MAGIC SQUARE SOLVER: T-03, B-01

PSAT MINI PROJECT PRESENTATION

Presented by ADITHYA N S
CB.EN.U4CYS22002
TIFAC-CORE in Cyber Security
Amrita Vishwa Vidyapeetham, Coimbatore Campus

Feb 28, 2023

AMRITA
VISHWA VIDYAPEETHAM

TABLES OF CONTENTS

INTRODUCTION

- Magic Square solver is a tool that is used to find solution for a magic square puzzle
- A magic square puzzle is basically a square grid with distinct el ements where the sum of elements in each row ,column and diagonal is equal.
- This fixed sum is known as the ‘magic’ constant or magic sum.
- A magic square puzzle is a nxn square grid and It bears the integers from 1 to n*n

EXAMPLES

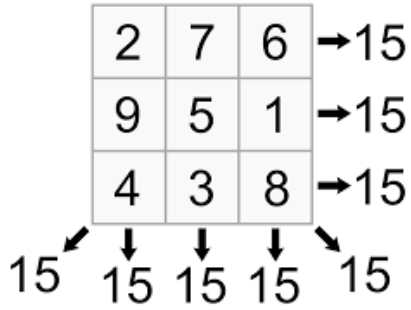

Figure: 3x3 Magic square grid

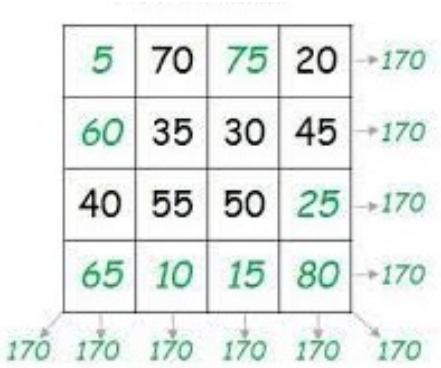

Figure: 4x4 Magic square grid

PROBLEM ANALYSIS

- The problem of solving a magic square is to fill in the cells in the square grid in such a way that the sum of elements in each row, column, diagonal is same.
- This can be achieved through the use of mathematical algorithms and constraints to find the solution.
- We can begin by identifying the size of the square grid and then finding the magic sum.

- To find magic sum, the equation **n(n*n + 1)/2** can be used where n is the size of square grid.
- We can also look for patterns symmetries in the placement of the numbers within the square.
- An important aspect in solving magic square is the understanding of the underlying mathematical and combinational properties. Algebraic and number-theoretic methods are useful for constructing and analyzing the magic squares.

PROBLEM DISSECTION

1. Create a 2D array to represent the magic square.
2. Implement an algorithm for generating all possible solutions for the magic square, such as backtracking or brute force.
3. Compare each generated solution to the original square to check if it is a valid magic square.
4. If a valid solution is found, display the solution to the user. If no solution is found, notify the user that the square is not a magic square.

ALGORITHM

ALGORITHM FOR MAGIC SQUARE SOLVER USING LUX METHOD

1. Create a blank square grid of a certain size.
2. Fill in the center square with the number 1.
3. Move to the next square clockwise, and fill it in with the next number.
4. If the next square is already filled in, move down one square instead.
5. Repeat steps 3 and 4 until all squares are filled in.
6. Check if the square is a magic square by verifying that the sum of each row, column, and diagonal is the same.

OTHER METHODS TO GENERATE A MAGIC SQUARE

A Magic Square can be generated using different methods out of which some are given below.

Siamese method

- Start With an empty square of size n x n, where n is an odd number.
- Place the number 1 in the middle cell of the top row.
- Move one cell to the right. If you are at the end of the row, move to the 1st cell of the next row.If the cell is already filled, move one cell down or vice versa. Fill the cell with the number i.
- Check if the square is a magic square by verifying that the sum of each row, column, and diagonal is the same.This method works for magic squares with odd order.

De la Loubère's method

- Start with the top-left corner of the magic square and fill in the first row and first column with the integers 1 through n*n in ascending order.
- Starting with the second row and second column, fill in the remaining cells of the square by placing the next integer in the cell that is diagonally up and to the right of the current cell. If the next cell is already filled, place the next integer in the cell immediately below the current cell.
- Check if the square is a magic square by verifying that the sum of each row, column, and diagonal is the same.

LIMITATIONS OF USING FLOWGORITHM FOR MAGIC SQUARE SOLVER

- One major drawback of using flowgorithm is that it does not support 2D arrays which puts forward a major challenge.
- Flowgorithm is suitable for a magic square solver that uses small square grids but when it comes to larger squares , it may become a complex task.
- Flowgorithm is a visual programming tool, and debugging can be difficult because it is harder to trace the flow of execution through a flowchart than through a traditional code.
- It is less flexible compared to traditional programming languages which limits its ability to optimize the magic square solver or add new features.

Thank you

THANK YOU!!!

References

- https://www.geeksforgeeks.org/magic-square-even-order/
- https://mathcommunities.org/magic-squares/
- https://byjus.com/maths/magic-square/