Assignment - 5 (A5)

Problem definition:
Write a C++ program to draw a 4x4 chessboard rotated 45° with the horizontal axes. Use bresenhams algorithm to draw all lines. Use seed fill algorithm to ~~draw~~ ~~all~~ ~~lines~~. fill the squares of the Chessboard

Objective: To be able to draw a chessboard & apply transformation to it. Learn seed fill algorithm.

Outcomes: Will be able to implement various transformation and be able to fill any polygon.

H/w & s/w requirements:
- 64 bit OS
- Qt creator

Theory: Seed filling algorithms are those that uses a point inside polygon to fill.
- Seed filling algorithms can be classified into
  - Flood fill
  - Boundary fill

To detect if a point lies inside a polygon, we have inside checking algorithm/

- tests such as even-odd test

1) Flood-fill - In this algorithm, we start filling from a point inside the polygon
- If the ~~point~~ is pixel colour matches the background colour, we fill with fill colour else we dont.
- This is a recursive algorithm. Once the current pixel is filled, its neighbours are checked & filled.

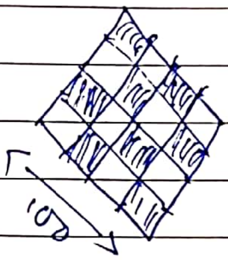
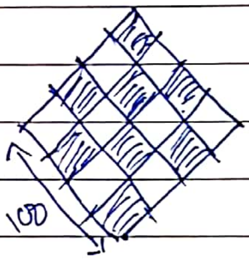
2) Boundary fill :
- If the pixel we want to fill, has the same colour as boundary, we skip it else we fill it.
- If a pixel is filled, we try to fill its neighbours.

Advantages :
- Algorithms of seed filling are simple
- Take very few lines of code.
- Easy to implement due to recursive definition.

Disadvantages:
- Point inside polygon must be known.
- Recursion may cause stack overflow.
- If using boundary fill, boundary colour must be know
- Multi-colour boundary fill is difficult

and complicated to implement.

Test cases :

Input	expected o/p	actual o/p	result.
side of square board 100			pass

Conclusion :

Thus, using transformations and seed filling algorithms, we were able to draw a rotated chessboard & fill it.