GOBANG GAME DESIGN BASED ON JAVASCRIPT

A Project

Presented to the

Faculty of

California State Polytechnic University, Pomona

In Partial Fulfillment

Of the Requirements for the Degree

Master of Science

In

Computer Science

By

Xuanxi Guo

2018

SIGNATURE PAGE

PROJECT:	GOBANG GAME DESIGN BASED ON JAVASCRIPT
AUTHOR:	Xuanxi Guo
DATE SUBMITTED:	Fall 2018
	Department of Computer Science

Dr. Gilbert S. Young
Project Committee Chair
Computer Science

Dr. Hao Ji
Project Committee
Computer Science

ABSTRACT

Gobang’s history can be retraced to thousands of years ago. It started from GO. It is so popular that people all over the world like to play it. At the same time, it is also a game that full of strategy. As a world level game, the Gobang world championship has been held for many years.

As a computer science student, I have been learning data structure, algorithms and programming skills for a few years. I want to design a game which is fully designed, coded, and tested by myself. Based on my research, most Gobang games can be played online are designed by C++ or Java. So, I decided to make it by using JavaScript and run it on all the popular browsers like Microsoft Edge, Chrome, and Safari. These three browsers can be found on any devices. It is easy to let people try my game.

This game is played by two sides, Human and AI. I searched some strategies and presented them by code. There are some algorithms to make AI smarter. There are two levels can be chosen. I am trying to design a neat and simple interface or players to access this game and easily understand how it works.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1
INTRODUCTION

Gobang is a simple and popular game, also called Gomoku, or Five in a Row (FIR). It is a logic board game played by two persons. It is played with black and white stones on a 15x15 intersection Go board made of wood.

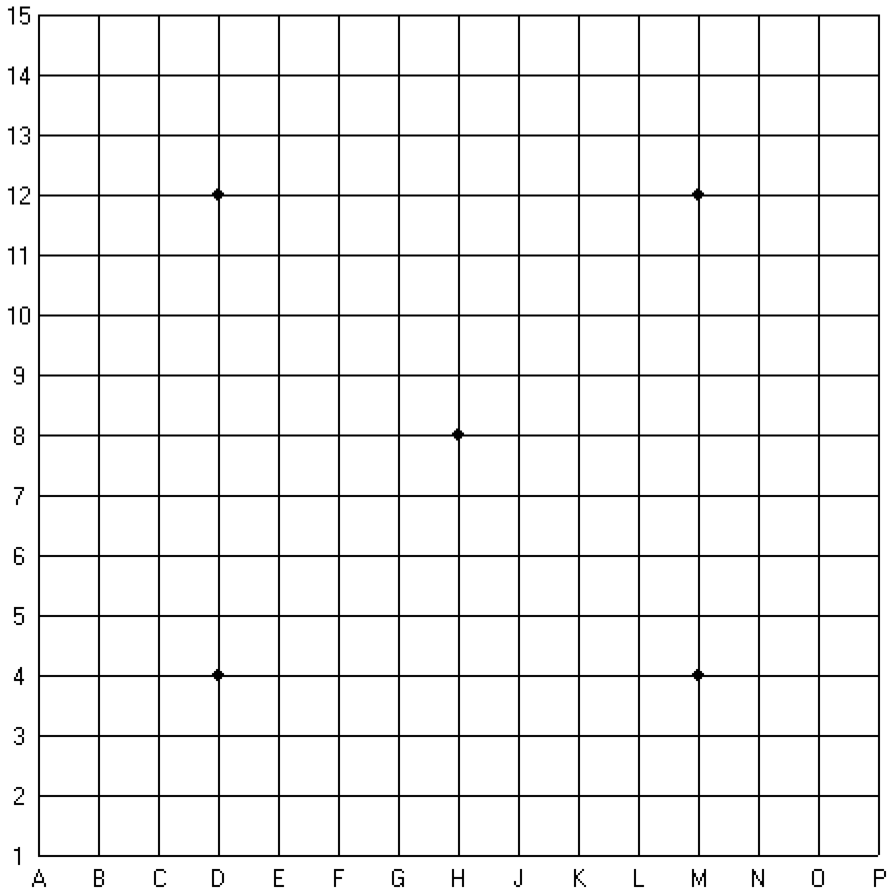

Figure 1. The game board

In the beginning of the game the board is empty. The player who controls the black pieces always starts the game. Moves must be placed on the intersections of the board, not on the squares closed by the lines. Black stone plays first, then white stone takes its turn, and this process continues until the end of the game. The players alternate in placing a stone of their color on an empty intersection.

The stones put on the table remain untouched till the end of the game. Getting down, removal or touching of stones are forbidden. It is naturally allowed to adjust a position of a piece when it is preceded by a sign. The winner is the first player to get an unbroken row of five stones either horizontally, vertically, or diagonally.

Winning the game requires a row of exactly five stones; a row of six or more stones - called overline - does not count.

1.1 The significance of the topic

The reason why human beings continue progressing is because human beings “think”. The development of the Gobang game program is a great driving force for the progress of thinking. This program allows people to play Gobang easily and quickly. Practicing logical and thinking abilities and expanding people's hobbies.

At the same time, the development of the Gobang program has also made the game widely promoted, letting people all over the world know and play Gobang, this is not a limitation. The Gobang makes more and more people like it and enjoy it. With the popularity of the Internet, more and more people can play Gobang with others.

1.2 The development and situation of Gobang

Gobang has quickly developed internationally. A lot people from different countries like this simple but full of strategies game and believe that Gobang can not only improve the logic, but also develop intelligence. Currently, more than 40 countries are playing Gobang and there are competitions of different sizes and levels. Now, Gobang is on every gaming platform provided for people to play. As long as the platform provides chess games, there must be a Gobang game in it. It develops fast, popularizes wide, and the prospects are very optimistic.

1.3 The world level games

Table 1. World Championship

Title year	Hosting city, country	Gold	Silver	Bronze	Opening rule
1989	Kyoto, Japan	Soviet Union Sergey Chernov	Soviet Union Yuriy Tarannikov	Japan Hirouji Sakamoto	Pro
1991	Moscow, Soviet Union	Soviet Union Yuriy Tarannikov	Soviet Union Ando Meritee	Soviet Union Sergey Chernov	Pro
2009	Pardubice, Czech Republic	Poland Artur Tamioła	Hungary Attila Demján	Czech Republic Pavel Laube	Swap2
2011	Huskvarna, Sweden	Hungary Attila Demján	Poland Artur Tamioła	Poland Michał Żukowski	Swap2
2013	Tallinn, Estonia	Hungary Attila Demján	Czech Republic Pavel Laube	Russia Mikhail Kozhin	Swap2
2015	Suzdal, Russia	Hungary Rudolf Dupszki	Hungary Gergő Tóth	Russia Mikhail Kozhin	Swap2

Table 2. Team World Championship

Title year	Hosting city, country	Gold	Silver	Bronze	Opening rule
2016	Tallinn, Estonia	Poland	Czech Republic	Chinese Taipei	Swap2
2018	Poland	Russia	Hungary	Poland	Swap2

CHAPTER 2
PROGRAM ENVIRONMENT AND TECHNOLOGY

2.1 Development Environment

1. Operating System: Windows 10
2. Developing Tool: Sublime Text 3
3. Programming Language: JavaScript, HTML5/CSS3

2.2 Tools and programming language

2.2.1 Sublime Text 3 Editor

Sublime Text Editor is a full featured Text editor for editing local files or a code base. It includes various features for editing code base which helps developers keep track of changes. Various features that are supported by Sublime are as follows −

- Syntax Highlight
- Auto Indentation
- File Type Recognition
- Sidebar with files of mentioned directory
- Macros
- Plug-in and Packages

Sublime Text editor is used as an Integrated Development Editor (IDE) like Visual Studio code and NetBeans. The current version of Sublime Text editor is 3.0 and is compatible with various operating systems like Windows, Linux and MacOS.

When I use Suitable Text editor, I can enjoy its rich beneficial features. Sublime Text offers its users the following benefits −

- Ability to solve linker errors.
- Keeping track of all files and folders to work with.
- Connectivity with version control systems like Git, Mercurial.
- Problem solving capabilities.
- Keeping color combination for syntax combination.

2.2.2 JavaScript

JavaScript often abbreviated as JS, is a high-level, interpreted programming language. It is a language which is also characterized as dynamic, weakly typed, prototype-based and multi-paradigm. The game is based on JavaScript, using multiple algorithms to implement.

It is lightweight and most commonly used as a part of web pages, whose implementations allow client-side script to interact with the user and make dynamic pages. It is an interpreted programming language with object-oriented capabilities. JavaScript was first known as LiveScript, but Netscape changed its name to JavaScript, possibly because of the excitement being generated by Java. JavaScript made its first appearance in Netscape 2.0 in 1995 with the name LiveScript. The general-purpose core of the language has been embedded in Netscape, Internet Explorer, and other web browsers. The ECMA-262 Specification defined a standard version of the core JavaScript language.

- JavaScript is a lightweight, interpreted programming language.
- Designed for creating network-centric applications.
- Complementary to and integrated with Java.
- Complementary to and integrated with HTML.
- Open and cross-platform.

CHAPTER 3
PROGRAM ANALYSIS AND DESIGN

3.1 Functions

1. Display a 15 * 15 gaming board
2. Put stones on the board by clicking mouse, display stones at the position where the mouse clicked, and the stones located on the cross line.
3. Automatically judge if the game is over, won by black or white, and pop up dialog showing the winning player.
4. Steps function, recording the totally steps have done.
5. Set entry level and advanced level.
6. Set AI first or Human First function.

3.2 The structure of the program

3.2.1 Draw the board

Drawing a 15 * 15 board, the distance between every two neighboring lines are the same. There are totally 5 stars on the board, the center on is called “tianyuan point”, same as GO.

3.2.2 Put the stone on the board

After clicking the mouse, there will be a stone put on the board. The black stone is solid black circle. The white stone is solid white circle. When a stone is put, there will be a small red dot on the stone shows it is new put.

3.2.3 Black and white turns one by one

Black goes first then white, then black… process alternately. This is the basic rule of chess games. So, we need to set a mutual exclusion action in it. When putting stones, players must put at different crosses, when trying to put at the cross but there is already a stone on it, it would not work.

3.2.4 Judge the winning

When a “five in a row” appears, we say it’s a win. The five stone can be line up in a row, a column, or slanted.

3.2.5 Game information

Winning counts, steps, first going, and difficulty levels.

3.2.6 Reset game

Clear the board and record who won last time and display total winning counts.

CHAPTER 4
FUNCTIONS AND CODE

4.1 Index.html

Index.html is not only provides the entrance of the game, but also connects each file and make them work.

There are only two functions in this page “board” and “mouse”.

“board” function is calling the “paint_chess()” function, so only two lines codes can make this happen.

```
window.onload = function(){
        paint_chess();          //drwa the board


}
```

The “mouse” function stars from “getMousePos()” which is getting mouse’s position and listening left click or right click all the time.

```
function mouse(){
        if(started == 1){
                var ad = getMousePos();
                step++;
                var stepp = document.getElementById("step");
                stepp.innerHTML = "Steps："+step;
                var score = document.getElementById("score");
                if(chessboard[ad.x][ad.y] == 0){
                        add_chessman(ad,turn)
                        if(1 == xianshou)
                            circle(ad.x*weight+8,ad.y*weight+8,14,"#fff");
                        else
                            circle(ad.x*weight+8,ad.y*weight+8,14,"#000");
                        turn = 1;
                        var nn = check();
                        if(nn == 1){
                            alert("AI won !");
                            step=0;
                            win_times[0]++;
                            score.innerHTML="AI won："+win_times[0]+"<br/>You
won：<br/>"+win_times[1];
                            started = 0;
                            return;
                        }else if(nn == 2){
                            alert("Congratulations，You won!");
                            step=0;
                            win_times[1]++;
                            score.innerHTML="AI won："+win_times[0]+"<br/>You
won：<br/>"+win_times[1];
                            started = 0;
                            return ;
                        }
                        initialize();
```

```
        pingfe();
        var temp = sosuo(turn);
        if(chessboard[temp.x][temp.y] == 0){
            add_chessman(temp,turn)
            if(1 == xianshou){

circle(temp.x*weight+8,temp.y*weight+8,14,"#000");

circle(temp.x*weight+8,temp.y*weight+8,4,"#f00");

circle(temp2.x*weight+8,temp2.y*weight+8,5,"#000");
            }
            else{

circle(temp.x*weight+8,temp.y*weight+8,14,"#fff");

circle(temp.x*weight+8,temp.y*weight+8,4,"#f00");

circle(temp2.x*weight+8,temp2.y*weight+8,5,"#fff");
            }

            temp2 = temp;
            turn = 2;
            }
            var nn = check();
            if(nn == 1){
                alert("AI won!");
                win_times[0]++;
                step=0;
                score.innerHTML="AI won："+win_times[0]+"<br/>You
won：<br/>"+win_times[1];
                started = 0;
            }else if(nn == 2){
                alert("Congratulations，You won!");
                step=0;
                win_times[1]++;
                score.innerHTML="AI won："+win_times[0]+"<br/>You
won：<br/>"+win_times[1];
                started = 0;
            }
        }
    }
}
```

4.2 Buttons.js

There are three main variables, “rank”, “turn”, “step”. “rank” is set to the level of game, there are two levels, entry-level and expert-level. Before starting the game, player can choose AI play first or player play first and choose the difficulty of level.

“rank” is set as default 0, which means entry-level. When rank equals 1, that is expert-level.

“turn” is set as default 1 means player first. When turn equals 2, that means AI first.

“step” is set as default 0, it records the steps currently both sides have done.

Choose first playing function:

```
function first (){                    //choose who first
        started = 0;
        chushihua();
        var start = document.getElementById("start");
                start.value = "Start";
        var first = document.getElementById("first");
        if(xianshou == 1){
                first.value = "Man First";
                xianshou = 2;
                turn = 2;
        }else{
                first.value = "AI First";
                xianshou = 1;
                turn = 1;
        }
}
```

When clicking the start/reset button, the function start() calls “chushihua()” which initials the game board. The main page displays records, including “AI win counts”, “Man win counts”. The number of each counts stores in an array “win_time[]”.

Start game/ Reset game:

```
function start(){
        if(step > 0){
                win_times[0]++;
        }
                score.innerHTML="AI Won :"+win_times[0]+"<br/>You Won :<br/>"+win_times[1];
        temp2 = 0;
        chushihua();
        started = 1;
        if(xianshou == 1){                //AI First
                chessboard[8][8] = 1;
                pingfen[8][8][0] = 1;
                circle(8*weight+8,8*weight+8,14,"#000");
        }
        turn = 2;
        var start = document.getElementById("start");
        start.value = "Reset";
}
```

“geta()” function is to get the actions from mouse.

```
function geta(){
    var ss = document.getElementById("sele");
    rank = ss.value;
    started = 0;
    chushihua();
    var start = document.getElementById("start");
        start.value = "Start";
    var first = document.getElementById("first");
        first.value = "AI First";
        xianshou = 1;
        turn = 1;
}
```

Function “chushihua()” will call “paint_chess()” function, which is explained in paint.js section. The board is 15 rows by 15 column square.

```
function chushihua(){
    paint_chess();
    step = 0;
    for(i = 0;i<15;i++){
        for(j = 0;j < 15;j++){
            chessboard[i][j] = 0;
            for(k = 0;k < 3; k++){
                pingfen[i][j][k] = 0;
            }
            for(k = 0; k < 8;k ++){
                for(l = 0;l < 5;l++){
                    chess[i][j][k][l] = 0;
                }   //initial every record
            }
        }
    }
}
```

4.3 Paint.js

"paint_chess" function is to draw the board. There are 15 rows and 15 columns. Five stars located on the board and the coordinates at (3,3), (3,11), (11,3), (11,11), (7,7).

```
function paint_chess(){
    var canvas=document.getElementById('myCanvas');
    var ctx=canvas.getContext('2d');

    ctx.fillStyle='#DDB844';
    ctx.fillRect(0,0,450,450);

        var startx = bianx;
        var starty = bianx;
        var endx = bianx;
        var endy = 450-bianx;

        ctx.lineWidth=2;
        for(var i = 1;i<=15;i++ ){

            ctx.beginPath();
            ctx.moveTo(startx,starty);
            ctx.lineTo(endx,endy);
            ctx.stroke();

            startx =  weight*i+bianx;
            endx =  weight*i+bianx;
        }
        startx = bianx;
        starty = bianx;
        endx = 450 - bianx;
        endy = bianx;
        for(var i = 1;i<=15;i++ ){
            ctx.beginPath();
            ctx.moveTo(startx,starty);
            ctx.lineTo(endx,endy);
            ctx.stroke();
```

```
            starty =  weight*i+bianx;
            endy =  weight*i+bianx;
        }
    circle(3*weight+bianx,3*weight+bianx,4,'#000');
    circle(3*weight+bianx,11*weight+bianx,4,'#000');
    circle(11*weight+bianx,3*weight+bianx,4,'#000');
    circle(11*weight+bianx,11*weight+bianx,4,'#000');
    circle(7*weight+bianx,7*weight+bianx,5,'#000');
}
```

“circle” function is to draw the stones. This function gets variables from “myCanvas” in index.css which set width 450-pixel, height 450-pixel.

```
function circle(x,y,radius,color){

    var canvas=document.getElementById('myCanvas');

    var ctx=canvas.getContext('2d');

    ctx.fillStyle=color;
    ctx.beginPath();
    ctx.arc(x,y,radius,0,Math.PI*2,true);
    ctx.closePath();
    ctx.fill();
}
```

“getMousePos” function is listening all the mouse actions.

```
function getMousePos(event) {

    var e = event || window.event;

    var scrollX = document.documentElement.scrollLeft || document.body.scrollLeft;

    var scrollY = document.documentElement.scrollTop || document.body.scrollTop;

    var x = e.pageX || e.clientX + scrollX;

    var y = e.pageY || e.clientY + scrollY;

            y = y - 40;


            x = x - (document.body.scrollWidth - 450)/2;

            x = Math.floor(x/30);

            y = Math.floor(y/30);


    //

    return { 'x': x, 'y': y };

 }
```

4.4 Index.css

CSS is short for Cascading Style Sheets which is a kind of style sheet language. It is used for describing the presentation of a document written in a markup language like HTML. CSS3 is the latest evolution of the Cascading Style Sheets language and aims at extending CSS2.1. It brings a lot of long-awaited novelties, like rounded corners, shadows, gradients, transitions or animations, as well as new layouts like multi-columns, flexible box or grid layouts. Experimental parts are vendor-prefixed, and should either be avoided in production environments, or used with extreme caution as both their syntax and semantics can change in the future.

Index.css set some default settings including buttons size, stones size, page style and so on.

4.5 New.js

All the algorithms are present here. Based on the stones' positions, these algorithms will calculate the weight of stones. Then, AI will choose the best position to put next stone.

There are some terminologies in Gobang in this part, coordinating with pictures to explain them better.

In this figure, every 5 stones put in a row from 0 to 4 or 4 to 0, it's a win.

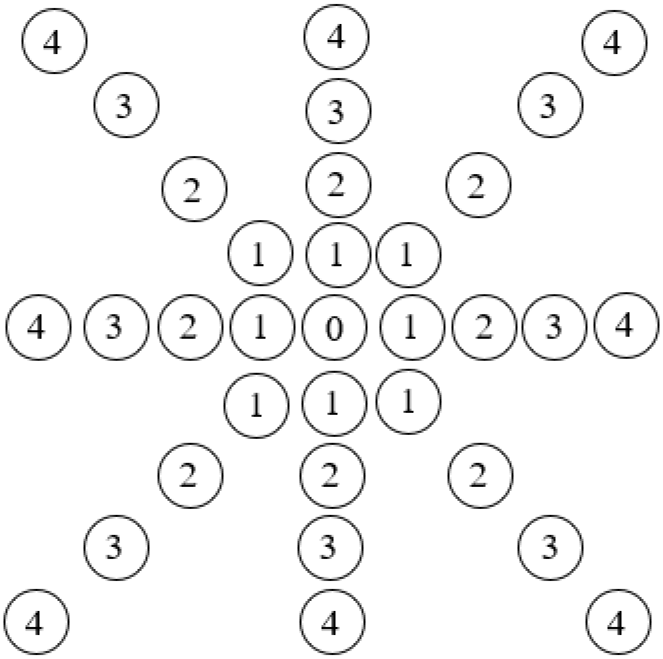

Figure 2. Five stones in a line

In this figure, a stone's 8 closed positions are marked as 0 to 7. This is used for calculating the weigh.

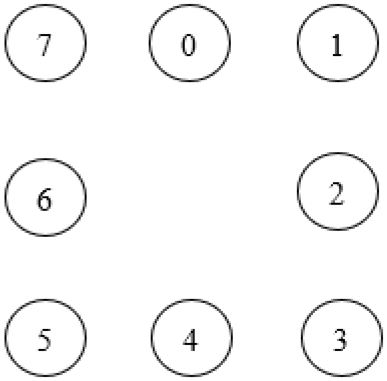

Figure 3. Eight directions surrounded a stone

CHAPTER 5
ALGORITHMS

5.1 Algorithms

5.1.1 What is a win?

The unbroken row of five stones can be collected in the middle and in the edge of the board as well.

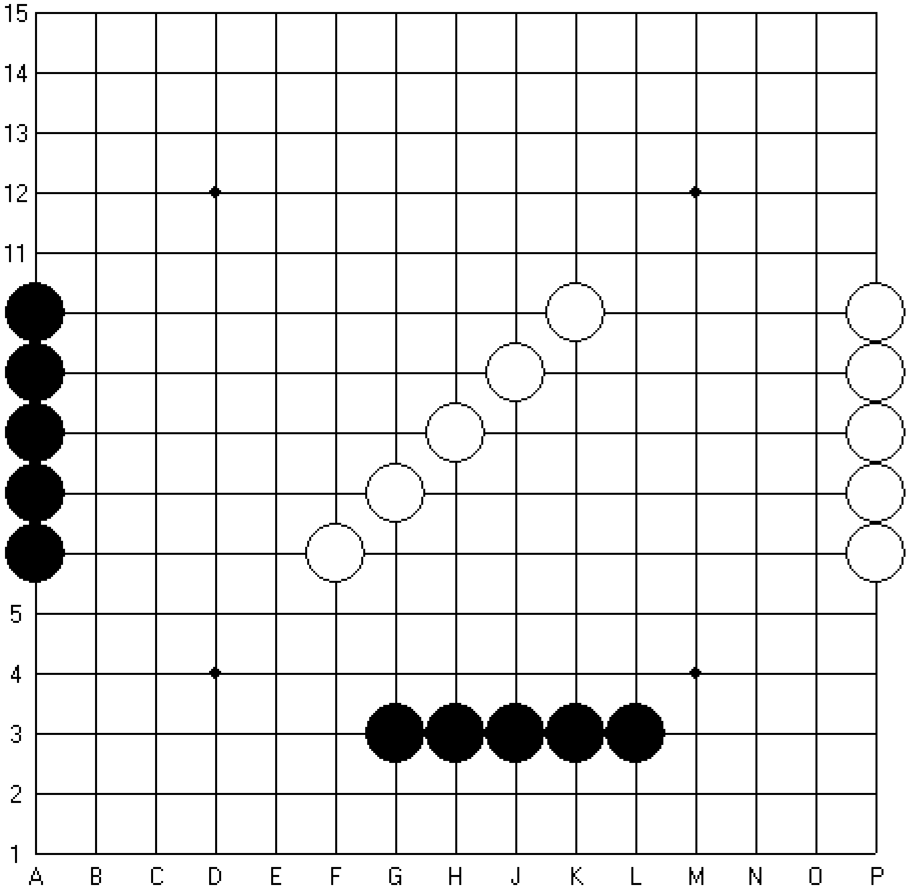

Figure 4. Example of a win

5.2 Winning Strategy

5.2.1 Blocking 4

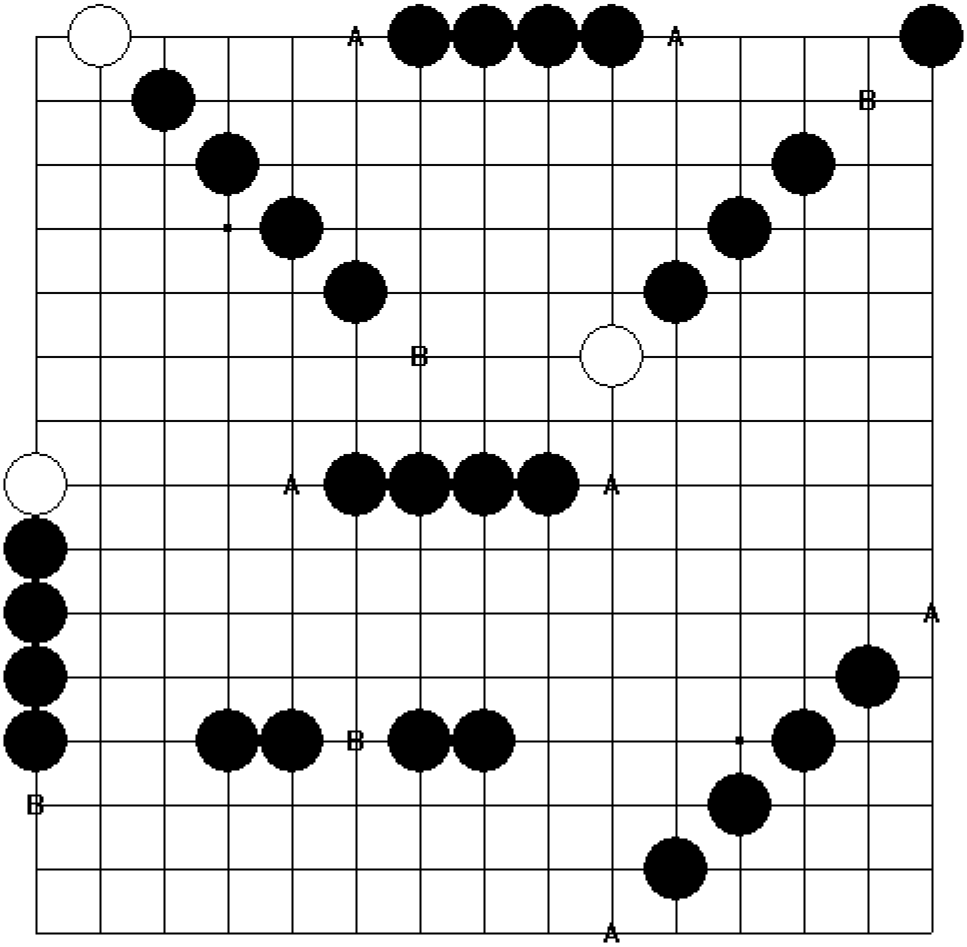

Figure 5. Example for blocking fours

On the upper diagram you can see all possible types of ’fours’. If black is on the move, he can gain victory by placing his fifth stone to points ’A’ or ’B’.

Points signed with ’B’ show ’half-opened fours’. This means that if it is white’s turn, he can avoid being defeated by blocking the four. Points signed with ’A’ show ’opened fours’. In this case white cannot avoid losing, because no matter which end, he blocks black will add a fifth stone to the opposite side and will have won the game. Thus, the main goal of the game is to create an open line of four stones in a row before your opponent does.

Code (AI):

```
for(k=0;k<4;k++){

        if(chess[i][j][k][0]==player1&&chess[i][j][k][1]==player1&&chess[i][j][k][2]==player1&&chess[i][j][k][3]==player2&&chess[i][j][k+4][0]==player2)
        // 2 0 0 0 1 2
                                pingfen[i][j][player1]+=10;

        if(chess[i][j][k+4][0]==player1&&chess[i][j][k+4][1]==player1&&chess[i][j][k+4][2]==player1&&chess[i][j][k+4][3]==player2&&chess[i][j][k][0]==player2)
// 2 1 0 0 0 2
                                pingfen[i][j][player1]+=10;

        if(chess[i][j][k][0]==player1&&chess[i][j][k][1]==player1&&chess[i][j][k+4][0]==player1&&chess[i][j][k][2]==player2&&chess[i][j][k+4][1]==player2)
        // 2 0 0 1 0 2
                                pingfen[i][j][player1]+=10;

        if(chess[i][j][k][0]==player1&&chess[i][j][k+4][0]==player1&&chess[i][j][k
+4][1]==player1&&chess[i][j][k+4][2]==player2&&chess[i][j][k][1]==player2)
        //  2 0 1 0 0 2
                                pingfen[i][j][player1]+=10;
                                }//AI
```

Code (Man):

```
for(k=0;k<4;k++){

        if(chess[i][j][k][0]==player2&&chess[i][j][k][1]==player2&&chess[i][j][k][2
]==player2&&chess[i][j][k][3]==player1&&chess[i][j][k+4][0]==player1)
                                pingfen[i][j][player2]+=10;

        if(chess[i][j][k+4][0]==player2&&chess[i][j][k+4][1]==player2&&chess[i][j]
[k+4][2]==player2&&chess[i][j][k+4][3]==player1&&chess[i][j][k][0]==player1)
                                pingfen[i][j][player2]+=10;

        if(chess[i][j][k][0]==player2&&chess[i][j][k][1]==player2&&chess[i][j][k+4
][0]==player2&&chess[i][j][k][2]==player1&&chess[i][j][k+4][1]==player1)
                                pingfen[i][j][player2]+=10;

        if(chess[i][j][k][0]==player2&&chess[i][j][k+4][1]==player2&&chess[i][j][k
+4][0]==player2&&chess[i][j][k+4][2]==player1&&chess[i][j][k][1]==player1)
                                pingfen[i][j][player2]+=10;
```

}//Man

5.2.3 Opened fours

Opened fours can only be made from opened threes. So, if you want to win it is indispensable to create opened three (points signed with ’C’ show places where opened fours can be made).

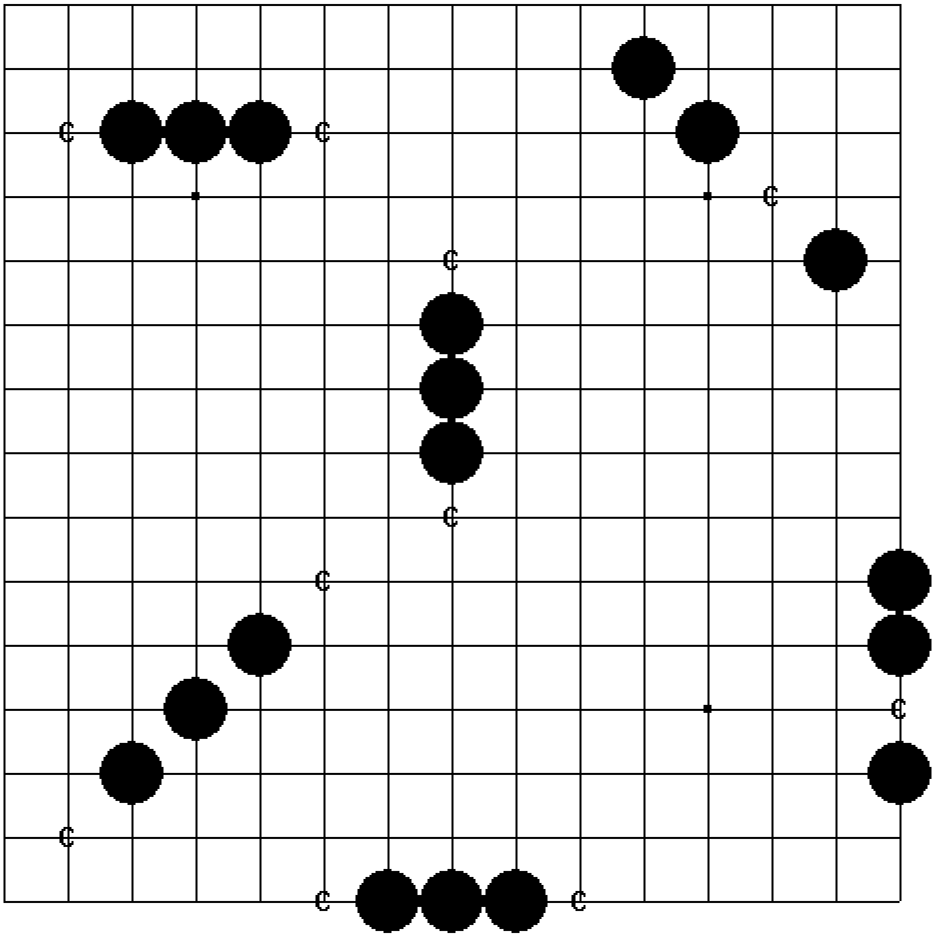

Figure 6. Example for creating opened fours

As you know already, if your opponent has an opened four, you will have lost the game. But before it, your opponent needs an opened three so as to create an opened four, that’s why you always have to be careful when you have discovered that your opponent has an opened three.

At any points signed with ’D’ white can easily stop blacks’ opened three lines and after that those lines won’t be dangerous anymore. It’s also important to pay attention to the points signed with ’E’, because these points mean possible defenses for white as well. Thanks to the edge of the board or an earlier put white stone there are several ways to block your opponents opened three. Putting your white stone to the ’E’ points black is not able anymore to create opened four in those planned lines. In practice such blocking ways are seldom used, because they aren’t as safe as the blocks at points ’D’.

Code(AI):

```
for(k=0;k<4;k++){


        if(chess[i][j][k][0]==player1&&chess[i][j][k][1]==player1&&chess[i][j][k][2
]==player1&&chess[i][j][k][3]==0&&chess[i][j][k+4][0]==0) // 6 0 0 0 1 6
                                                pingfen[i][j][player1]+=fen[rank][1];


        if(chess[i][j][k+4][0]==player1&&chess[i][j][k+4][1]==player1&&chess[i][j]
[k+4][2]==player1&&chess[i][j][k+4][3]==0&&chess[i][j][k][0]==0)//6 1 0 0 0 6
                                                pingfen[i][j][player1]+=fen[rank][1];


        if(chess[i][j][k][0]==player1&&chess[i][j][k][1]==player1&&chess[i][j][k][2
]==0&&chess[i][j][k+4][0]==player1&&chess[i][j][k+4][1]==0)     // 6 0 0 1 0 6
                                                pingfen[i][j][player1]+=fen[rank][1];
        if(chess[i][j][k+4][0]==player1&&chess[i][j][k+4][1]==player1&&chess[i][j][k+4][2]==0&&chess[i][j][k][0]==player1&&chess[i][j][k][1]==0)  //6 0 1 0 0 6
                                        pingfen[i][j][player1]+=fen[rank][1];
                           }//AI
```

Code(Man):

```
for(k=0;k<4;k++){

        if(chess[i][j][k][0]==player2&&chess[i][j][k][1]==player2&&chess[i][j][k][2]==player2&&chess[i][j][k][3]==0&&chess[i][j][k+4][0]==0)
                                        pingfen[i][j][player2]+=fen[rank][5];

        if(chess[i][j][k+4][0]==player2&&chess[i][j][k+4][1]==player2&&chess[i][j][k+4][2]==player2&&chess[i][j][k+4][3]==0&&chess[i][j][k][0]==0)
                                        pingfen[i][j][player2]+=fen[rank][5];

        if(chess[i][j][k][0]==player2&&chess[i][j][k][1]==player2&&chess[i][j][k][2]==0&&chess[i][j][k+4][0]==player2&&chess[i][j][k+4][1]==0)
                                        pingfen[i][j][player2]+=fen[rank][5];

        if(chess[i][j][k+4][0]==player2&&chess[i][j][k+4][1]==player2&&chess[i][j][k+4][2]==0&&chess[i][j][k][0]==player2&&chess[i][j][k][1]==0)
                                        pingfen[i][j][player2]+=fen[rank][5];
}//Man
```

*5.2.4 3*3 fork*

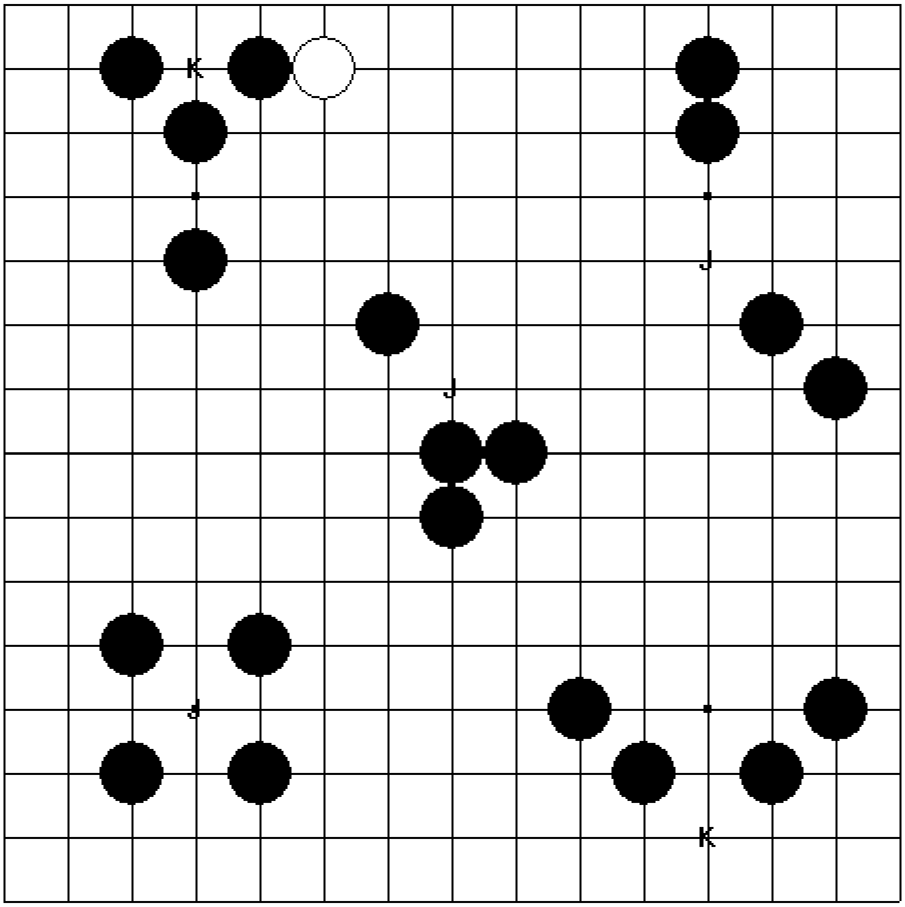

*Figure 7. Example for 3*3 fork*

A fork consist of two threes can only be winning if both threes are opened. Points signed with 'J' show winning forks because of opened threes.

Points 'K' don't mean a winning fork because one of the threes is a closed one.

Code(AI):

```
for(k=0;k<4;k++){
        if(chess[i][j][k][0]==player1&&chess[i][j][k][1]==player1&&chess[i][j][k][2
]==player2&&chess[i][j][k+4][0]==player2) // 2 0 0 1 2
                pingfen[i][j][player1]+=5;


        if(chess[i][j][k+4][0]==player1&&chess[i][j][k+4][1]==player1&&chess[i][j]
[k+4][2]==player2&&chess[i][j][k][0]==player2)  // 2 1 0 0 2
                pingfen[i][j][player1]+=5;


        if(chess[i][j][k][0]==player1&&chess[i][j][k][1]==player2&&chess[i][j][k+4
][0]==player1&&chess[i][j][k+4][1]==player2)    // 2 0 1 0 2
                pingfen[i][j][player1]+=5;
         }//AI
```

Code(Man):

```
for(k=0;k<4;k++){


        if(chess[i][j][k][0]==player2&&chess[i][j][k][1]==player2&&chess[i][j][k][2
]==player1&&chess[i][j][k+4][0]==player1)
                pingfen[i][j][player2]+=5;


        if(chess[i][j][k+4][0]==player2&&chess[i][j][k+4][1]==player2&&chess[i][j]
[k+4][2]==player1&&chess[i][j][k][0]==player1)
                pingfen[i][j][player2]+=5;
```

```
    if(chess[i][j][k][0]==player2&&chess[i][j][k][1]==player1&&chess[i][j][k+4
][0]==player2&&chess[i][j][k+4][1]==player1)
            pingfen[i][j][player2]+=5;
    }//Man
```

CHAPTER 6
EXTENSION

WeChat is a very famous App that is used by over 1 billion users all over the world. Thus, I plan to publish this Gobang game as a personal development game in WeChat since the WeChat program is open to everyone. For this simple game to satisfy people is another goal of mine.

REFERENCES

1. Complexity of Basic Algorithms. (n.d.). *Algorithms in Real Algebraic Geometry Algorithms and Computation in Mathematics,*281-322. doi:10.1007/3-540-33099-2_9
2. Cost Approximation Algorithms, CA Algorithms. (n.d.). *SpringerReference*. doi:10.1007/springerreference_72179
3. Cooper, B. (2001). *Javascript: An introduction*. New York: Dorling Kindersley Pub.
4. Genetic Algorithms. (n.d.). *Optimization Algorithms in Physics,*159-184. doi:10.1002/3527600876.ch8
5. *Gomoku*. (2011). Place of publication not identified: Betascript Publishing.
6. Gosselin, D. (2011). *Javascript*. Boston, MA: Course Technology/Cengage Learning.
7. Lin, Y. M., & Rao, H. (2013). Design and Implementation of Gobang Online Game Based on MFC Socket. *Applied Mechanics and Materials,380-384*, 1834-1837. doi:10.4028/www.scientific.net/amm.380-384.1834
8. Luo, W., Duan, P., & Li, X. (2013). Human–Machine Gomoku with Artificial Intelligence Based on Genetic Gambling Algorithm. *Proceedings of the 9th International Symposium on Linear Drives for Industry Applications, Volume 4 Lecture Notes in Electrical Engineering,*261-270. doi:10.1007/978-3-642-40640-9_34
9. Melchor, J. (2017). *Understanding platform business models in the telecommunication industry*.
10. Qiu, Q., & Zhang, T. (2013). *Wu zi qi zhi shi*. Taibei Shi: Pin guan wen hua chu ban she.
11. Rzepka, C. J. (1995). *Sacramental commodities: Gift, text, and the sublime in De Quincey*. Amherst: University of Massachusetts Press.

12. Wang, Z., & Ding, L. (2018). Design and development of data acquisition system based on WeChat hardware. doi:10.1063/1.5041167

13. Zakas, N. C. (2010). *High performance JavaScript*. Beijing: OReilly/Yahoo! Press.

14. Zhang, M., Wu, J., & Li, F. (2013). Design of evaluation-function for computer gobang game system. *Journal of Computer Applications*,*32*(6), 1969-1972. doi:10.3724/sp.j.1087.2012.01969

15. Zhang, M., Wu, J., & Li, F. (2013). Design of evaluation-function for computer gobang game system. *Journal of Computer Applications*,*32*(6), 1969-1972. doi:10.3724/sp.j.1087.2012.01969