Latin Squares Meet Group Theory: Tools and Insights in GAP - LatinSquare GAP Package

Seyyed Ali Mohammadiyeh
Department of Pure Mathematics, Faculty of Mathematical Sciences
University of Kashan, Kashan 87317-53153, I. R. Iran
alim@kashanu.ac.ir, max@std.kashanu.ac.ir

April 9, 2025

Who I am

ALI

Senior Software Engineer and CTO of IT company

Programming professional 10 years, up to 15 years coding

B.Sc. in Mathematics with double major in Computer Science and Computer Engineering

Definition

A **Latin square** of order n is an $n \times n$ array filled with n different symbols, each occurring exactly once in each row and exactly once in each column.

Definition

A **Latin square** of order n is an $n \times n$ array filled with n different symbols, each occurring exactly once in each row and exactly once in each column.

- Often, symbols are the integers $\{1, 2, \ldots, n\}$.
- No symbol repeats in any row or column.

Applications

Latin squares appear in:

- **Statistics:** Experimental design to ensure that treatments are balanced across different conditions, allowing for a more reliable analysis of experimental results.
- **Cryptography:** Encryption algorithms.
- **Game and Puzzle design:** Sudoku is a partial Latin square!
- **Combinatorics:** Designs, finite geometry, etc.

Group Theory Relation

Latin squares are related to group theory and quasigroups. A quasigroup $(Q,)$ is a non-empty set Q with a binary operation (that is, a magma, indicating that a quasigroup has to satisfy the closure property), obeying the Latin square property.

Historical Notes

- Studied by **Leonhard Euler** in the 18th century.
- Euler's conjecture on orthogonal Latin squares was famous (later disproven for $n = 6$).
- The Korean mathematician Choi Seok-jeong was the first to publish an example of Latin squares of order nine, in order to construct a magic square in 1700, predating Leonhard Euler by 67 years.

Euler's Contribution

The concept of "Latin square" was inspired by mathematical papers by Leonhard Euler (1707–1783), who used Latin characters as symbols, but any set of symbols can be used: the integer sequence 1, 2, 3 can be replaced by the alphabetic sequence A, B, C. Euler began the general theory of Latin squares.

Choi Seok-jeong

Choi Seok-jeong was a Korean politician and mathematician who was the first to find orthogonal Latin squares. He constructed magic squares and invented the Hexagonal Tortoise Problem.

Choi Seok-jeong

Here is Choi Seok-jeong's orthogonal Latin squares of order 9 in modern notation:

From the orthogonal Latin squares, he was able to construct a magic square of order 9.

(5,1)	(6,3)	(4,2)	(8,7)	(9,9)	(7,8)	(2,4)	(3,6)	(1,5)
(4,3)	(5,2)	(6,1)	(7,9)	(8,8)	(9,7)	(1,6)	(2,5)	(3,4)
(6,2)	(4,1)	(5,3)	(9,8)	(7,7)	(8,9)	(3,5)	(1,4)	(2,6)
(2,7)	(3,9)	(1,8)	(5,4)	(6,6)	(4,5)	(8,1)	(9,3)	(7,2)
(1,9)	(2,8)	(3,7)	(4,6)	(5,5)	(6,4)	(7,3)	(8,2)	(9,1)
(3,8)	(1,7)	(2,9)	(6,5)	(4,4)	(5,6)	(9,2)	(7,1)	(8,3)
(8,4)	(9,6)	(7,5)	(2,1)	(3,3)	(1,2)	(5,7)	(6,9)	(4,8)

Terminology

- **Order:** The size n of the square.
- **Symbol set:** The n elements used (commonly 1 to n).
- **Orthogonal Latin squares (OLS):** Two Latin squares where each ordered pair occurs once.
- **Mutually orthogonal Latin squares (MOLS):** ...

How Many Latin Squares?

The number grows extremely fast:

- $n = 1$: 1
- $n = 2$: 2
- $n = 3$: 12
- $n = 4$: 576
- $n = 5$: 161280
- $n = 6$: over 8.9 billion!

How Many Latin Squares?

The number grows extremely fast:

- $n = 1$: 1
- $n = 2$: 2
- $n = 3$: 12
- $n = 4$: 576
- $n = 5$: 161280
- $n = 6$: over 8.9 billion!

Exact formulas exist only for small n.

Latin Squares in Sudoku

- A valid Sudoku solution is a Latin square with extra region constraints.
- Sudoku = Latin square + 3×3 box conditions.

Latin Squares in Sudoku

9	1	3	4	2	7	5	8	6
6	8	7	9	1	5	3	2	4
2	5	4	6	8	3	1	7	9
4	7	9	1	3	2	6	5	8
1	6	2	5	9	8	7	4	3
5	3	8	7	6	4	2	9	1
3	4	5	8	7	1	9	6	2
7	2	6	3	4	9	8	1	5
8	9	1	2	5	6	4	3	7

Latin Squares of Order 1

The only Latin square of order 1 is:

$$\begin{bmatrix} 1 \end{bmatrix}$$

Latin Squares of Order 2

There are two Latin squares of order 2:

$$\begin{bmatrix} 1 & 2 \\ 2 & 1 \end{bmatrix} \quad \begin{bmatrix} 2 & 1 \\ 1 & 2 \end{bmatrix}$$

These two are isotopically equivalent and represent the cyclic group $\mathbb{Z}/2\mathbb{Z}$.

Latin Squares of Order 3

There are 12 Latin squares of order 3. One example is:

$$\begin{bmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 3 & 1 & 2 \end{bmatrix}$$

This represents the cyclic group $\mathbb{Z}/3\mathbb{Z}$.

$$\begin{bmatrix} 1 & 2 & 3 \\ 2 & 3 & 1 \\ 3 & 1 & 2 \end{bmatrix}, \begin{bmatrix} 1 & 2 & 3 \\ 3 & 1 & 2 \\ 2 & 3 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 3 & 2 \\ 2 & 1 & 3 \\ 3 & 2 & 1 \end{bmatrix}, \begin{bmatrix} 1 & 3 & 2 \\ 3 & 2 & 1 \\ 2 & 1 & 3 \end{bmatrix},$$

$$\begin{bmatrix} 2 & 1 & 3 \\ 1 & 3 & 2 \\ 3 & 2 & 1 \end{bmatrix}, \begin{bmatrix} 2 & 1 & 3 \\ 3 & 2 & 1 \\ 1 & 3 & 2 \end{bmatrix}, \begin{bmatrix} 2 & 3 & 1 \\ 1 & 2 & 3 \\ 3 & 1 & 2 \end{bmatrix}, \begin{bmatrix} 2 & 3 & 1 \\ 3 & 1 & 2 \\ 1 & 2 & 3 \end{bmatrix},$$

$$\begin{bmatrix} 3 & 2 & 1 \\ 1 & 3 & 2 \\ 2 & 1 & 3 \end{bmatrix}, \begin{bmatrix} 3 & 2 & 1 \\ 2 & 1 & 3 \\ 1 & 3 & 2 \end{bmatrix}, \begin{bmatrix} 3 & 1 & 2 \\ 1 & 2 & 3 \\ 2 & 3 & 1 \end{bmatrix}, \begin{bmatrix} 3 & 1 & 2 \\ 2 & 3 & 1 \\ 1 & 2 & 3 \end{bmatrix},$$

Latin Squares of Order 4

There are 576 Latin squares of order 4. Here are two examples:

$$\begin{bmatrix} 1 & 2 & 3 & 4 \\ 2 & 3 & 4 & 1 \\ 3 & 4 & 1 & 2 \\ 4 & 1 & 2 & 3 \end{bmatrix} \quad \begin{bmatrix} 1 & 2 & 4 & 3 \\ 2 & 3 & 1 & 4 \\ 3 & 4 & 2 & 1 \\ 4 & 1 & 3 & 2 \end{bmatrix}$$

Latin Squares of Order 5

There are 161280 Latin squares of order 5. Here are some examples:

$$\begin{bmatrix} 1 & 2 & 3 & 4 & 5 \\ 2 & 3 & 4 & 5 & 1 \\ 3 & 4 & 5 & 1 & 2 \\ 4 & 5 & 1 & 2 & 3 \\ 5 & 1 & 2 & 3 & 4 \end{bmatrix}, \quad \begin{bmatrix} 1 & 2 & 3 & 5 & 4 \\ 2 & 3 & 4 & 1 & 5 \\ 3 & 4 & 5 & 2 & 1 \\ 4 & 5 & 1 & 3 & 2 \\ 5 & 1 & 2 & 4 & 3 \end{bmatrix},$$

and 161278 more.
(You can display these squares across multiple slides if necessary.)

Numbers of Latin Squares of Order $n = 1, 2, \ldots$

n	$L(n, n)$	Reference
1	1	
2	1	
3	1	
4	4	
5	56	1782, Euler, 1890, Cayley - 1915, MacMahon
6	9408	1890, Frolov - 1900, Tarry
7	16942080	1939, Norton - 1948, Sade - 1951, Saxena
8	535281401856	1967, Wells
9	377597570964258816	1975, Bammel and Rothstein
10	7580721483160132811489280	1995, McKay and Rogoyski
11	5363937773277371298119673540771840	2005, McKay and Wanless
12	1.62×10^{44}	1995, McKay and Rogoyski
13	2.51×10^{56}	1995, McKay and Rogoyski
14	2.33×10^{70}	1995, McKay and Rogoyski
15	1.5×10^{86}	1995, McKay and Rogoyski

Table: Numbers of Latin Squares of order $n = 1, 2, \ldots$.

Numbers of Latin Squares of Order $n = 1, 2, \ldots$

n	reduced Latin squares of size n (sequence A000315 in the OEIS)	all Latin squares of size n (sequence A002860 in the OEIS)
1	1	1
2	1	2
3	1	12
4	4	576
5	56	161,280
6	9,408	812,851,200
7	16,942,080	61,479,419,904,000
8	535,281,401,856	108,776,032,459,082,956,800
9	377,597,570,964,258,816	5,524,751,496,156,892,842,531,225,600
10	7,580,721,483,160,132,811,489,280	9,982,437,658,213,039,871,725,064,756,920,320,000
11	5,363,937,773,277,371,298,119,673,540,771,840	776,966,836,171,770,144,107,444,346,734,230,682,311,065,600,000
12	1.62×10^{44}	
13	2.51×10^{56}	
14	2.33×10^{70}	
15	1.50×10^{86}	

Construction Methods

Several techniques:

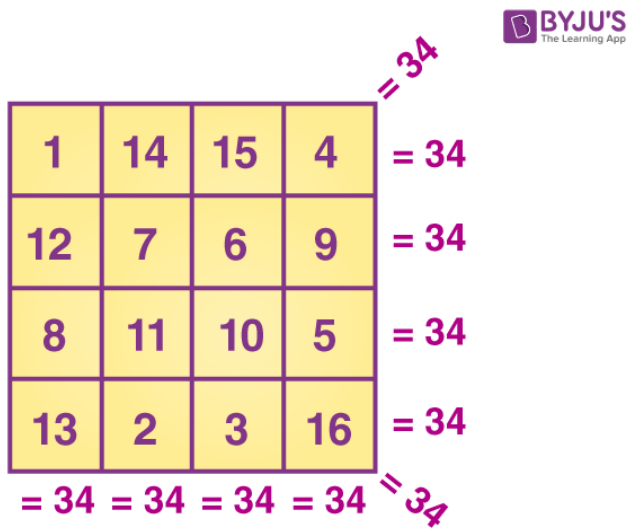
- **Cyclic method:** Rotate rows.
- **Backtracking:** Recursive filling (used in our package).
- **Group-based construction:** Use permutation groups.

Reduced Latin Squares and Normalized Latin Squares

Latin squares are also sometimes called "reduced" or "normalized" Latin squares when the first row and first column are in natural order.

Magic square

2	7	6	→15
9	5	1	→15
4	3	8	→15
↙15	↓15	↓15	↓15 ↘15



Trivial Case Latin squares

A 1x1 square is considered trivial because it only contains one cell, and any symbol can be placed there without violating the Latin square property (each symbol appears only once in each row and column).

Non-Trivial Latin squares

A Latin square is non-trivial if it has more than one row and column, and each symbol appears exactly once in each row and column.

Mutually orthogonal Latin squares (MOLS)

A set of Latin squares of the same order such that every pair of squares are orthogonal (that is, form a Graeco-Latin square) is called a set of mutually orthogonal Latin squares (or pairwise orthogonal Latin squares) and usually abbreviated as MOLS or MOLS(n) when the order is made explicit.
For example, a set of MOLS(4) is given by:

$$\begin{array}{cccc} 1 & 2 & 3 & 4 \\ 2 & 1 & 4 & 3 \\ 3 & 4 & 1 & 2 \\ 4 & 3 & 2 & 1 \end{array} \qquad \begin{array}{cccc} 1 & 2 & 3 & 4 \\ 4 & 3 & 2 & 1 \\ 2 & 1 & 4 & 3 \\ 3 & 4 & 1 & 2 \end{array} \qquad \begin{array}{cccc} 1 & 2 & 3 & 4 \\ 3 & 4 & 1 & 2 \\ 4 & 3 & 2 & 1 \\ 2 & 1 & 4 & 3 \end{array}.$$

And a set of MOLS(5):

$$\begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \\ 2 & 3 & 4 & 5 & 1 \\ 3 & 4 & 5 & 1 & 2 \\ 4 & 5 & 1 & 2 & 3 \\ 5 & 1 & 2 & 3 & 4 \end{array} \quad \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \\ 3 & 4 & 5 & 1 & 2 \\ 5 & 1 & 2 & 3 & 4 \\ 2 & 3 & 4 & 5 & 1 \\ 4 & 5 & 1 & 2 & 3 \end{array} \quad \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \\ 5 & 1 & 2 & 3 & 4 \\ 4 & 5 & 1 & 2 & 3 \\ 3 & 4 & 5 & 1 & 2 \\ 2 & 3 & 4 & 5 & 1 \end{array} \quad \begin{array}{ccccc} 1 & 2 & 3 & 4 & 5 \\ 4 & 5 & 1 & 2 & 3 \\ 2 & 3 & 4 & 5 & 1 \\ 5 & 1 & 2 & 3 & 4 \\ 3 & 4 & 5 & 1 & 2 \end{array}.$$

Balanced Latin Square

Balanced Latin Squares are special cases of latin square that remove immediate carry-over effects: A condition will precede another exactly once (or twice, if the number of conditions is odd). The balanced latin square generator proposed and mathematically proved by James V. Bradley in "Complete Counterbalancing of Immediate Sequential Effects in a Latin Square Design".

Albrecht Dürer's magic square

...

Luo Shu magic square

...

Sagrada Família magic square

...

Gardner square

...

Parker square

...

Recap

- Latin squares are $n \times n$ grids with unique symbols in each row and column.
- They have many real-world applications.
- Enumeration and generation are key challenges.

Next, we'll look at how the GAP package works with Latin squares.

Introduction

This package provides functions to generate and count Latin squares using GAP. It offers:

- **Generation of Latin squares:** Construct all Latin squares of a given order.
- **Counting Latin squares:** Count them without generating all explicitly. More can be added as needed.

Installation

Installation: Place the package inside GAP's `pkg` directory.

```
ali@DESKTOP-GOMG29E:~/.gap/pkg$ ls
AutoDoc-2023.06.19    Semigroups            digraphs-1.10.0  images-1.3.3  latinsquare  packagemaker
PackageManager-1.6.2  datastructures-0.3.1  genss-1.6.9      io-4.9.1      orb-5.0.0
ali@DESKTOP-GOMG29E:~/.gap/pkg$ git clone https://github.com/BaseMax/LatinSquareGAP latinsquare
```

Loading

Loading:

```
gap> LoadPackage("LatinSquare");
```

You should see a confirmation message.

Prototype: `LatinSquareList(n, c)`
Description: Generates all Latin squares of order `n`.

LatinSquareCount

Prototype: `LatinSquareCount(n)`
Description: Counts all Latin squares of order `n`, optionally using partial data.

```
gap> LatinSquareCount(3);
12
```

LatinSquareRow

Prototype: `LatinSquareRow(n, r, c)`
Description: Computes valid completions of a row for a Latin square of order `n`.

LatinSquareCountRow

Prototype: `LatinSquareCountRow(n, k, r, c)`
Description: Recursively counts valid completions of the current row.

- `n`: Order of the square.
- `k`: Rows fixed so far.
- `r`: Column restrictions.
- `c`: Current row.

Counting Latin Squares

To count the number of Latin squares of order 4:

```
gap> LatinSquareCount(4);
```

Generating Latin Squares

To generate all Latin squares of order 3:

```
brk> LatinSquareList(1);
[ [ [ 1 ] ] ]
brk> LatinSquareList(2);
[ [ [ 1, 2 ], [ 2, 1 ] ], [ [ 2, 1 ], [ 1, 2 ] ]
    ]
brk> LatinSquareList(3);
[ [ [ 1, 2, 3 ], [ 2, 3, 1 ], [ 3, 1, 2 ] ], [ [
    1, 2, 3 ], [ 3, 1, 2 ], [ 2, 3, 1 ] ], [ [
   1, 3, 2 ], [ 2, 1, 3 ], [ 3, 2, 1 ] ],
  [ [ 1, 3, 2 ], [ 3, 2, 1 ], [ 2, 1, 3 ] ], [ [
      2, 1, 3 ], [ 1, 3, 2 ], [ 3, 2, 1 ] ], [ [
      2, 1, 3 ], [ 3, 2, 1 ], [ 1, 3, 2 ] ],
  [ [ 2, 3, 1 ], [ 1, 2, 3 ], [ 3, 1, 2 ] ], [ [
      2, 3, 1 ], [ 3, 1, 2 ], [ 1, 2, 3 ] ], [ [
      3, 1, 2 ], [ 1, 2, 3 ], [ 2, 3, 1 ] ],
  [ [ 3, 1, 2 ], [ 2, 3, 1 ], [ 1, 2, 3 ] ], [ [
      3, 2, 1 ], [ 1, 3, 2 ], [ 2, 1, 3 ] ], [ [
      3, 2, 1 ], [ 2, 1, 3 ], [ 1, 3, 2 ] ] ]
```

Algorithm Overview

Backtracking, a problem-solving technique in computer science, is generally considered a good and powerful approach for exploring multiple possibilities systematically and finding solutions, especially when dealing with constraints and complex problems. This package uses recursive backtracking:

- `LatinSquareList`: Builds full Latin squares.
- `LatinSquareCount`: Quickly counts Latin squares without saving generated squares.
- `LatinSquareRow`: Valid row generation.
- `LatinSquareCountRow`: Counts completions via recursion.

Acknowledgements

Thanks to the GAP community and contributors!

Thank You!

Thank you so much for your attention!

Any questions?

References I

- GAP Group. GAP – Groups, Algorithms, Programming, Version 4.11.1; 2023. https://www.gap-system.org
- Ackoff, R. L. (1953). The design of social research. Chicago: University of Chicago Press.
- Edwards, A. L. (1998). Experimental design. N.Y.: Addison-Wesley.
- Keppel, G. (2006). Introduction to design and analysis. New York: NY: Worth.
- Mason, R. L., Gunst, R. F., & Hess, J. C. (1989). Statistical design and analysis of experiments. New York: NY: Wiley.
- Myers, J. L,. & Well, A. D. (2003). Research design and statistical analysis. Mahwah, NJ: Erlbaum.

References II

- St Andrews University. (n.d.). Choi Seok-jeong. Retrieved from https://mathshistory.st-andrews.ac.uk/Biographies/Choi_Seok-jeong/
- https://github.com/BaseMax/gap-days-2025
- https://github.com/BaseMax/LatinSquareGAP
- https://github.com/BaseMax/LatinSquareGen
- https://github.com/BaseMax/LatinSquareGenerationC
- https://oeis.org/A000315