Computer Graphics Project

2D Chess using OpenGL

Rhitwik Saha (18075048)

Open GL

- OpenGL(Open Graphics Library) is a standard specification defining a cross language cross platform API for writing applications that produce 2D and 3D computer graphics.
- The interface consists of over 250 different function calls which can be used to draw complex 3D scenes from simple primitives.

Project Goal

- The aim of the project is to successfully implement all the functionalities of a 2D chess game, using OpenGL.
- The pieces have been formed using primitive shapes at varying scales.

Scope

- The Project is developed in Sublime Text IDE. It has been implemented on Ubuntu Platform.
- The 2D graphics package designed here provides an interface for the users for handling the display of chessboard and different chess pieces.
- The mouse is the main and only input device used.

Requirement Specifications

HARDWARE SPECIFICATION:

- 4GB of RAM
- 150MB of hard disk space required (for installation of Software)

SOFTWARE REQUIREMENT:

- Development Tool: LINUX(Ubuntu 18.04 LTS)
- Language: C/C++
- IDE: Sublime Text
- Library: OpenGL

Basic Design

- The code comprises of two section:

  Graphic UI: This renders with the graphical user interface, of the chessboard and the chess piece along with a display board on the right.

  Logic: This serves as the logic behind the possible moves that a chess piece can perform, taking cases for check, illegal moves and checkmate.

Design(1)

- I have incorporated several inbuilt OpenGL functions in the Graphic UI file.
- Header files used:

#include<stdio.h> : This is C library for standard input and output and display, and general purpose functions.

#include<GL/glut.h> : This header is included to read the glut.h, gl.h, glu.h.

#include<math.h> : This is a C library used for performing certain mathematical operations.

#include<string.h> : This is a C library used to perform manipulation on character array.

#include<vector> : This is a C++ library used to perform manipulation on vector arrays, that I have used in the logic part of the project.

Design(2)

- The following functions have been used which enables the rendering of chess pieces and chessboard with different specific colors and the display box.

  glColor3ub(...) : sets the current color (takes rgb values)
  glVertex2i(...) : specifies a vertex (takes x,y coordinate)
  glBegin(...) and glEnd(...) : delimits the vertices between them.
  glLineWidth(...) : specifies the width of rasterized lines
  glPointSize(...) : specifies the diameter of rasterized points.
  glRasterPos2f(...) : specifies the bottom left position of text.
  glutBitmapCharacter(...) : displays the character.

Design(3)

- In the logic portion of the project, I have implemented several method functions for each chess piece.
- It includes all the current possible moves for the current player for all his chess pieces.
- I have also implemented the code for check and checkmate conditions for every chess piece.

User Interface

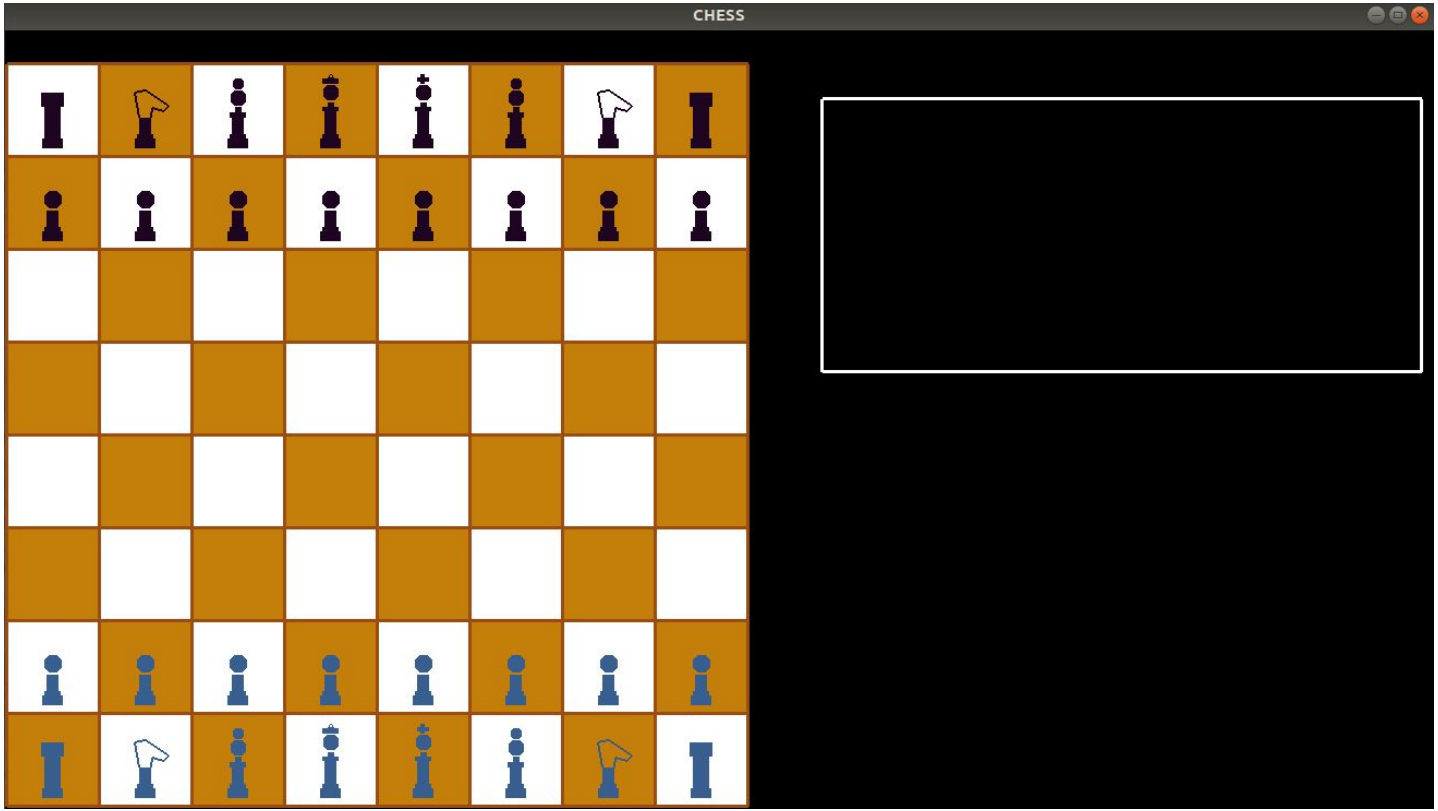
- The mouse is the major and only user input used here:

  Left Button: Allows the user to select a chess piece and chess square.

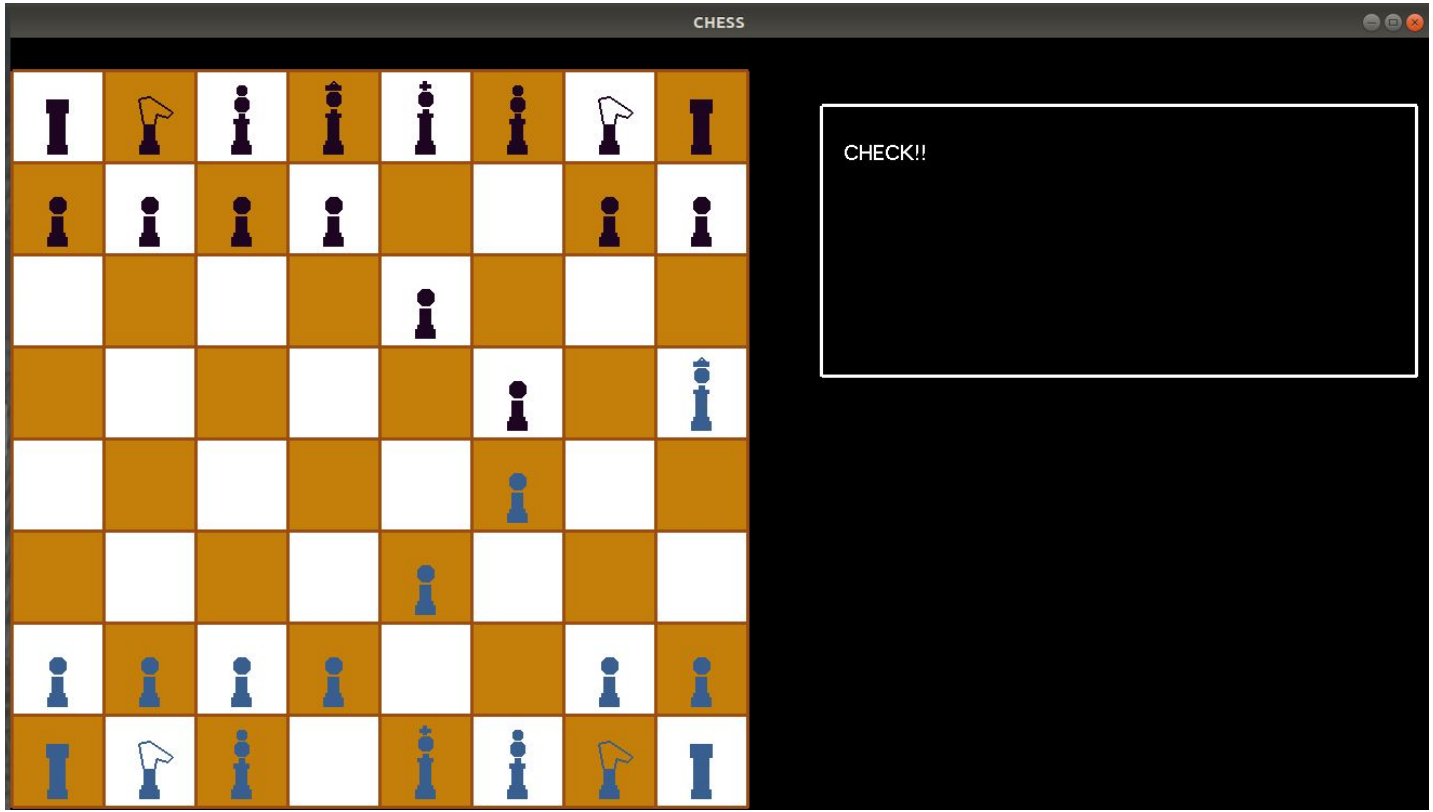
  Right Button: Allows the user to undo move.

SnapShots

- **Snapshot 1:** In this snapshot the initial structure and arrangement of the chess pieces are shown along with a display box on the right which will show the alerts during the game.
- **Snapshot 2:** in this snapshot an in-game screenshot has been shown where BLACK king is CHECKED with WHITE queen, which is displayed on the display box on the right.



Snapshot 1



Snapshot 2

Future Advancements

- 3D implementation of this game can be done, with more advanced structure formation of the chess pieces, using primitive shapes like, cones, spheres, cuboids, etc.
- AI can be introduced in this which will play according to designated difficulty.

THANK YOU!