Magic squares J(p)
by Jarosław Wróblewski

Version 1 (Oct. 16, 2005)

Let p be a prime of the form $8n \pm 3$. We are going to construct a magic square $J(p)$ of size $2^p \times 2^p$.

We are going to identify integers from 0 to $2^p - 1$ with sequences of their p binary digits (bits), possibly filled with leading zeros. We refer to positions of bits as from 0-th to $(p-1)$-th. It doesn't really matter whether we put oldest bit last or first as long as we are consistent.

Let for $0 \leqslant i < p$ the sequence a_i has all bits 0 except i-th bit, which is 1.

In Mathematica format:

```
a[0]=Join[{1},Table[0,{i,1,p-1}]];
Do[a[i]=RotateRight[a[0],i],{i,1,p-1}];
```

Let a_p has 1 on position i iff i is quadratic residue mod p, 0 otherwise. We consider $i = 0$ to be quadratic residue here.

Let a_{p+i}, where $1 \leqslant i < p$, be a_p with bits rotated right by i positions.

```
a[p]=Ceiling[Mod[PowerMod[Range[p]-1,(p-1)/2,p]+1,p]/2];
Do[a[p+i]=RotateRight[a[p],i],{i,1,p-1}];
```

Let $b_i = a_{i+1}$ for $0 \leqslant i \leqslant p-2$ and $b_{p-1} = a_0$.

Let b_{p+i} be a_{p+i-1} with all bits reversed, for $1 \leqslant i \leqslant p-1$. Let b_p be a_{2p-1} with all bits reversed.

```
Do[b[i]=a[Mod[i+1,p]],{i,0,p-1}];
Do[b[p+i]=1-a[p+Mod[i+p-1,p]],{i,0,p-1}];
```

The table $J(p)$ has entries

$$m_{ij} = \sum_{k=0}^{2p-1} 2^k \cdot (i \circ a_k + j \circ b_k)_{(mod\ 2)}\ , \qquad (\heartsuit)$$

where $i \circ a_k$ means bitwise multiplication and then adding the products, i.e. counting common occurences of 1's in i and a_k. The sum in parentheses is then taken modulo 2. Indices i and j are ranging from 0 to $2^p - 1$.

```
m=Table[
Sum[
2^k*Mod[Plus@@(Drop[IntegerDigits[2^p+i,2],1]*a[k]+
Drop[IntegerDigits[2^p+j,2],1]*b[k]),2]
,{k,0,2p-1}]
,{i,0,2^p-1},{j,0,2^p-1}];
```

Matrix $J(p)$ has consecutive integers from 0 to 4^p-1 as entries if the $2p\times 2p$ matrix X whose rows are concatenated a_i and b_i has odd determinant.

```
X=Table[Join[a[i],b[i]],{i,0,2p-1}];
```

That is the reason for assuming $p=8n\pm 3$.

Let c_i be bitwise XOR of a_i and b_i.

We will call a set of p-bit sequences XOR-d-independent iff every nonempty subset of at most d elements has nonzero bitwise XOR of its elements.

If $(a_i)_{0\leqslant i<2p}$ is XOR-d-independent, then square $J(p)$ has d-magic columns.

If $(b_i)_{0\leqslant i<2p}$ is XOR-d-independent, then square $J(p)$ has d-magic rows.

If $(c_i)_{0\leqslant i<2p}$ is XOR-d-independent, then square $J(p)$ has both main diagonals d-magic.

I hope that the above facts are known or can be verified by people deep in the subject. I would hate to go through detailed proof of them. The idea is to replace powers of 2 by variables in (♡) and to observe that under above XOR-independency conditions, sum of powers up to d-th of a row, column or diagonal can be expressed without actually looking at particular bits of a_i and b_i. Note that this sum of powers is a polynomial in $2p$ variables. Under XOR-independency conditions coefficients of this polynomial are "averaged" the same way, no matter what particular a's and b's are.

Multimagic degree of J(p)

p	columns	rows	diagonals	square
5	3	2	2	2
11	6	5	6	5
13	5	6	6	5
19	6	7	6	6
29	11+	10	10	10
37	6+	6+	6+	6+
43	6+	6+	6+	6+

Note: 6+ means I have verified 6-magic (hexamagic) but haven't tested for 7-magic (heptamagic).

Checking XOR-independence

In C: store XOR-sums of d elements on linked lists. Keep checking whether newly stored XOR-sum is already there. If so, system is not XOR-2d-independent.

If no XOR-sum is repeated, system is XOR-2d-independent provided it has been known to be XOR-(2d-1)-independent.

Keep previously stored XOR-sums of d elements on linked lists and check them against XOR-sums of $d+1$ elements. If no sum is repeated, we are sure system is XOR-(2d+1)-independent.

Remarks

You can take any integer as p and any binary vectors as a_i and b_i to create your own magic square. But if matrix X has even determinant, you do not get distinct entries.

If XOR-independence of (a_i), (b_i) and (c_i) is small, multimagic degree of your square is small. You can always present the square by generating 0-th row an 0-th column of the square. The rest is filled as XOR table: m_{ij} is bitwise XOR of m_{0j} and m_{i0}.

Files `ab<p>.txt` contain a_i and b_i in the form of decimal numbers.

In formula (♡) you can replace 2^k by ANY numbers and you get multimagic square. You need to put there ANY permutation of powers of 2 to get a square with consecutive integers.

I have verified that X has odd determinant for $p = 8n \pm 3$ and $p < 50$. I have no general proof of that, but I am 99,99999999% sure that is true for all p of that form.

I feel that multimagic degree of $J(p)$ tends to ∞ as $p \to \infty$, but I have no clue how to prove it.

Using 5magic.exe

Create file `ab<p>.txt` with a_i and b_i in decimal form. One number per line, a's come first from a_0 to a_{2p-1}, then b's. Number p must be less than 32.

Then run `5magic p`

Same applies to 7magic.exe and next programs.

Decamagic J(29)

Computations I have performed indicate that $J(29)$ is 10-magic (decamagic ???).

It has size $2^{29} \times 2^{29}$ or $536870912 \times 536870912$ and contains integer entries from 0 to $2^{58} - 1 = 288230376151711743$.

It has 11-magic columns, unlikely 12-magic, but it hasn't been ruled out at the moment.