KNIGHT'S TOUR PROBLEM

By,
Purushotham Tumuluri - (0726386)
Chandralekha Chavva(1081063)
Upasana Patel(1030860)
Chamundeswari Nune(1080550)

Abstract

The knight's tour problem is the mathematical problem of finding a way for knight to cover all the squares in chess board in a way that each square is covered only once. In this presentation we are concentrating on famous Knight's Tour problem to find the optimal path with count of steps.

Contents:

Introduction

- knight, (also known as horse) is a piece in the game of chess. It does not move in a straight line but makes L-shaped moves to reach an empty square on a chessboard. There are only two legal moves
  - Two squares in the x-direction and one square in the y-direction
  - One square in the x-direction and two squares in the y-direction
- Two types of tours, Closed tour and Open tour

  Closed Tour [1] : Using Knight L shaped moves, if the last move of the knight reaches the first square where the tour started then it is considered as closed tour.

  Open Tour [2] : If the last move of the knight does not traverse the square where tour started then it is considered as open tour.

Figures for Closed and Open tours:

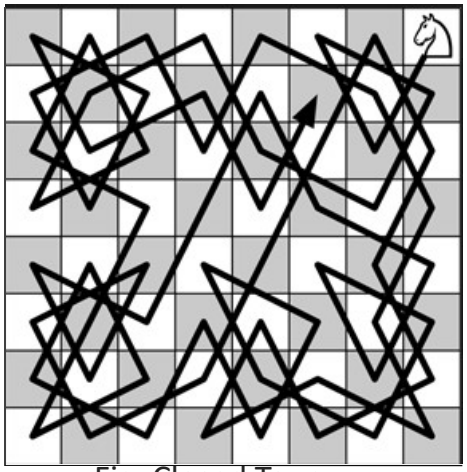


Fig: Closed Tour

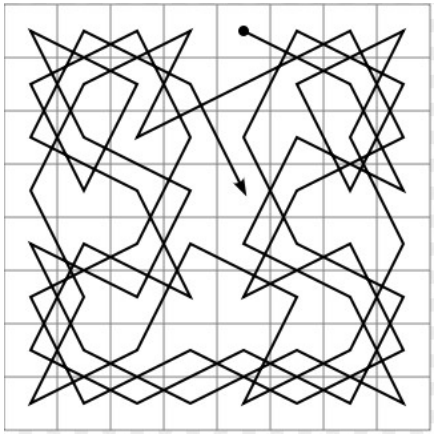


Fig: Open Tour

Tools/Technologies used

- **Technologies:**
  - Java Language
    - Swings
    - Awt Components
  - Operating system: Windows
- **Tools:**
  - Eclipse Mars V2.0

Related Work

- Euler's Method:

  In this, the Knight travels all the squares on a chess board where it visits every square only once and return to the originated square [3][4].

- Warnsdorff's Rule: In this, method Warnsdorff's took below objectives into consideration [5].
  - Always move to an adjoining, un-visited square with minimal degree
  - It is impossible to have three mutually adjacent squares on a chessboard
  - It is impossible for a knight's tour to deviate from Warnsdorff's Rule in the last four moves

The Eulers and Warnsdorff's rule helps the user or the developer to find a better solution

Algorithms:

In this application we have used two types of algorithms:

1. Brute Force
2. Divide and Conquer

Brute Force Algorithm

- Brute Force means, it will go through all possible solutions extensively[6].

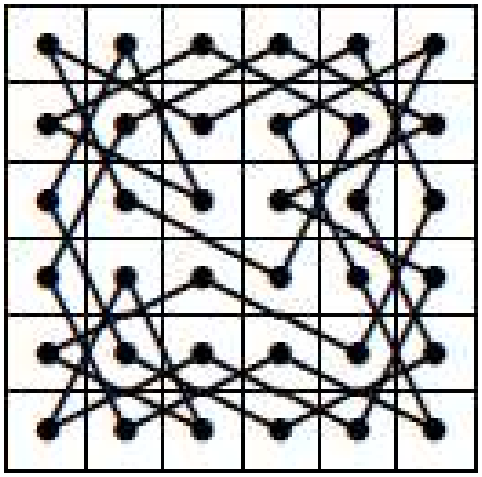
1. Start.
2. If knight at i=1, j=1 position.
3. Moves = [j+1] [i+2] or [j+2] [i+1].
4. Knight move continuous L-shaped moves.
5. Finally if the last move reach the start position closed tour.
6. Otherwise open tour.
7. Count Moves.
8. End.

Example

- This is the open Knight's Tour found by the algorithm in a **5×5 board**, starting from position [0,0] :[6]
  [0,0][1,2][2,4][4,3][3,1][1,0][2,2][0,3][1,1][3,0][4,2][3,4][1,3][0,1][2,0][4,1][3,3]
  [1,4][0,2][2,1][4,0][3,2][4,4][2,3][0,4]

Divided & conquer

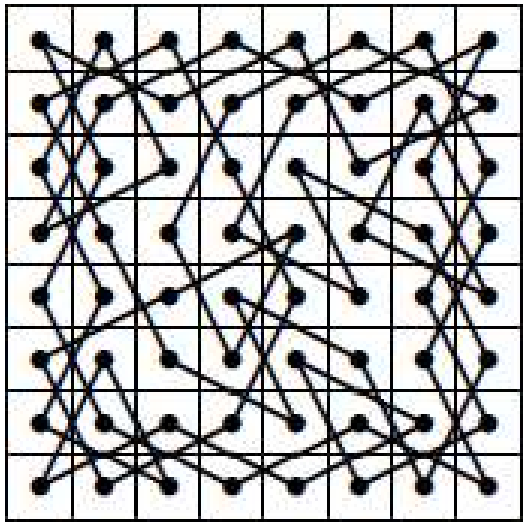
- The following small knight's tour was used as part of the base of the recursion in a divide-and-conquer algorithm described by Ian Parberry, "An Efficient Algorithm for the Knight's Tour Problem",



6x6, Random

Divided & conquer

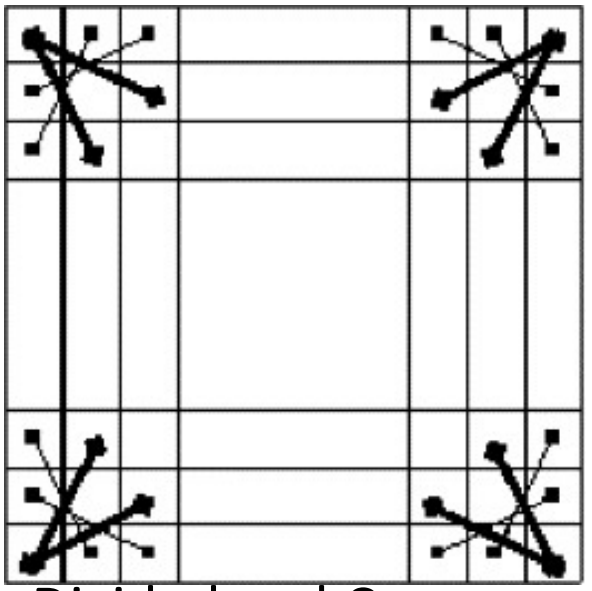
- Parberry presented a divide-and-conquer algorithm that can generate closed knight's tours on n×n or n×(n+2) boards in linear time (i.e. $O(n^2)$) for all even n and$n \geqslant 10$, and closed knight's tours missing one corner in linear time if n is odd and greater than 4 [7].



8x8, Random

Divided & conquer

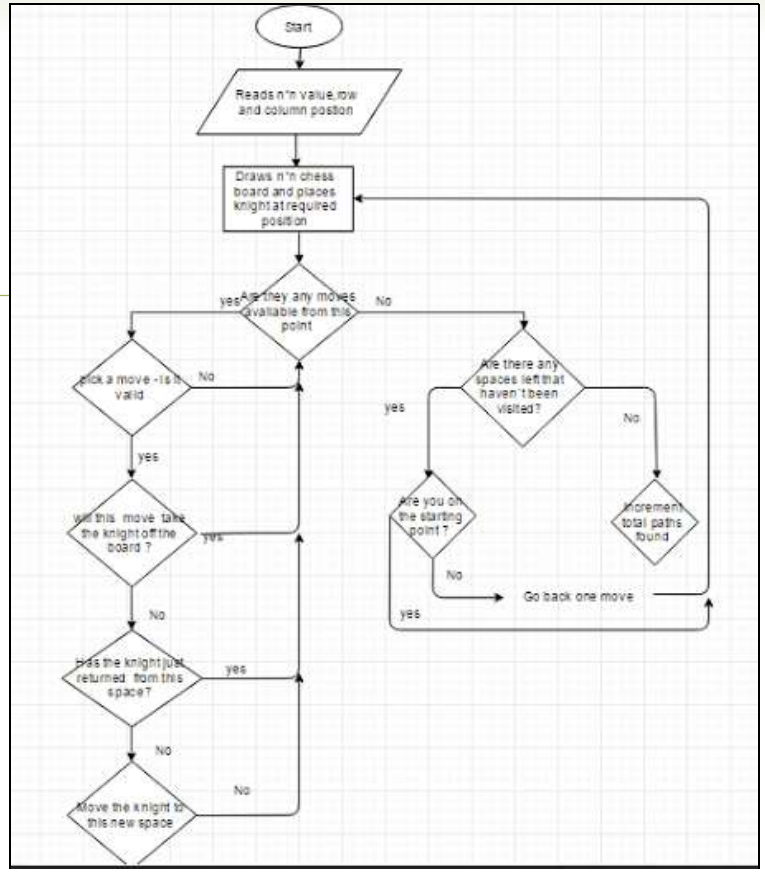
1. Start
2. Divide the chessboard into more than two halves.
3. Solve individual chessboards by Knights Legal Moves (L -shaped).
4. Combine all individual boards.
5. Final Graph is drawn by linking each move.
6. End



Divided and Conquer Graph [8]

Work Flow Diagrams

This is our work flow diagram this illustrates the work flow in our project.



Flow Diagram of The Process [9]

Execution

- This is the Window that pops when we Run the project.
- 3 drop down Boxes
  - The 1st drop down box allow us to select the size of the chess board
  - The 2nd, 3rd dropdown boxes represents the columns and rows.

  We display Count of moves made by Knight in the window when it covers the squares.

Screen appeared once the code is executed

Cont..

- 4 buttons
  - Draw button to draw the chess board
  - Play button start the tour
  - Solution button gives us the solution for the tour
  - Finally stop button pauses the trip.

Cont..

- we selected a chess board of size 5 and we placed the knight at column ‘a’ and row ‘1’ as shown.

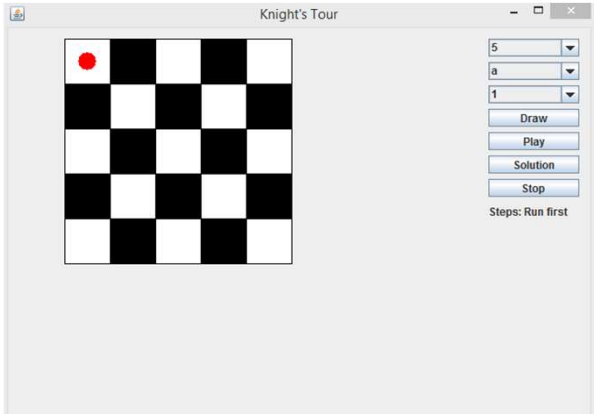


Figure : Screen appeared once the above option were selected and drawn

Few Solution Screens

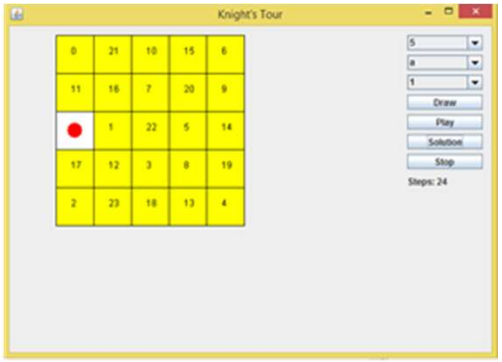


Figure : Solution for 5*5 board

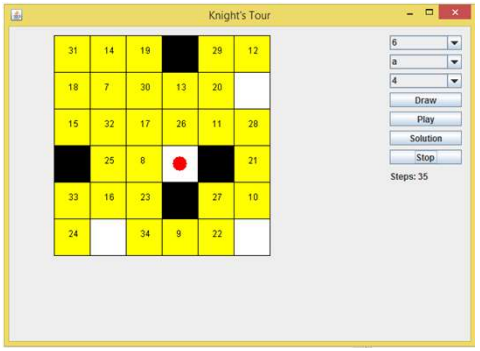


Figure : The tour of the knight is stopped in between the tour.

No Complete Tour

- we found that our knight does not complete the tour by covering all the squares. For instance on a 5*5 chess board when a knight is placed at column c and row 2 the knight starts it tour and reaches the maximum legal moves and stops at the corner of the chess board as seen in the figure, the knight was not able to make any legal moves further and stops.

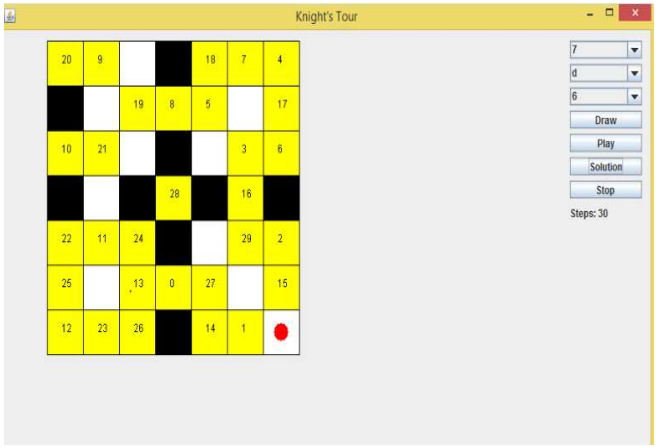


Figure: No complete tour on 7*7 when knight is placed at column d and row 6

Conclusion

- In our work we mainly concentrated on finding a solution for the knight mathematical tour problem using various methods.
- It was helped us in understanding various algorithms and researches made on this problem.
- We also conclude that Warnsdorffs rule helped us to find the optimal solution.

Future Work

- We are planning to use more simple and effective algorithms to find a best solution for the problem.
- There are few other cases that the knight not able to perform the complete tour we are planning to find the solution for it.

References

- [1] J. Erde, “The closed knight tour problem in higher dimensions,”25-Oct-2012.[Online]. Available at: http://www.combinatorics.org/ojs/index.php/eljc/article/viewFile/v19i4p9/pdf
- [2] C. N., “Can you Solve the Knight's Tour Math Problem?,” YouTube, May-2012. [Online]. Available at: https://www.youtube.com/watch?v=9fSFC00ZKPg [Accessed: 08-May-2016].
- [3] “Introducing Knight's Tours,” Introducing Knight's Tours.[Online].Availableat:http://www.mayhematics.com/t/1n.htm
- [4] The Knight’s Tour by Colleen Raimondi http://academics.smcvt.edu/jellis-monaghan/combo2/Archive/Combo%20s03/special%20topics%2003/The%20Knights%20Tour.ppt
- [5] A NEW ALGORITHM FOR KNIGHT’S TOURS by SAM GANZFRIED [Online]. Available at : https://www.cs.cmu.edu/~sganzfri/Knights_REU04.pdf

References

- [6] “Knight’s Tour brute force algorithm example,” SIGQUIT, Dec-2010. [Online]. Available at: https://sigquit.wordpress.com/2010/01/13/knights-tour-brute-force-algorithm-example/
- [7] “An efficient algorithm for the Knight’s tour problem” by Ian Parberry. [Online]. Available at: https://pdfs.semanticscholar.org/927e/dbaa256301f500e8b3fcd52eaa6f0cc2c768.pdf.
- [8] “The Knight’s Tour,” Gordon Tumilty, 2010. [Online]. Available at: https://gtumilty.wordpress.com/2010/02/25/the-knights-tour/
- [9] “Optimal algorithms for constructing knight's tours on arbitrary n×mn×m chessboards,” Optimal algorithms for constructing knight's tours on arbitrary n×m chessboards. http://www.sciencedirect.com/science/article/pii/S0166218X04003488
- [11] “Knight's tour,” Wikipedia. Available at: https://en.wikipedia.org/wiki/Knight%27s_tour