

USER MANUAL

1. About the Game

Chain Reaction is a multiplayer game intended for groups of player strength up to 8(as per our coding this has been reduced to 6 because for more than 6 players our grid size is not large enough). It is a strategy game and tests a player's ability to plan ahead and play intelligently. Since its inception in 2011, the game has become very popular on all gaming platforms and is the app of choice for mobile game downloads. Apart from the visualization that is involved, this game also exposes students to the concept of critical mass and the speed with which chain reactions can propagate.

2. Motivation

As budding programmers - our main motivation for making this program was to exercise within a single program all the various tools we have learnt so far in C++. The code of this ingenious game leans heavily on looping constructs, function calls and recursive functions, arrays, structures and pointers. While honing our programming skills we aim to produce a polished product which will allow people to exercise their minds even while playing.

As for who can play- this is an intellectual game which can cut across various age groups and will appeal to a varied audience. We hope you will have as much fun playing it as we had coding it!

3. How to Play

The aim of the game is to take control of the board by eliminating orbs of every other player. The turns between players change hands cyclically. A player is allowed to place an orb on either an empty square or a square in which his own orbs are present. He cannot place an orb on a square which is occupied by orbs of a color different from his own.

There are three basic types of squares- the corners, the edges and the interior squares. The corners have a critical mass of 1, the edges have a critical mass of 2 and the interior squares have a critical mass of 3. Critical mass implies the number of orbs that can be placed in any square without it exploding. Adding orbs to the same square so as to bring it to the critical mass is called "powering". A corner exploding sends out orbs along the two sides it is bounded by. An edge sends out balls in three different directions which are along the edge and perpendicularly

outwards. An interior orb explodes and sends out 4 orbs in 4 perpendicular directions.

Orbs landing on same colour orbs simply add on to them while orbs landing on enemy orbs convert them to their own colour and then add on. At each step and for every square we check the critical condition and begin explosions.

It is thus easy to see that this game will require extensive iteration and efficient checking mechanisms at each move.

The game ends when orbs of only one colour are present on the board.

4. A) Some Techniques

The game requires a lot of strategic thinking on where to place orbs and how to restrict and block enemy growth by appropriate placing of orbs. A useful beginning move is to occupy the corners and construct chains of orbs emanating from these corners. Powering up these chains allow us to inflict huge chain conversions on enemy orbs. Placing orbs along the edges and powering them up allows us to block enemy chain growth.

4. B) Some Techniques for AI

However the above type of initial strategy works well in those board games only in which initially a player is restricted to one side of the board and cannot much affect the other player thus allowing each to implement an opening strategy as in chess. However in chain reaction this would not hold true. We therefore changed the approach and thought of using the brute force of a computer to aid in playing the game. We have essentially coded the computer such that the computer calculates every possibility of a move on the screen for two moves. It will then go on to calculate the ratio of the number of its balls to the number of balls of the opposite player. The ratio which is highest is selected and then the sequence of moves leading up to that ratio is chosen. The computer then accordingly plays its move.

5. Navigation around the Game

On opening the game, a screen displaying the name of the game opens. It also contains a menu which gives the player three options - It will flash the options the players can choose from.

Option 1- "Play"

Option 2- "Instructions"

Option 3 - "Exit"

They can go on directly to the game by clicking 1 from where they will be directed to another screen wherein they will be able to choose the number players (the color of each players orbs is preset). After this they can proceed towards playing the game.

By clicking 2, a brief synopsis of the game along with the instructions will appear. This would be mainly to explain to the players how the game would be played.

By clicking 3, we would exit the game. Note that once the game is over, the player who won the game would be shown in a separate screen. The players then have the option to return to the main menu from where a new game can be started or they can exit the game.

6. During Game play

The game begins with a grid of 8 by 6 in which the players can place orbs. During the game the players will have to click on the grid position they want to place their orb on. The turns of the player will vary cyclically with the first player being the one who was initially player 1. To notify the players about whose turn it is we have written code which flashes the number of the player whose turn it is. The color of the square in which the player id is displayed is of the players color. This prevents any confusion regarding whose turn it is.

Also, the win condition is met when only the orbs of one player are left.

When this does occur, the screen displays a message – “Player X wins!” (X is the id of the winning player).