 36  
 116 7 | 97 113 53 94 34 75 |15 26 |  
 106-22-38-87-103-19-68-84-125-49-65-76- 5 41 57  
 12 28 | 118 9 99 115 55 91 31 72 |  
 43 59 24 40 105 16 81 122 62 78  
 74 | 30 6 112 93 |  
 80-----56-----37-----18-----124

1/H 4/M 10/L 39#

2-----111-----98-----70-----34  
 |36 23 120 84 |52  
 39 73 22 45 116 9 83 102 55 86  
 |72 95 41 59 8 27 105 11 |89 123  
 75 91 109-44-58-77- 7-30-61-101-14-48-88-122-20  
 |94 108 | 57 80 26 64 13 47 125 16 |  
 93 107 5 60 76 114 29 63 97 12 50 66 121 19 33  
 110 1 | 79 113 62 100 46 69 |18 32 |  
 106- 4-38-78-112-25-65-96-119-49-68-82-17 35 51  
 3 37 | 115 21 99 118 67 85 31 54 |  
 40 71 24 43 117 10 81 104 53 87  
 74 | 42 6 103 90 |  
 92-----56-----28-----15-----124

1/H 4/M 13/L 42#

2-----115-----98-----66-----34  
 |39 22 120 83 |51  
 40 71 23 44 116 7 84 105 52 88  
 |72 93 45 56 8 29 101 12 |89 125  
 73 94 110-41-57-78- 9-30-61-102-13-49-90-121-17  
 |95 106 | 58 79 26 62 14 50 122 18 |  
 91 107 3 59 80 111 27 63 99 15 46 67 123 19 35  
 108 4 | 76 112 64 100 47 68 |20 31 |  
 109- 5-36-77-113-24-65-96-117-48-69-85-16 32 53  
 1 37 | 114 25 97 118 70 81 33 54 |  
 38 74 21 42 119 10 82 103 55 86  
 75 | 43 6 104 87 |  
 92-----60-----28-----11-----124

4/H 1/M 10/L 45#

2-----71-----118-----90-----34  
 |56 103 100 44 |12  
 59 113 102 85 96 29 43 22 15 66  
 112 95 81 39 28 7 25 51 |69 123  
 115 91 49-84-38-17-27-10-61-21-54-108-68-122-80  
 |94 48 | 37 20 6 64 53 107 125 76 |  
 93 47 5 40 16 74 9 63 117 52 110 86 121 79 33  
 |50 1 | 19 73 62 120 106 89 |78 32 |  
 46- 4-58-18-72-105-65-116-99-109-88-42-77 35 11  
 3 57 | 75 101 119 98 87 45 31 14 |  
 60 111 104 83 97 30 41 24 13 67  
 114 | 82 26 23 70 |  
 92-----36-----8-----55-----124

4/H 1/M 13/L 48#

2-----75-----118-----86-----34  
 |59 102 100 43 |11  
 60 111 103 84 96 27 44 25 12 68  
 112 93 85 36 28 9 21 52 |69 125  
 113 94 50-81-37-18-29-10-61-22-53-109-70-121-77  
 |95 46 | 38 19 6 62 54 110 122 78 |  
 91 47 3 39 20 71 7 63 119 55 106 87 123 79 35  
 |48 4 | 16 72 64 120 107 88 |80 31 |  
 49- 5-56-17-73-104-65-116-97-108-89-45-76 32 13  
 1 57 | 74 105 117 98 90 41 33 14 |  
 58 114 101 82 99 30 42 23 15 66  
 115 | 83 26 24 67 |  
 92-----40-----8-----51-----124

4/H 5/M 10/L 51#

2-----71-----108-----90-----44  
 |66 103 100 34 |12  
 69 113 102 95 96 29 33 22 15 56  
 112 85 91 39 28 17 25 51 |59 123  
 115 81 49-94-38- 7-27-20-61-21-54-118-58-122-80  
 |84 48 | 37 10 16 64 53 117 125 76 |  
 83 47 5 40 6 74 19 63 107 52 120 86 121 79 43  
 |50 1 | 9 73 62 110 116 89 |78 42 |  
 46- 4-68- 8-72-105-65-106-99-119-88-32-77 45 11  
 3 67 | 75 101 109 98 87 35 41 14 |  
 70 111 104 93 97 30 31 24 13 57  
 114 | 92 26 23 60 |  
 82-----36-----18-----55-----124

4/H 5/M 13/L 54#

2-----75-----108-----86-----44  
 |69 102 100 33 |11  
 70 111 103 94 96 27 34 25 12 58  
 112 83 95 36 28 19 21 52 |59 125  
 113 84 50-91-37- 8-29-20-61-22-53-119-60-121-77  
 |85 46 | 38 9 16 62 54 120 122 78 |  
 81 47 3 39 10 71 17 63 109 55 116 87 123 79 45  
 |48 4 | 6 72 64 110 117 88 |80 41 |  
 49- 5-66- 7-73-104-65-106-97-118-89-35-76 42 13  
 1 67 | 74 105 107 98 90 31 43 14 |  
 68 114 101 92 99 30 32 23 15 56  
 115 | 93 26 24 57 |  
 82-----40-----18-----51-----124

1/H 2/M 20/L 57#

4-----106-----88-----70-----47  
 |32 14 116 98 |55  
 46 65 3 42 110 24 87 101 69 83  
 |54 93 31 75 13 27 120 9 |97 111  
 68 82 121-50-64-78- 2-41-60-109-23-37-86-105-19  
 |96 115 | 53 92 35 74 12 26 119 8 |  
 90 104 18 67 81 125 49 63 77 1 45 59 108 22 36  
 118 7 | 100 114 52 91 34 73 |11 30 |  
 107-21-40-89-103-17-66-85-124-48-62-76- 5 44 58  
 15 29 | 117 6 99 113 51 95 33 72 |  
 43 57 25 39 102 16 84 123 61 80  
 71 | 28 10 112 94 |  
 79-----56-----38-----20-----122

1/H 3/M 26/L 81#

5-----112-----98-----59-----41  
 |36 25 107 93 |54  
 37 74 23 31 109 20 91 102 55 88  
 |75 83 32 69 18 26 104 15 |86 122  
 73 81 117-34-70-78-16-27-64-105-13-46-87-124-10  
 |82 119 | 68 76 29 65 11 47 125 8 |  
 84 120 3 66 77 114 30 63 96 12 49 60 123 6 42  
 118 1 | 79 115 61 97 50 58 | 7 44 |  
 116- 2-39-80-113-21-62-99-110-48-56-92- 9 45 53  
 4 40 | 111 22 100 108 57 94 43 51 |  
 38 71 24 35 106 17 95 103 52 89  
 72 | 33 19 101 90 |  
 85-----67-----28-----14-----121

1/H 10/M 3/L 97#

6-----103-----90-----72-----44  
 |28 15 122 94 |56  
 43 65 10 47 102 19 89 106 71 78  
 |60 97 27 69 14 31 121 3 |93 115  
 75 77 119-42-64-81- 9-46-53-101-18-40-88-110-22  
 |92 114 | 59 96 26 68 13 35 125 2 |  
 87 109 21 74 76 118 41 63 85 8 50 52 105 17 39  
 124 1 | 91 113 58 100 30 67 |12 34 |  
 104-16-38-86-108-25-73-80-117-45-62-84- 7 49 51  
 11 33 | 123 5 95 112 57 99 29 66 |  
 48 55 20 37 107 24 79 116 61 83  
 70 | 32 4 111 98 |  
 82-----54-----36-----23-----120

1/H 10/M 14/L 117#

7-----101-----88-----75-----44  
 |29 12 121 93 |60  
 41 65 8 49 105 17 89 106 72 78  
 |57 98 26 70 13 34 125 2 |94 111  
 73 79 116-45-62-83- 9-46-55-102-18-39-86-110-22  
 |91 115 | 58 99 30 67 14 31 122 3 |  
 90 107 23 74 76 120 42 63 84 6 50 52 103 19 36  
 123 4 | 95 112 59 96 27 68 |11 35 |  
 104-16-40-87-108-24-71-80-117-43-64-81-10 47 53  
 15 32 | 124 1 92 113 56 100 28 69 |  
 48 54 20 37 109 21 77 118 61 85  
 66 | 33 5 114 97 |  
 82-----51-----38-----25-----119

1/H 10/M 20/L 133#

9-----101-----88-----75-----42  
 |27 14 121 93 |60  
 41 65 8 47 105 19 87 106 74 78  
 |59 98 26 70 13 32 125 4 |92 111  
 73 77 116-45-64-83- 7-46-55-104-18-37-86-110-24  
 |91 115 | 58 97 30 69 12 31 124 3 |  
 90 109 23 72 76 120 44 63 82 6 50 54 103 17 36  
 123 2 | 95 114 57 96 29 68 |11 35 |  
 102-16-40-89-108-22-71-80-119-43-62-81-10 49 53  
 15 34 | 122 1 94 113 56 100 28 67 |  
 48 52 20 39 107 21 79 118 61 85  
 66 | 33 5 112 99 |  
 84-----51-----38-----25-----117

3/H 9/M 26/L 149#

10-----67-----123-----39-----76  
 |51 110 42 98 |14  
 52 114 108 26 44 85 96 17 15 73  
 115 48 27 89 83 1 19 60 |71 117  
 113 46 92-29-90-23-81- 2-64-20-58-101-72-119-35  
 |47 94 | 88 21 4 65 56 102 120 33 |  
 49 95 8 86 22 69 5 63 121 57 104 40 118 31 77  
 |93 6 | 24 70 61 122 105 38 |32 79 |  
 91- 7-54-25-68-106-62-124-45-103-36-97-34 80 13  
 9 55 | 66 107 125 43 37 99 78 11 |  
 53 111 109 30 41 82 100 18 12 74  
 112 | 28 84 16 75 |  
 50-----87-----3-----59-----116

1/H 18/M 3/L 161#

```

16-----103-----90-----72-----34
|28      15      122      84      |66
33 65    20 47    102 9     89 116   71 78
|70 97    27 59    14 41    121 3     |83 115
75 77 109-32-64-91-19-46-53-101- 8-40-88-120-22
|82 114 | 69 96    26 58    13 45    125 2 |
87 119 21 74 76 108 31 63 95 18 50 52 105 7 39
124 1 | 81 113    68 100    30 57    |12 44 |
104- 6-38-86-118-25-73-80-107-35-62-94-17 49 51
11 43 | 123 5     85 112    67 99    29 56 |
48 55    10 37    117 24    79 106   61 93
60 | 42      4      111      98 |
92-----54-----36-----23-----110

```

1/H 18/M 14/L 181#

```

17-----101-----88-----75-----34
|29      12      121      83      |70
31 65    18 49    105 7     89 116   72 78
|67 98    26 60    13 44    125 2     |84 111
73 79 106-35-62-93-19-46-55-102- 8-39-86-120-22
|81 115 | 68 99    30 57    14 41    122 3 |
90 117 23 74 76 110 32 63 94 16 50 52 103 9 36
123 4 | 85 112    69 96    27 58    |11 45 |
104- 6-40-87-118-24-71-80-107-33-64-91-20 47 53
15 42 | 124 1     82 113    66 100    28 59 |
48 54    10 37    119 21    77 108   61 95
56 | 43      5      114      97 |
92-----51-----38-----25-----109

```

1/H 18/M 20/L 197#

```

19-----101-----88-----75-----32
|27      14      121      83      |70
31 65    18 47    105 9     87 116   74 78
|69 98    26 60    13 42    125 4     |82 111
73 77 106-35-64-93-17-46-55-104- 8-37-86-120-24
|81 115 | 68 97    30 59    12 41    124 3 |
90 119 23 72 76 110 34 63 92 16 50 54 103 7 36
123 2 | 85 114    67 96    29 58    |11 45 |
102- 6-40-89-118-22-71-80-109-33-62-91-20 49 53
15 44 | 122 1     84 113    66 100    28 57 |
48 52    10 39    117 21    79 108   61 95
56 | 43      5      112      99 |
94-----51-----38-----25-----107

```

3/H 17/M 26/L 213#

```

20-----57-----123-----39-----76
|51      120     32      98      |14
52 114   118 26   34 95    96 7     15 73
115 48   27 89    93 1     9 70    |71 107
113 46 82-29-90-23-91- 2-64-10-68-101-72-109-45
|47 84 | 88 21     4 65     66 102   110 43 |
49 85 18 86 22 59 5 63 121 67 104 40 108 41 77
|83 16 | 24 60     61 122   105 38   |42 79 |
81-17-54-25-58-116-62-124-35-103-36-97-44 80 13
19 55 | 56 117    125 33    37 99    78 11 |
53 111   119 30    31 92    100 8     12 74
112 | 28      94      6      75 |
50-----87-----3-----69-----106

```

3/H 26/M 1/L 225#

```

21-----60-----114-----43-----77
|52      121     35      89      |18
57 118   111 27   45 96    79 10    23 64
123 39   32 93    86 2     20 71    |54 110
113 29 85-42-98-14-76- 7-68-25-61-102-59-120-46
|34 95 | 88 4      17 73     51 107   125 36 |
44 100 11 78 9 70 22 63 104 56 117 48 115 26 82
|90 1 | 19 75     53 109   122 38   |31 92 |
80- 6-67-24-65-101-58-119-50-112-28-84-41 97 13
16 72 | 55 106    124 40    33 94     87 3 |
62 103   116 47    30 81    99 15     8 69
108 | 37      91      5      74 |
49-----83-----12-----66-----105

```

3/H 26/M 9/L 233#

```

22-----59-----115-----43-----76
|51      122     34      90      |18
56 118   112 26   44 97    80 9     23 65
123 40   31 93    87 1     19 72    |55 109
113 30 84-41-98-15-77- 6-68-24-62-101-60-119-47
|35 94 | 88 5      16 73     52 106   124 37 |
45 99 12 78 10 69 21 63 105 57 116 48 114 27 81
|89 2 | 20 74     53 110   121 38   |32 91 |
79- 7-66-25-64-102-58-120-49-111-28-85-42 96 13
17 71 | 54 107    125 39    33 95     86 3 |
61 103   117 46    29 82    100 14    8 70
108 | 36      92      4      75 |
50-----83-----11-----67-----104

```

3/H 26/M 17/L 241#

```

24-----57-----115-----43-----76
|51      124     32      90      |18
56 118   114 26   42 99    80 7     23 65
123 40   31 93    89 1     17 74    |55 107
113 30 82-41-98-15-79- 6-68-22-64-101-60-117-49
|35 92 | 88 5      16 73     54 106   122 39 |
45 97 14 78 10 67 21 63 105 59 116 48 112 29 81
|87 4 | 20 72     53 110   121 38   |34 91 |
77- 9-66-25-62-104-58-120-47-111-28-85-44 96 13
19 71 | 52 109    125 37    33 95     86 3 |
61 103   119 46    27 84    100 12     8 70
108 | 36      94      2      75 |
50-----83-----11-----69-----102

```

3/H 26/M 27/L 249#

```

25-----57-----113-----44-----76
|52      123     34      86      |20
56 118   115 29   42 96    78 10    24 62
125 39   32 91    88 5     19 72    |51 108
114 27 81-41-98-15-80- 9-67-22-61-103-58-120-49
|31 93 | 90 4      17 71     53 110   124 37 |
43 100 14 79 7 66 21 63 105 60 119 47 112 26 83
|89 2 | 16 73     55 109   122 36   |33 95 |
77- 6-68-23-65-104-59-117-46-111-28-85-45 99 12
18 75 | 54 107    121 38    35 94     87 1 |
64 102   116 48    30 84    97 11     8 70
106 | 40      92      3      74 |
50-----82-----13-----69-----101

```

\*\* Counts according to the Values of n1: \*\*

1: 32, 2: 24, 3: 0, 4: 24, 5: 16, 6: 20, 7: 16, 8: 0,  
9: 16, 10: 12, 11: 0, 12: 0, 13: 0, 14: 0, 15: 0, 16: 20,  
17: 16, 18: 0, 19: 16, 20: 12, 21: 8, 22: 8, 23: 0, 24: 8,  
25: 8, 26: 0, 27: 0, 28: 0, 29: 0, 30: 0, 31: 0, 32: 0,  
33: 0, 34: 0, 35: 0, 36: 0, 37: 0, 38: 0, 39: 0, 40: 0,  
41: 0, 42: 0, 43: 0, 44: 0, 45: 0, 46: 0, 47: 0, 48: 0,

[Total Count of Standard Solutions = 256] OK!

## 10. What I think about the result:

I know about the total solution count at last.

How happy I am that I can report about that!

The list of solution counts above tells us all of them are the multiples of 4 ( $=2^2$ ):

$8=2 \times 4$ ;  $12=3 \times 4$ ;  $16=4 \times 4$ ;  $20=5 \times 4$ ;  $24=6 \times 4$ ;  $32=8 \times 4$ ;

The total count is finally divided as:  $256 = 64 \times 4 = 2^6 \times 2^2$ ;

What does it mean by these factors? What structure do they imply about?

The total solution count also means:  $32 \times 32 \times 32 \times (3/8) \times (1/48) = 256$ .

32 is the count of Latin Units,  $3/8$  indicates the ratio of correct solutions to all the possible combinations and  $1/48$  means the ratio of standard solutions to primitive.

It is certain that we could have got the rarest solution set of all. I think it surely means we have got one of the most precious sets of magic jewels.

The next job for us must be to make any analysis on the structure of solution set itself.

---

(English Version written by Kanji Setsuda on July 6, 2010 with MacOSX and Xcode3)

E-mail Address: [jag12001@nifty.com](mailto:jag12001@nifty.com)