Tizen课程项目报告  
《井字棋》

## 项目简介

井字棋的棋盘共有3\*3个格子。先行标示为绿色的叉（X），后行标示为红色的圈（O）。横向、纵向或斜线上有三个相同棋子的一方获胜。井字棋规则简单，易学易懂，常作为儿童开发智力的游戏。

## 需求分析



## 概要设计

井字棋基于TIZEN web project开发，主要使用了Html与Javascript技术。通过在onDown方法中计算手指触碰的位置来判断用户要在哪里落子。通过update方法更新棋盘，画上新下的棋子。

|  |  |
| --- | --- |
| 接口 | 接口功能 |
| gameInit | 初始化游戏界面 |
| initLayer | 初始化棋盘 |
| onDown | 手指在棋盘上的点击事件处理 |
| computerThink | 电脑思考下一步棋走哪里 |
| update | 更新棋盘，画上新下的棋子 |

## 核心算法

井字棋核心算法代码如下图所示。

|  |
| --- |
| init(30,"mylegend",390,420,main);  var backLayer,chessLayer,overLayer;  var statusText = new LTextField();  var statusContent="电脑正在思考中……";  var matrix = [  [0,0,0],  [0,0,0],  [0,0,0]  ];  var usersTurn = true;  var step = 0;  var title = "井字棋";  var introduction = ""  var infoArr = [title,introduction];  function main(){  gameInit();  addText();  addLattice();  }  function gameInit(){  initLayer();  computerThink();  addEvent();  }  function initLayer(){  backLayer = new LSprite();  addChild(backLayer);  chessLayer = new LSprite();  backLayer.addChild(chessLayer);  overLayer = new LSprite();  backLayer.addChild(overLayer);  }  function addEvent(){  backLayer.addEventListener(LMouseEvent.MOUSE\_DOWN,onDown);  }  function onDown(){  var mouseX,mouseY;  mouseX = event.offsetX;  mouseY = event.offsetY;  var partX = Math.floor(mouseX/130);  var partY = Math.floor(mouseY/130);  if(matrix[partX][partY]==0){  usersTurn=false;  matrix[partX][partY]=-1;  step++;  update(partX,partY);    if(win(partX,partY)){  statusContent = "帅呆了，你赢啦！点击屏幕重开游戏。";  gameover();  addText();  }else if(isEnd()){  statusContent = "平局啦~~点击屏幕重开游戏。";  gameover();  addText();  }else{  statusContent = "电脑正在思考中……";  addText();  computerThink();  }  }  }  function addText(){  statusText.size = 15;  statusText.weight = "bold";  statusText.color = "white";  statusText.text = statusContent;  statusText.x = (LGlobal.width-statusText.getWidth())\*0.5;  statusText.y = 393;    overLayer.addChild(statusText);  }  function addLattice(){  backLayer.graphics.drawRect(10,"dimgray",[0,0,390,420],true,"dimgray");  backLayer.graphics.drawRect(10,"dimgray",[0,0,390,390],true,"lavender");  for(var i=0;i<3;i++){  backLayer.graphics.drawLine(3,"dimgray",[130\*i,0,130\*i,390]);  }  for(var i=0;i<3;i++){  backLayer.graphics.drawLine(3,"dimgray",[0,130\*i,390,130\*i]);  }  }  function update(x,y){  var v = matrix[x][y];  if(v>0){  chessLayer.graphics.drawArc(10,"green",[x\*130+65,y\*130+65,40,0,2\*Math.PI]);  }else if(v<0){  chessLayer.graphics.drawLine(20,"#CC0000",[130\*x+30,130\*y+30,130\*(x+1)-30,130\*(y+1)-30]);  chessLayer.graphics.drawLine(20,"#CC0000",[130\*(x+1)-30,130\*y+30,130\*x+30,130\*(y+1)-30]);  }  }  function computerThink(){  var b = best();  var x = b.x;  var y = b.y;  matrix[x][y]=1;  step++;  update(x,y);    if(win(x,y)){  statusContent = "哈哈你输了！点击屏幕重开游戏。";  gameover();  addText();  }else if(isEnd()){  statusContent = "平局啦~~点击屏幕重开游戏。";  gameover();  addText();  }else{  statusContent = "该你了！！！";  addText();  }  }  function isEnd(){  return step>=9;  }  function win(x,y){  if(Math.abs(matrix[x][0]+matrix[x][1]+matrix[x][2])==3){  return true;  }  if(Math.abs(matrix[0][y]+matrix[1][y]+matrix[2][y])==3){  return true;  }  if(Math.abs(matrix[0][0]+matrix[1][1]+matrix[2][2])==3){  return true;  }  if(Math.abs(matrix[2][0]+matrix[1][1]+matrix[0][2])==3){  return true;  }  return false;  }  function best(){  var bestx;  var besty;  var bestv=0;  for(var x=0;x<3;x++){  for(var y=0;y<3;y++){  if(matrix[x][y]==0){  matrix[x][y] = 1;  step++;  if(win(x,y)){  step--;  matrix[x][y] = 0;  return {'x':x,'y':y,'v':1000};  }else if(isEnd()){  step--;  matrix[x][y]=0;  return {'x':x,'y':y,'v':0};  }else{  var v=worst().v;  step--;  matrix[x][y]=0;  if(bestx==null || v>=bestv){  bestx=x;  besty=y;  bestv=v;  }  }  }  }  }  return {'x':bestx,'y':besty,'v':bestv};  }  function worst(){  var bestx;  var besty;  var bestv = 0;  for(var x=0;x<3;x++){  for(var y=0;y<3;y++){  if(matrix[x][y] == 0){  matrix[x][y] = -1;  step++;  if(win(x,y)){  step--;  matrix[x][y] = 0;  return {'x':x,'y':y,'v':-1000};  }else if(isEnd()){  step--;  matrix[x][y]=0;  return {'x':x,'y':y,'v':0};;  }else{  var v=best().v;  step--;  matrix[x][y]=0;  if(bestx==null || v<=bestv){  bestx=x;  besty=y;  bestv=v;  }  }    }  }  }  return {'x':bestx,'y':besty,'v':bestv};  }  function gameover(){  backLayer.removeEventListener(LMouseEvent.MOUSE\_DOWN,onDown);  backLayer.addEventListener(LMouseEvent.MOUSE\_DOWN,function(){  chessLayer.removeAllChild();  backLayer.removeChild(chessLayer);  backLayer.removeChild(overLayer);  removeChild(backLayer);  matrix = [  [0,0,0],  [0,0,0],  [0,0,0]  ];  step = 0;  main();  statusContent = "您先请吧……";  addText();  });  } |

## 软件功能界面

