CS101 PROJECT REPORT

(YEAR : 2015)

CHAIN REACTION

Team ID -231

GROUP MEMBER	ROLL NO.
1) PUSHKARAJ DHAKE	140070019
2) MOHAMMED RAFI	140070049
3) GAURAV SINHA	140040082
4) KARTHIK MADDELA	140040102

Table Of Contents

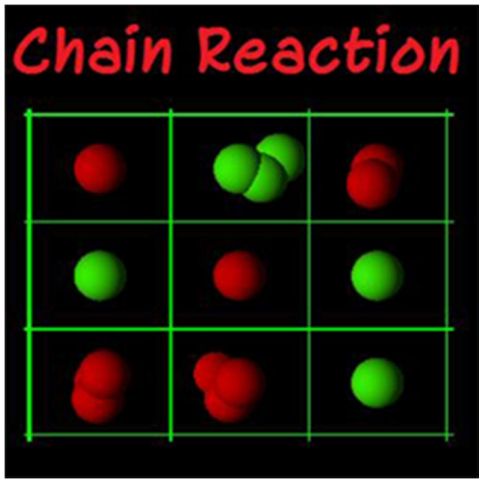
1.INTRODUCTION:

Now-a-days, there are lot of games available in the online world. As a kid everyone is fascinated with the video games they see while growing up and thinks of making one such game.

We also tried to make an entertaining and recreational game "**CHAIN-REACTION**", using all the concepts that we got to learn from "CS 101" course, which also involves the use of your logical skills to defeat your opponents.

This was the basic motivation of us behind choosing this as our project as the game is fascinating as well as sharpens the mind of the players playing it.

This is a strategy game for 2 to 4 players. The objective of Chain Reaction is to take control of the board by eliminating your opponents' balls.



2. PROBLEM STATERMENT:

- The aim of the project is to develop a game known as **"CHAIN REACTION".**

- The game should follow all its rules and regulations.(refer to the RULES)

- ***RULES OF THE GAME :-***

1. WITH THE HELP OF MOUSE CLICK ON THE REQUIRED CELL PLAYER CAN PLACE BALL OF HIS COLOUR ON THE BOARD.
2. THE PLAYER CAN CLICK ONLY ON THE CELL CONTAINING SAME COLOUR BALL OR ON AN EMPTY CELL WITH NO BALLS IN IT.
3. THERE IS A LIMIT ON THE MAXIMUM NUMBER OF BALLS THAT CAN BE CONTAINED BY A CELL AND IF THE NUMBER OF BALLS IN A PARTICULAR CELL EXCEED THAT LIMIT THEN EXPLOSION TAKES PLACE.
4. WHILE EXPLODING IF ANOTHER PLAYER'S BALLL IS PRESENT IN THE ADJACENT CELL IT ALSO BECOMES THE COLOUR OF THE PLAYER WHO HAS DONE EXPLOSION.
5. WHILE EXPLODING IF ADJACENT CELL ALSO CONTAINS MAXIMUM NUMBER OF BALLS THEN AFTER ADDITION OF BALL FROM EXPLODING CELL IT ALSO EXPLODES AND THIS CHAIN FURTHER CONTINUES.
6. IN THIS WAY THE PLAYER HAS TO ELIMINATE THE OPPONENTS'S BALLS AND HAS TO TAKE CONTROL OF THE BOARD.

3. REQUIREMENTS:

- **Hardware Requirements :**

  - **A PC or Laptop :**The software shall display information to the user on the monitor screen of the PC or Laptop.

  - **Mouse** : The software shall interact with the movement of the mouse and the mouse buttons. The mouse shall activate areas for data input, and select options from menus.

- **Software Requirements :**

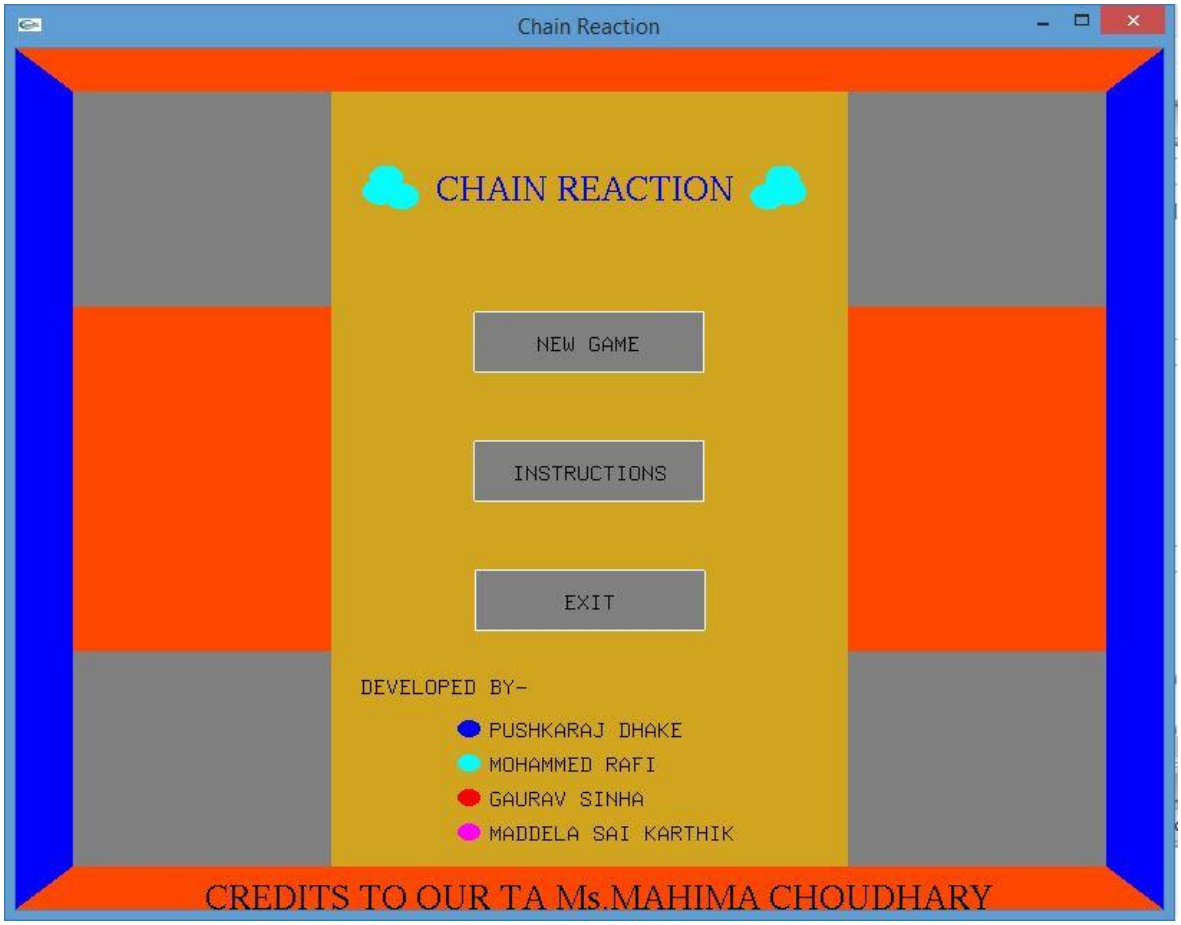
  - **Codeblocks (SVN Build)** : To compile the written code.
  - **OpenGL graphics library** : It should be included in the code as a header file. This is the graphics package included by us while developing this game.

4. IMPLEMENTATION:-

- **Functionality:**

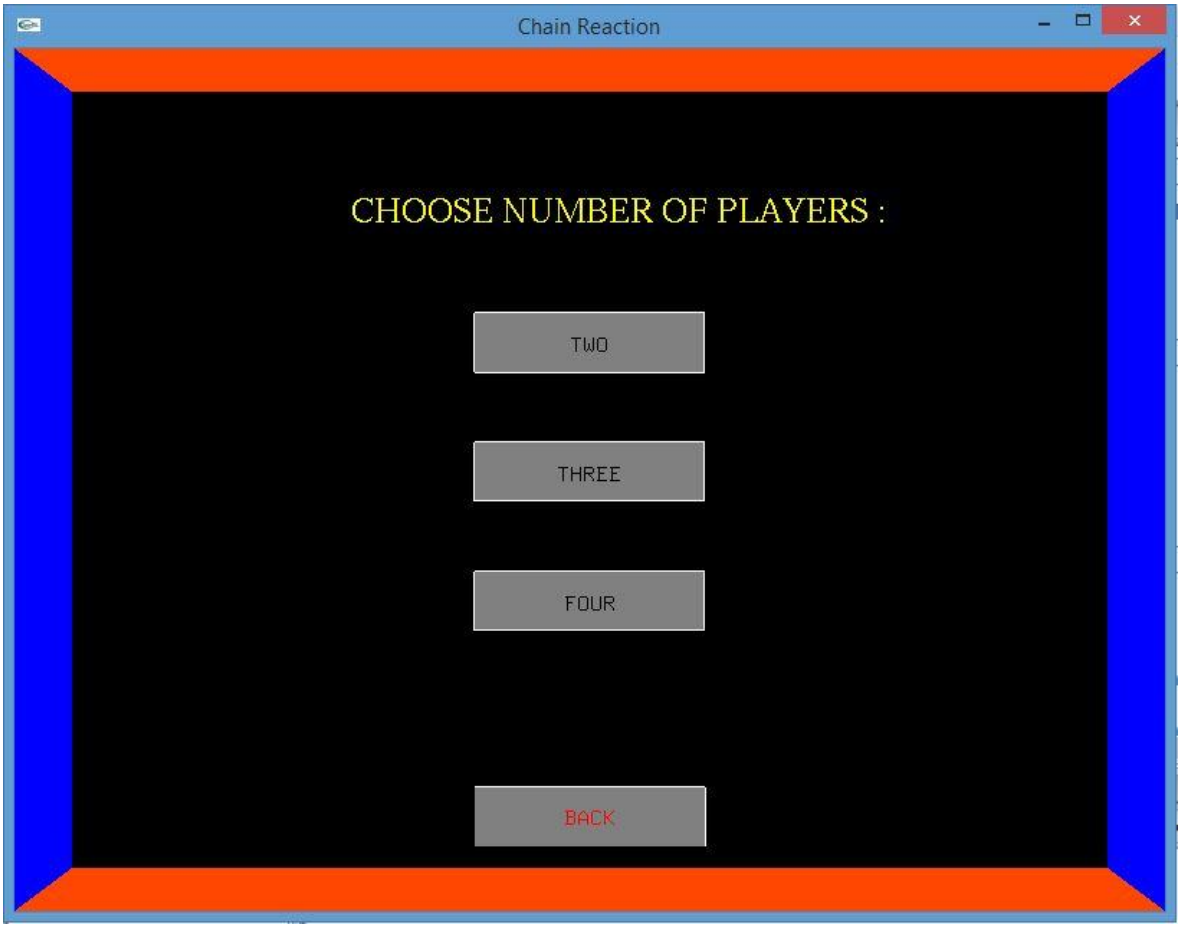
  After running the game an initial window will appear which will contain 3 buttons.

  - ***Initial Window :***



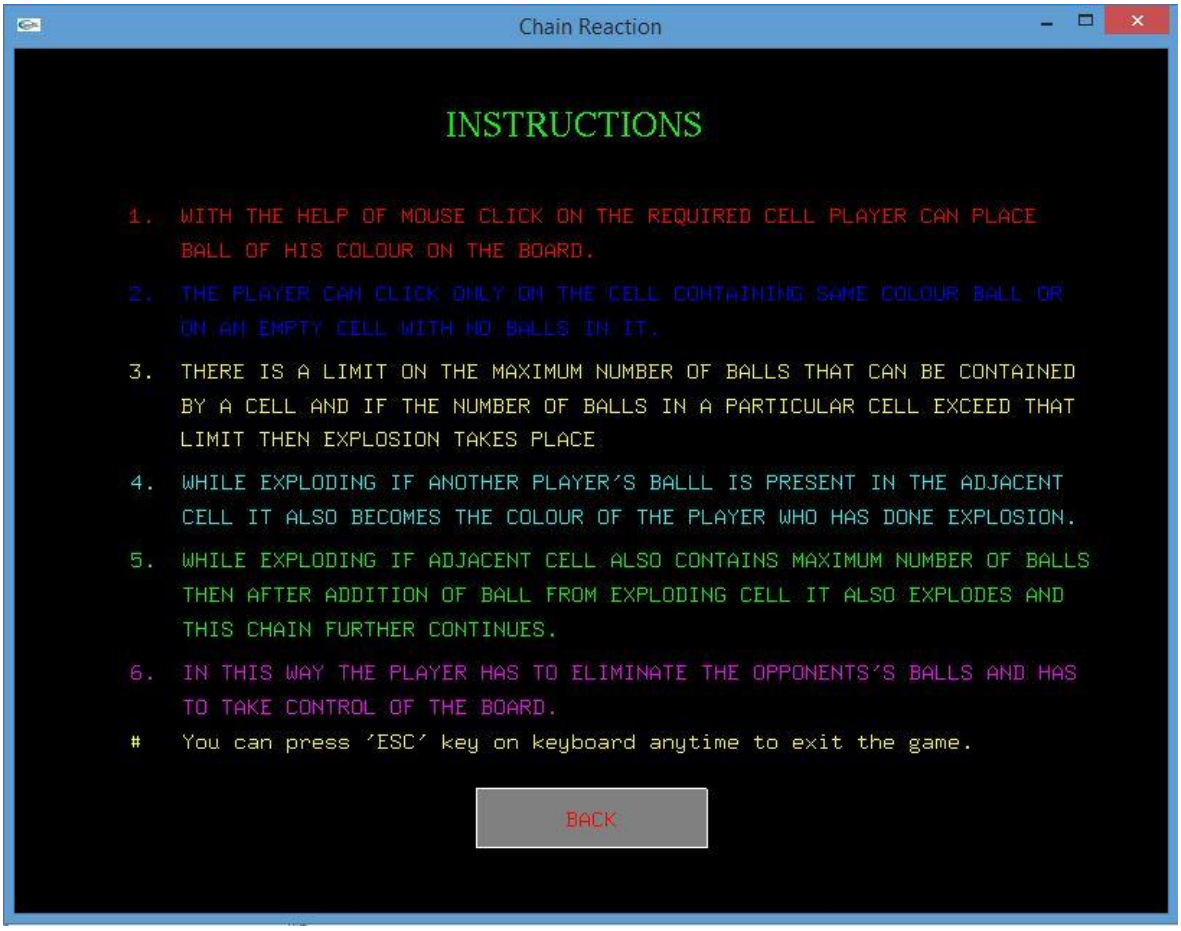
1. ***NEW GAME:*** If you click on this button, another window will appear which will ask for choosing no. of players (2 to 4).You can choose no. of players by clicking on the button showing respective number.

- **Window to choose no. of players :**



2. ***INSTRUCTIONS*:** You can click on this button if you don't know how to play this game. After clicking a window showing instructions to play game will appear.

- **Instructions window :**



3. **EXIT:** You can exit the application by clicking on this button.

- Now, if you have chosen no. of players, game will be started. Initially a grid will appear in a new window. A text will be shown above the grid telling whose turn it is to play. The colour of the grid will show the respective colour of player whose turn it is.

- A player whose turn it is to play can place his in a particular cell in the grid ball by taking the mouse pointer on corresponding cell on the grid and clicking.

- A player can place a ball only in empty cells or the cells containing his balls.

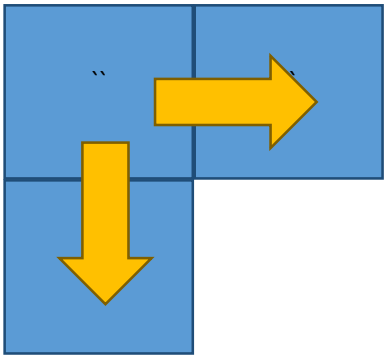
- If a player places a ball where his ball is already present, one ball gets added to that cell of his colour.

- There is a limit to maximum balls that a cell can occupy.

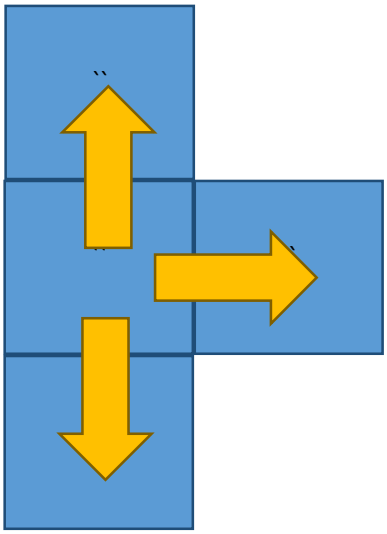
- Cells on the corners of the grid can occupy 1 ball, cells on the sides but not corners of the grid can occupy 2 balls and rest of the cells can occupy max. 3 balls.

- If a cell has maximum no. of balls it can occupy and the player clicks one more time into the same cell to place a ball, the balls in the cell explode in all possible directions where another cell is present with one ball going into the adjacent cell in every direction.

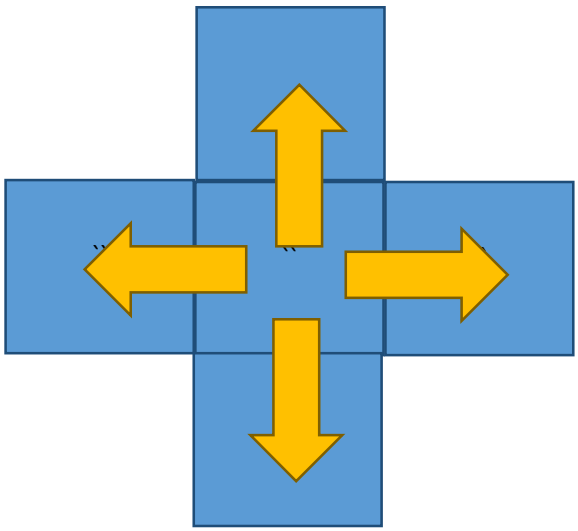
- ***If the exploding ball is at top left corner it will explode with one ball in each direction as shown below :***



- ***If the exploding ball is in the cell which is on the left edge of the grid but not corner,it will explode with one ball in each direction as shown below :***



- ***If the exploding ball is in the cell which is neither in cell at the edge nor in the corner cell, it will explode with one ball in each direction as shown below :***



- While exploding if another player's ball is present in adjacent cell, it also becomes of the colour of the player who has done explosion with addition of one ball from exploding cell to it.

- While exploding if the adjacent cell also contains max. no. of balls it can have, after addition of the ball from exploding cell it also explodes and this chain continues till the final exploding cell has all its adjacent cells with less no. of balls than max. they can have.

- If no balls of a player are present on the grid, he is eliminated.

- The game ends when all the balls on the grid belong to same player and that player is the winner.

- In between the running game, if the player wants to leave the current game and start a new game, there is a button below the grid as “LEAVE THE GAME & GO TO MAIN MENU” clicking on which the current game will get destroyed and the initial window will appear with 3 buttons (New game, Instructions, Exit).

- If some player has won a window showing his win will appear. It will also contain two buttons as :

1. ***MAIN MENU*****:** *Clicking on this will take you to the initial window.*

2. ***EXIT*** **:** *Clicking on this will exit the application*

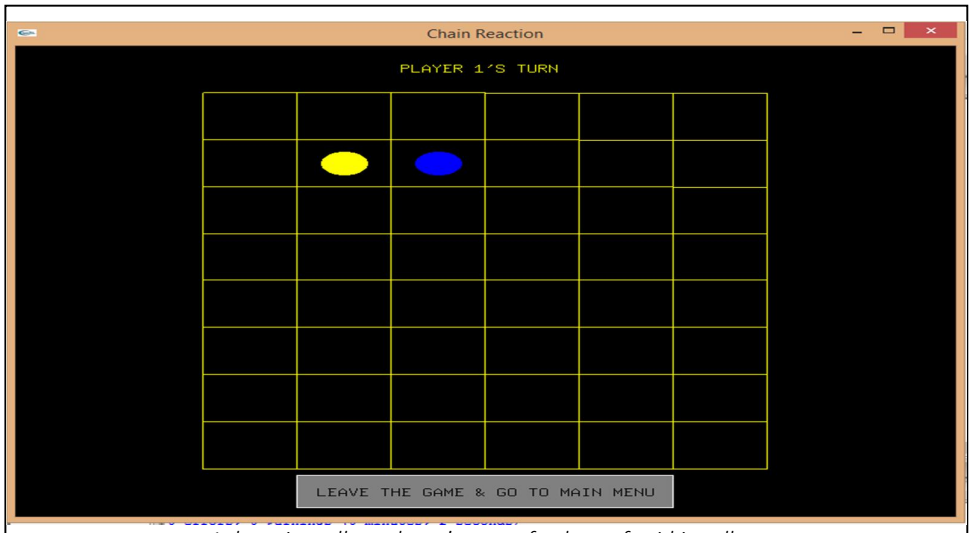
- **Algorithm :**

  - We have used OpenGL graphics library for GUI.
  - We have switched the windows by switching a variables value after every click of user after which a new window is expected to be shown.
  - We have defined few matrices to keep update of no. of balls and colour of balls in each of the cells. One constant matrix which has the values of max. no. of balls in each cells.
  - We have written functions which rearrange the values in this matrices is after every player has played his turn.
  - We have also written functions to check if some player has eliminated and to show win when someone wins at the end and also to initialize everything of previous game if user wants to play a new game.

5. TESTING STRATEGY & DATA:

- ***The game takes input from mouse click coordinates and responds correspondingly.***

1. **CHANGE IN COLOUR OF GRID** : The color of grid should change according to player.

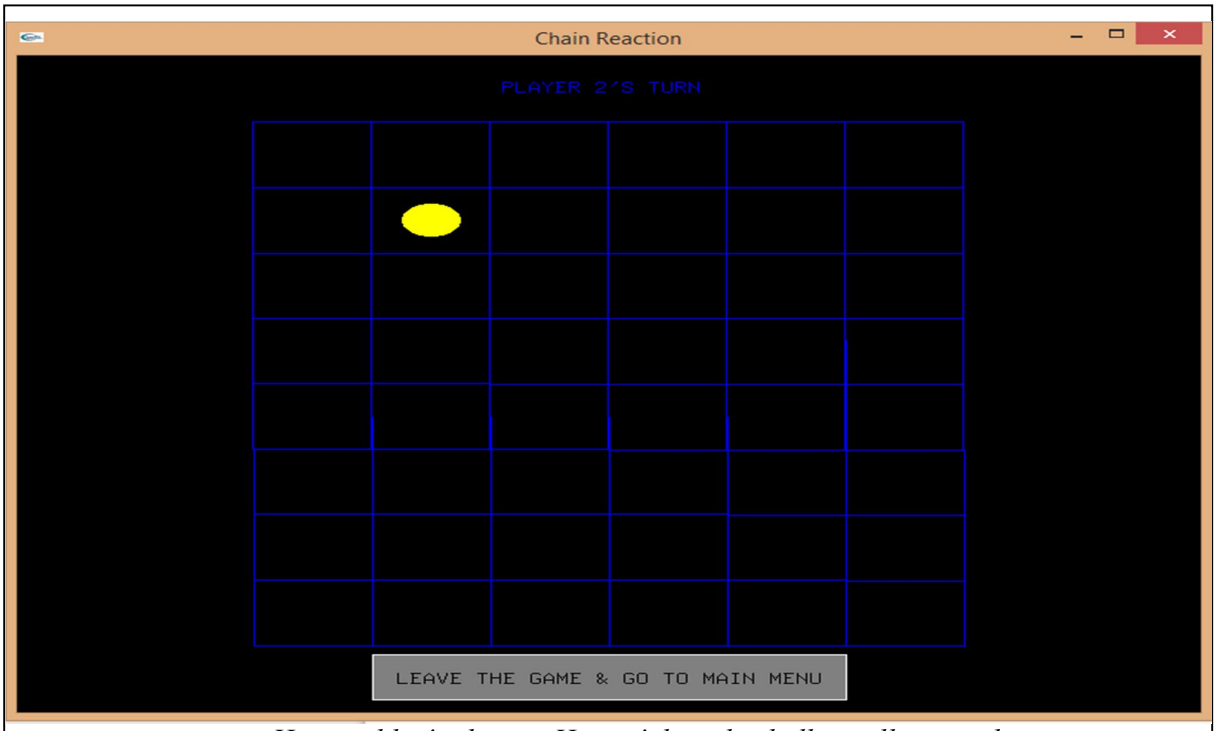


1: here its yellow player's turn of colour of grid is yellow

2. **NOT TO TAKE UNNNECESSARY MOUSE CLICKS AS INPUTS :`**

- It should not respond to unnecessary inputs of mouse clicks.

- Player-y can't keep his balls in player-x's boxes.



. *Here its blue's chance. He can't keep his ball in yellows grid.*

3. **EXLODING CRRECTLY AND CHANGE COLOUR OF BALLS IN ADJACENT CELLS:** If player-x's ball explodes into players-y's balls, then the balls of player-y are changed to player-x's balls.

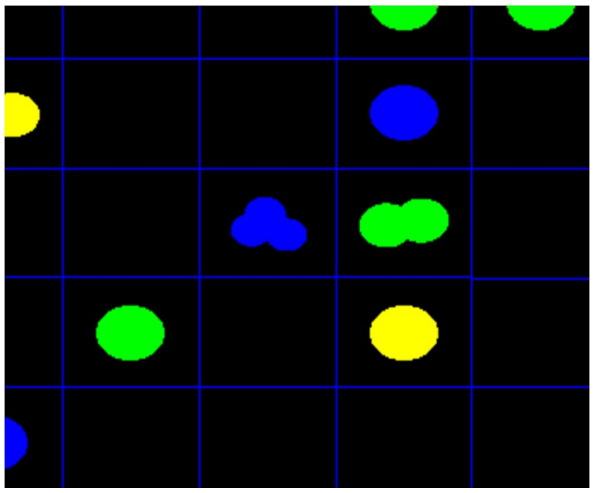


figure.3

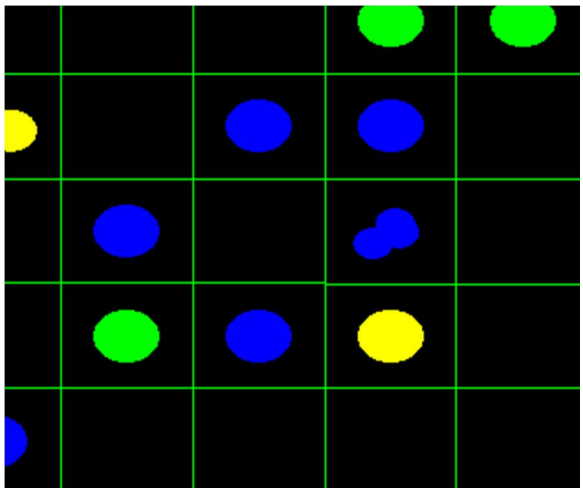
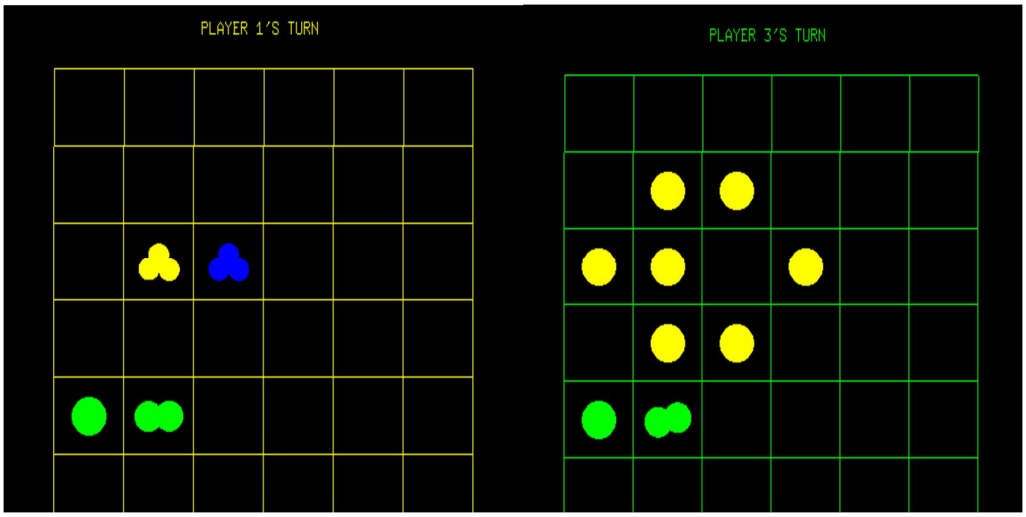


figure.4

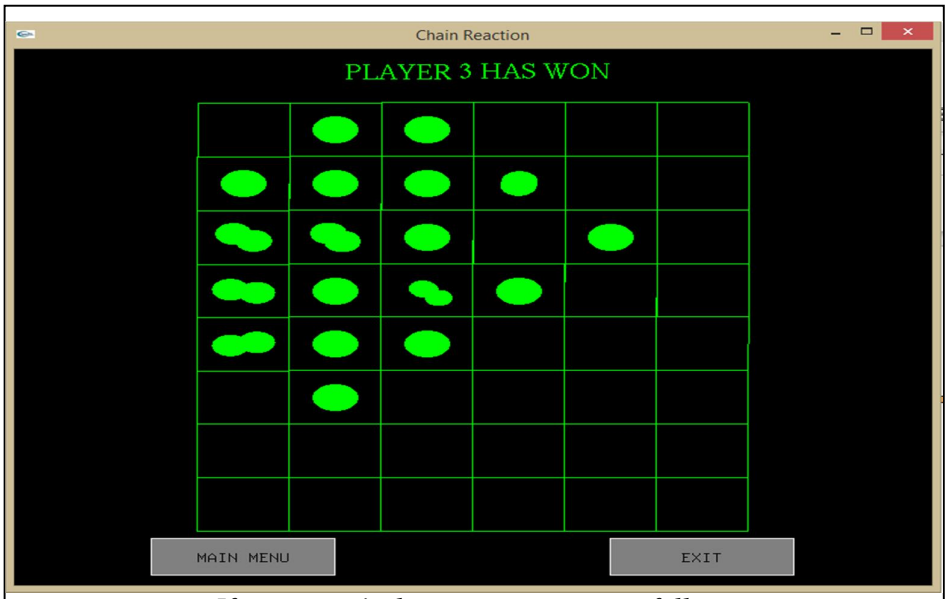
✓ In figure.3 if the blue player clicks on the cell where three balls are already placed then after explosion color of the balls in adjacent cell will also change (figure.4)

4. **ELIMINATION AFTER LOOSING OF ANY PLAYER :** Player should be eliminated after he loses game (necessary in multi-player).



- ✓ In Fig.1 screenshot is taken during a three player and yellow player's chance. If he clicks on yellow balls then the result will be as follows.
- ✓ Blue player lost the game after yellow's chance so he is removed from the game yellow balls then the result will be as follows.

5. **SHOWING WIN :** Should show win if only one player remains and shouldn't take input until new game is started.



If green win's the game output is as follows

6 . DISCUSSION OF SYSTEM :

A) What are worked as per plan?

- ***OpenGL Library:-***

We decided to do the graphics part of our project in this library and it worked out well for us. The overall design of the game was up to the mark and according to our expectations.

- ***Multiplayer Game:-***

This game gives the user options to select the number of players playing the game. This was essential for increasing fun quotient of the game and to increase user's involvement it.

- ***Indication of chance of the player with visual effects:-***

The colour of the grid shows which players' chance it is along with the display of text at the top of the window.

B) What we added more than discussed in SRS?

- ***Rotation of the balls placed in the grid:-***

. When two or three balls are placed together in a cell their combination starts rotating. This added feature enhances the visual effects of the game making it more fascinating for the users.

- ***Renewal of the Game:-***

When a game is finished and the winner is declared, the user gets an option for "main menu" and "exit". The user can continue to play a new game or exit depending on his choice without having to compile the code again to enjoy the game.

- ***Keyboard input for exit:-***

Anytime in the game if user wants to exit he may exit the game by pressing the escape key. This further enhances the user interface.

7. PROJECT COMPLETION TIMELINE :

- WEEK 1 (03/03/15 to 09/03/15) :-

  - **TASK**:-

    ✓ Deciding the project and understanding it.
    ✓ Making SRS.

  - **COURSE OF ACTION:-**

    ✓ We discussed many ideas and sample projects and finally decided to make the game CHAIN REACTION in C++ in OpenGL graphics Library.
    ✓ According to the given standard template, we made the SRS of our project.
    ✓

- WEEK 2 (10/03/15 to 16/03/15) :-

  - **TASK:-**

    ✓ Learning about various functions and commands in OpenGL graphics library that we would be using in our project by reading various online tutorials and websites.
    ✓ To get a rough idea on paper how would we proceed to complete our project.

- **COURSE OF ACTION:-**

  - ✓ We all studied about various functions in OpenGL from various online resources available and discussed among ourselves.
  - ✓ We discussed among ourselves about the basic mathematical functions that we would be requiring to complete our project.
  - ✓

- WEEK 3 (17/03/15 to 23/03/15) :-

  - **TASK:-**

    - ✓ To draw the grid using openGL graphics library.
    - ✓ To write the basic mathematical functions that we would be requiring to update the data variables while executing the code of our project.
    - ✓ To get the mouse-click co-ordinates and placing the spheres(balls) accordingly.

  - **COURSE OF ACTION:-**

    - ✓ We searched for the code about drawing a line in OpenGL and using that as the basic idea we wrote the code for drawing our grid.
    - ✓ We declared the basic data variables that we would be using in our code like the matrices such as " max_balls[8][6] ball_count[8][6] , ball_colour[8][6]" and the mathematical function "rearrange" which rearranges the ball after a explosion.

- ✓ We wrote a function to place the spheres in that cell of the grid where the mouse is clicked according to the condition of that cell.

- WEEK 4 (24/03/15 to 30/03/15):-

  - **TASK:-**

    - ✓ To get the spheres exploding according to the rules of the game.
    - ✓ To get the grid color change according to the player's chance and also to display the turn in form of text.
    - ✓ To get our prototype ready with a 2-player game.

  - **COURSE OF ACTION:-**

    - ✓ We showed the explosion of the balls by redisplaying everything on the window with the rearranged spheres using "glutPostRedisplay" function in OpenGL.
    - ✓ Again by using the same function we were able to change the color of the grid after every chance indicating player's chance and got the text of it displayed.
    - ✓ In this way we were able to complete our two player game prototype without being able to display the winner.

- WEEK 5 (31/03/15 to 06/04/15) :-

  - **TASK:-**

    ✓ To make the multiplayer version of the game.
    ✓ To be able to eliminate the lost players and declare the winner at the end.

  - **COURSE OF ACTION:-**

    ✓ We made changes in our functions accordingly to make the game multiplayer by extending the values of few variables to the maximum number of players.
    ✓ We defined two functions "eliminate" and "win" respectively to be able to eliminate the lost player in multiplayer game and to show the winner at the end of the game.

- WEEK 6 & LATER TILL SUBMISSION (07/03/15 to 18/03/15) :-

  - **TASK:-**

    ✓ To add windows to show initial main menu, instructions, choose the number of players playing the game having certain buttons by clicking on them user can give his input.
    ✓ To make the visual effects of our game better by modifying the graphics part of our game.

- **COURSE OF ACTION:-**

  - ✓ We used a variable "window" and made different switch cases to display different windows with the buttons in our game.
  - ✓ We added the rotation of spheres about an axis in the plane of the grid to give it a 3D touch so that our game looks better and more realistic.

8. CHALLENGES & MITIGATIONS:

- ***Mouse click function:-***

The input of our game is mostly carried out with the mouse click. We had to include that functionality in our code and it was challenging task to get the syntax for this function. The co-ordinates that we were getting through this function was not according to our window co-ordinates so we had to convert it accordingly.

- ✓ Through extensive study of various tutorials available on the internet, we got the syntax for writing mouse click function.

- ✓ We used an expression for further converting the obtained co-ordinates through this function to our desired window co-ordinates.

- ***Eliminate and end check(for removing the lost players):-***

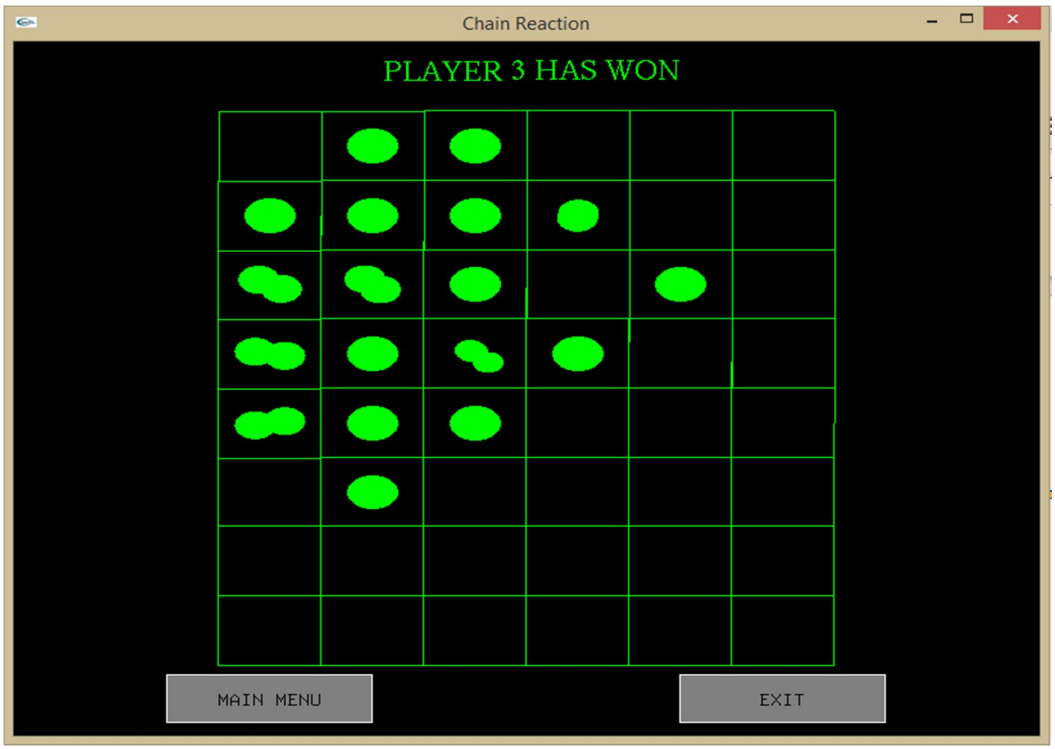
  In a multiplayer game like a three player game, if one of the player looses all his colour balls then his chance should not come next time and the game should continue between the remaining two players. Moreover, when a single colour ball is present in the entire grid (except at the beginning of the game), the declaration of the winner along with no further placing of the balls by any player must occur.

- ✓ We defined two functions accordingly i.e. "eliminate" and "win" to satisfy these conditions and fulfil our requirements. "Eliminate" function deals with elimination of players from a multiplayer game so that a lost player does not get further chance to place his ball in the grid.
- ✓ "Win" function declares the winner of the game when only a single colour ball is present in the entire cell.

- ***Renewal of the Game:-***

When a game is finished and the winner is declared the user gets an option for “main menu” and "exit". The user can continue to play a new game or exit depending on his choice without having to compile the code again.

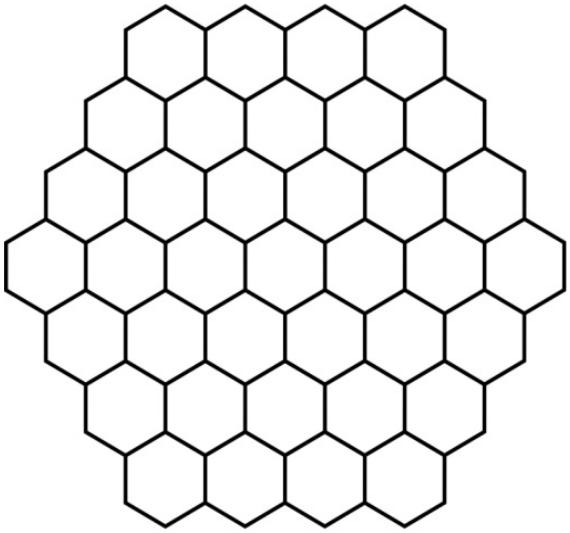
- ✓ This was achieved by again initializing all the variables through a function call so as to make the gaming experience uninterrupted for our users.



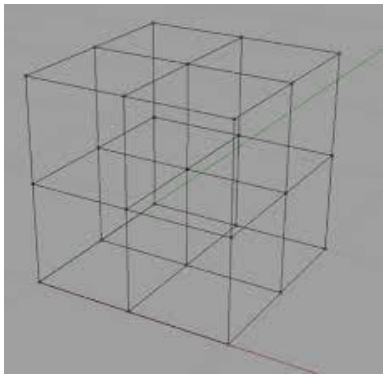
In fig.5, main menu button option gives the users option of restarting the game again

9. FUTURE WORK :

1. We have used square cells in this game. This game can be extended to different shaped cells (like pentagon or hexagons) by which no. directions to explode the balls can be increased.



2. This game can also be extended to 3-D where the grid will also be 3-D and the balls will explode in adjacent cells in 3-D.



3. This game can be extended to be played by only one player with whom computer will play (adding artificial intelligence).

4. This game can be extended by adding one more feature ''time'', so that if someone wants to play the game till certain time, and after that much time the one who has the most balls on the grid will win.

10. YOUTUBE VIDEO LINK :

- You can go to following link of youtube where a video is provided with setup and configuration instructions of softwares and library files required to run this game (code) :
- Youtube Link : https://youtu.be/kO7iurXhV1s

11. REFERENCES:

1. **Code-blocks installation guide and manual:**
   - http://www.cse.iitb.ac.in/~ranade/simplecpp/
   - http://www.it.iitb.ac.in/frg/wiki/images/e/e8/CodeBlockManual.pdf
2. **Guide for including OpenGL library files in Code-blocks :**
   - http://www.deannicholls.co.uk/tutorials/show/cpp_glut
3. **OpenGL Tutorials:**
   - http://www.glprogramming.com/red/
   - http://www.cs.uccs.edu/~ssemwal/indexGLTutorial.html
   - https://www.opengl.org/resources/libraries/glut/spec3/node113.html