

Taller de Computación Gráfica

por: Joaquín Arias Herrero (sites.google.com/site/tallercomputaciongrafica2014/)

Hablaremos de:

- Teoría de juegos
- Programación OO
- Estructura de Datos
- Historia Computación
- Inteligencia Artificial
- Matemáticas Discretas
- Geometría Afín
- Álgebra Lineal
- Estructuras Algebraicas

Objetivos:

- Crear el Tres en Raya
- Pasarlo bien

Prerequisito:

- Conocer la biografía de John von Neumann
http://es.wikipedia.org/wiki/John_von_Neumann



Viernes 7 febrero 2014

ETSIInf

15:30* a 17:30



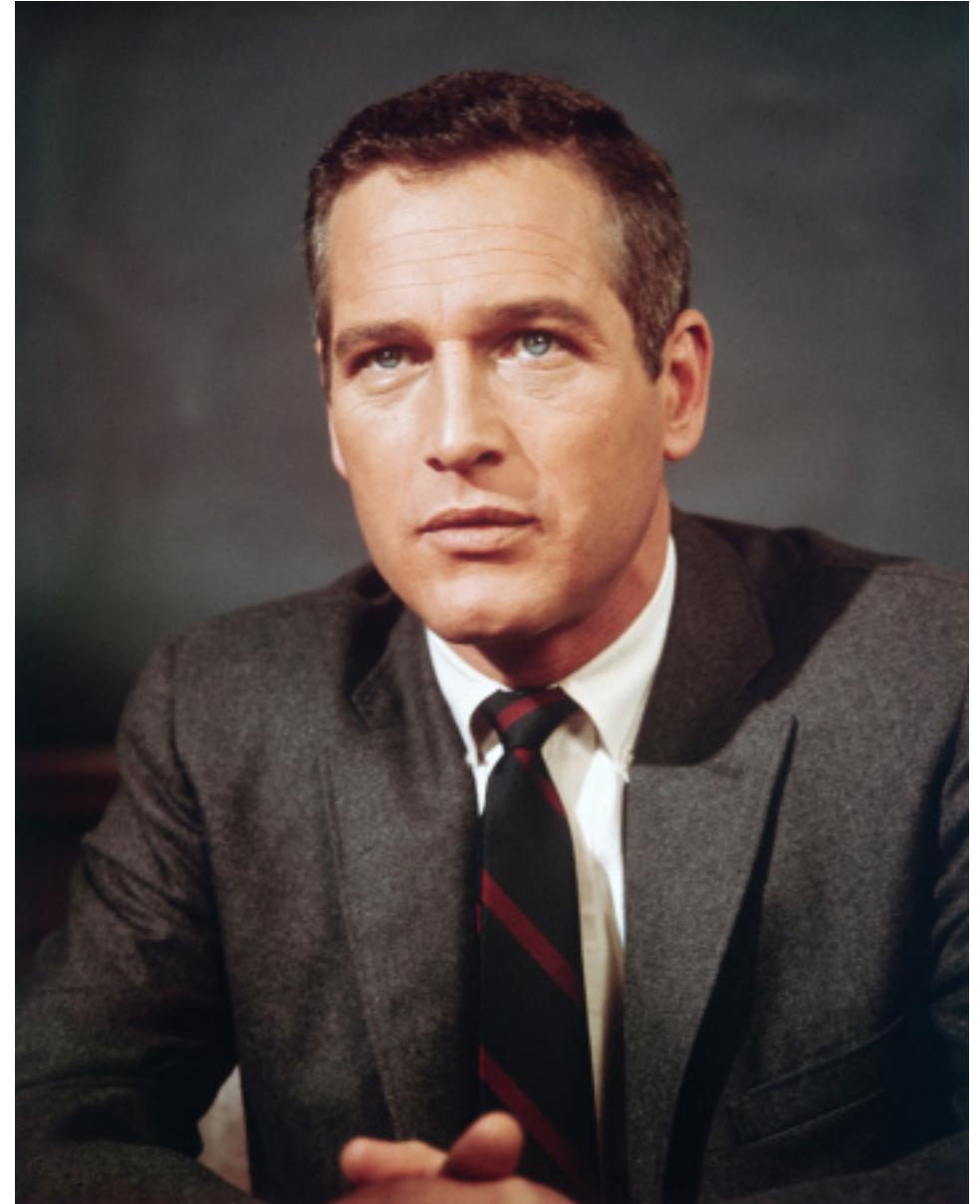
* a partir de las 15:00 para configuración de Processing + Android-SDK

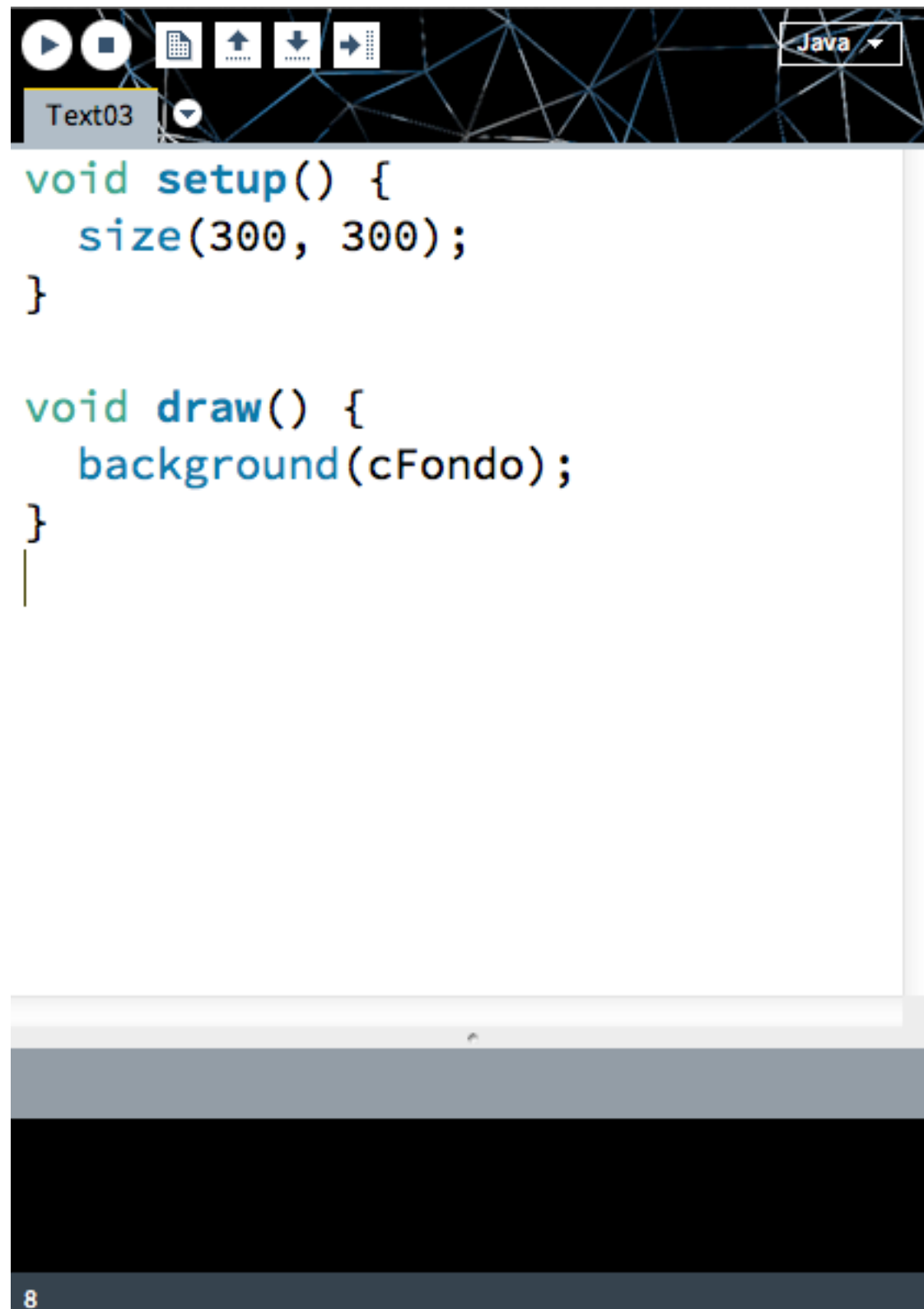
WAR GAMES

Juegos de Guerra (1983)



¿Quien es J.v. Neumann?





```
void setup() {  
  size(300, 300);  
}  
  
void draw() {  
  background(cFondo);  
}
```

8

int cFondo = 0, cFichas = 255;

```
void setup() {  
  size(300, 300);  
}
```

```
void draw() {  
  background(cFondo);  
  strokeWeight(1);  
  stroke(cFichas);  
  line(0, 100, 300, 100);  
  line(0, 200, 300, 200);  
  line(100, 0, 100, 300);  
  line(200, 0, 200, 300);  
}
```



```
int cFondo = 0, cFichas = 255;
```

```
void setup() {  
  size(300, 300);  
}
```

```
void draw() {  
  background(cFondo);  
  strokeWeight(1);  
  stroke(cFichas);  
  line(0, 100, 300, 100);  
  line(0, 200, 300, 200);  
  line(100, 0, 100, 300);  
  line(200, 0, 200, 300);  
  dibujaFichaO(50, 50);  
  dibujaFichaO(150,50);  
  dibujaFichaX(150, 150);  
}
```

```
void dibujaFichaO(int x, int y){  
  strokeWeight(70);  
  stroke(cFichas);  
  point(x, y);  
  strokeWeight(50);  
  stroke(cFondo);  
  point(x, y);  
}
```

```
void dibujaFichaX(int x, int y){  
  strokeWeight(10);  
  stroke(cFichas);  
  line(x-30, y-30, x+30, y+30);  
  line(x-30, y+30, x+30, y-30);  
}
```



```

int cFondo = 0, cFichas = 255;
Ficha fichaA, fichaB;

void setup() {
  size(300, 300);
  fichaA = new Ficha(50, 50, 1);
  fichaB = new Ficha(150, 50, 2);
}

void draw() {
  background(cFondo);
  strokeWeight(1);
  stroke(cFichas);
  line(0, 100, 300, 100);
  line(0, 200, 300, 200);
  line(100, 0, 100, 300);
  line(200, 0, 200, 300);
  fichaA.dibuja();
  fichaB.dibuja();
}

class Ficha {
  int x, y, tipo;
  Ficha(int x, int y, int tipo) {
    this.x = x;

```

```

    this.y = y;
    this.tipo = tipo;
  }
void dibuja() {
  switch (tipo) {
    case 1:
      strokeWeight(70);
      stroke(cFichas);
      point(x, y);
      strokeWeight(50);
      stroke(cFondo);
      point(x, y);
      break;
    case 2:
      strokeWeight(10);
      stroke(cFichas);
      line(x-30, y-30, x+30, y+30);
      line(x-30, y+30, x+30, y-30);
      break;
    default:
      break;
  }
}
}

```



```
int cFondo, cFichas;  
Ficha[] fichas = new Ficha[9];
```

```
void setup() {  
    size(300, 300);  
    inicia();  
}
```

```
void inicia() {  
    for (int i = 0; i<9; i++) {  
        fichas[i] = new Ficha(i, i%3);  
    }  
    cFondo = 0;  
    cFichas = 255;  
}
```

```
void draw() {  
    background(cFondo);  
    strokeWeight(1);  
    stroke(cFichas);  
    line(0, 100, 300, 100);  
    line(0, 200, 300, 200);  
    line(100, 0, 100, 300);
```

```
    line(200, 0, 200, 300);  
    for (int i = 0; i<9; i++) {  
        fichas[i].dibuja();  
    }  
}
```

```
class Ficha {  
    int posicion, x, y, tipo;  
    Ficha(int posicion, int tipo) {  
        this.posicion = posicion;  
        this.x = (posicion % 3) * 100 + 50;  
        this.y = (posicion / 3) * 100 + 50;  
        this.tipo = tipo;  
    }  
    void dibuja() {  
        ...  
    }  
}
```



...

```
void inicia() {
```

...

```
    fichas[i] = new Ficha(i, 0);
```

...

```
}
```

```
void draw() {
```

...

```
}
```

```
void mousePressed(){  
    fichas[4].setTipo((mouseX/100)%3);  
}
```

```
class Ficha {
```

...

```
    void setTipo(int tipo) {  
        this.tipo = tipo;
```

```
    }
```

```
}
```




```
int cFondo, cFichas, jugador;
```

```
void inicia() {
```

```
    ...
```

```
    frameRate(10);
```

```
    jugador = 1;
```

```
}
```

```
void mousePressed() {
```

```
    if (fichas[(mouseX / 100) + (mouseY / 100) * 3].setTipo(jugador)) {
```

```
        jugador = (jugador == 1)? 2: 1;
```

```
    } else background(cFichas);
```

```
}
```

```
class Ficha {
```

```
    ...
```

```
    boolean setTipo(int tipo) {
```

```
        if (this.tipo == 0) {
```

```
            this.tipo = tipo;
```

```
            return true;
```

```
        }
```

```
        else return false;
```

```
    }
```

```
}
```



```
int cFondos, cFichas, jugador, colocadas;  
boolean FIN;
```

```
void inicia() {  
    ...  
    colocadas = 0;  
    FIN = false;  
}
```

```
void mousePressed() {  
    if (colocadas == 9 || FIN)  
        inicia();  
    else if (fichas[(mouseX / 100) + (mouseY / 100) * 3].setTipo(jugador)) {  
        jugador = (jugador == 1)? 2: 1;  
        colocadas ++;  
        FIN = check();  
    } else  
        background(cFichas);  
}
```

```
boolean check() {  
    return (enLinea(0, 1, 2) || enLinea(3, 4, 5) || enLinea(6, 7, 8) || enLinea(0, 3, 6) ||  
            enLinea(1, 4, 7) || enLinea(2, 5, 8) || enLinea(0, 4, 8) || enLinea(2, 4, 6));  
}
```

```
boolean enLinea (int a, int b, int c) {  
    return (fichas[a].tipo != 0 && fichas[a].tipo == fichas[b].tipo &&  
            fichas[b].tipo == fichas[c].tipo);  
}
```



```
int cFondo, cFichas, jugador, colocadas, a, c;
```

```
void draw() {
```

```
...
```

```
if (FIN) {
```

```
    strokeWeight(10);
```

```
    stroke(cFichas);
```

```
    line(fichas[a].x, fichas[a].y, fichas[c].x, fichas[c].y);
```

```
}
```

```
}
```

```
void mousePressed() {
```

```
...
```

```
    FIN = check();
```

```
    if (colocadas == 9 || FIN) {
```

```
        jugador = 1;
```

```
        cFondo = 255;
```

```
        cFichas = 0;
```

```
    }
```

```
    } else
```

```
        background(cFichas);
```

```
}
```

```
boolean enLinea (int a, int b, int c) {
```

```
    if (fichas[a].tipo != 0 && fichas[a].tipo == fichas[b].tipo && fichas[b].tipo == fichas[c].tipo) {
```

```
        this.a = a;
```

```
        this.c = c;
```

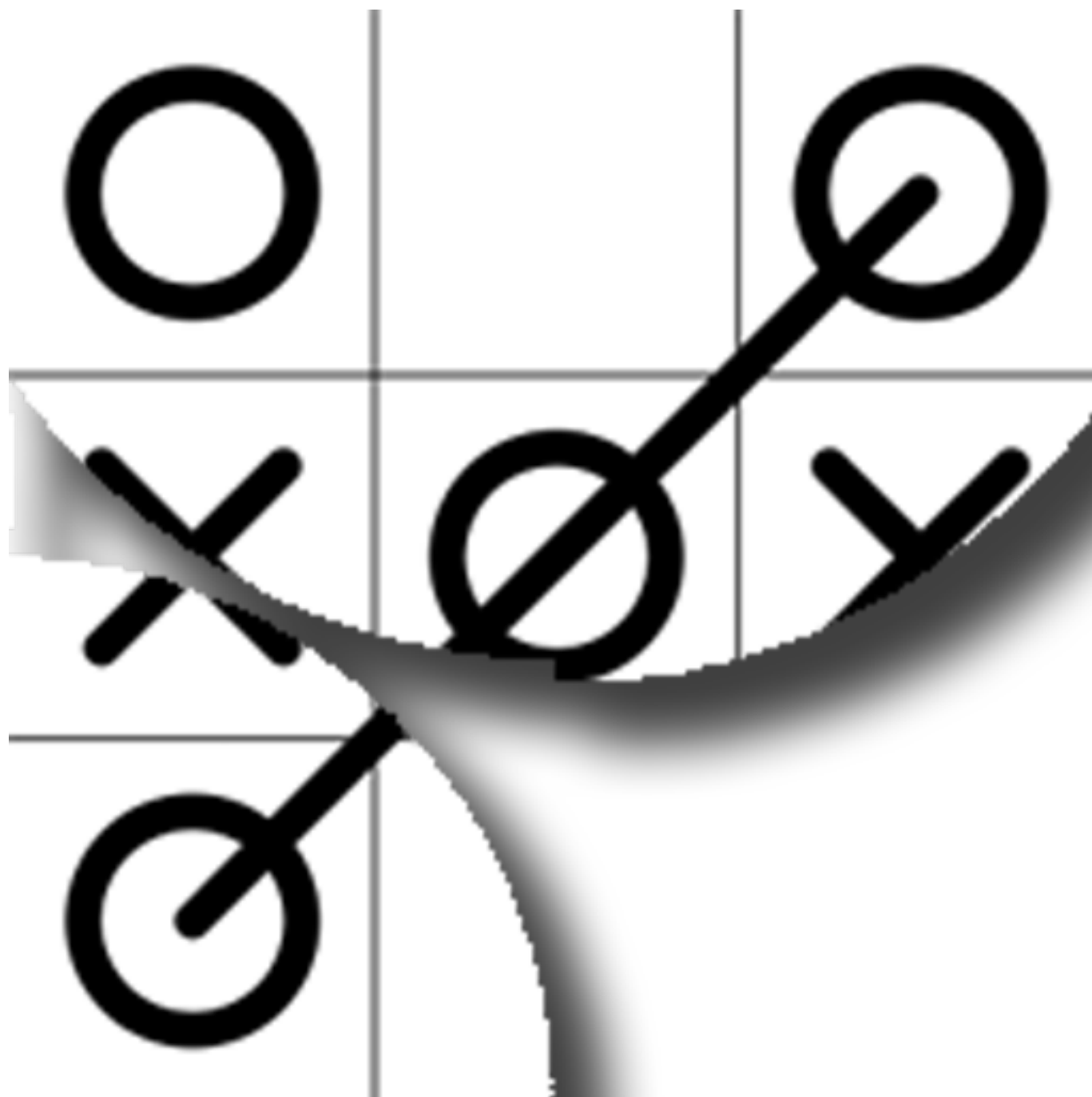
```
        return true;
```

```
    } else
```

```
        return false;
```

```
}
```





Zur Theorie der Gesellschaftsspiele¹⁾.

Von
J. v. Neumann in Berlin.

Einleitung.

1. Die Frage, deren Beantwortung die vorliegende Arbeit anstrebt, ist die folgende:

n Spieler, S_1, S_2, \dots, S_n , spielen ein gegebenes Gesellschaftsspiel G . Wie muß einer dieser Spieler, S_m , spielen, um dabei ein möglichst günstiges Resultat zu erzielen?

Die Fragestellung ist allgemein bekannt, und es gibt wohl kaum eine Frage des täglichen Lebens, in die dieses Problem nicht hineingeht. Trotzdem ist der Sinn dieser Frage kein eindeutiger. Für $n > 1$ ist (d. h. ein eigentliches Spiel) die Frage nach dem Resultat für jeden Spieler auf

Teoría de los juegos sociales [J.v. Neumann, 1987]

Este artículo da respuesta a la siguiente pregunta:

Dados n jugadores, S_1, S_2, \dots, S_n , jugando a un juego G .
¿Cómo debería jugar el jugador, S_m ,
con el fin de lograr el mejor resultado posible?

¹⁾ Der Inhalt dieser Arbeit ist (mit einigen Kürzungen) am 7. XII. 1926 der Göttinger Math. Ges. vorgetragen worden.

²⁾ Es ist das Hauptproblem der klassischen Nationalökonomie: was wird, unter gegebenen äußeren Umständen, der absolut egoistische „homo oeconomicus“ tun?

