

**UNIVERSIDAD DE GRANADA
E.T.S. DE INGENIERÍA INFORMÁTICA**



Departamento de Lenguajes y Sistemas Informáticos

DISEÑO ASISTIDO POR COMPUTADORA

PROYECTO: *AJEDREZ*

Manuel González López

Fecha: Enero 2003

1. Introducción.

El proyecto elegido para esta asignatura ha sido un *juego de Ajedrez*. Se ha llevado a cabo el diseño de todas las piezas que incluye el juego así como el correspondiente tablero. Además he implementado todas las reglas del juego: movimientos de las fichas, jaque, jaque mate y demás.

A través de esta documentación repasaremos los aspectos más importantes llevados a cabo en este proyecto.

2. Modelos jerárquicos.

En este apartado explicaremos como se ha llevado a cabo la estructura del modelo geométrico de las distintas partes que forman nuestro juego de ajedrez.

En primer lugar comentar que todas las piezas diseñadas llevan asociadas un *Display List* con el fin de acelerar el proceso de dibujo en pantalla. Dicho esto pasemos a detallar el proceso de modelado:

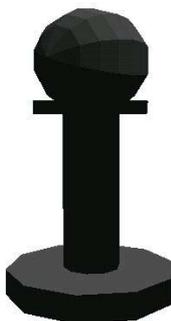
- **Tablero:** Evidentemente, una parte imprescindible es el tablero de juego. Su diseño es sencillo, consiste simplemente en una rejilla de tamaño 8x8 formada por cubos de igual tamaño alternando los colores negro y blanco.

Sigamos con el modelado de las distintas piezas que reviste un poco más de complejidad. Para cada pieza veremos un dibujo de la misma seguido de la descripción de cómo se ha llevado a cabo:

Nota: cada pieza lleva asociada una llamada al método `base()`, este procedimiento se encarga del modelado de la parte común a todas las piezas, su funcionamiento es el siguiente:

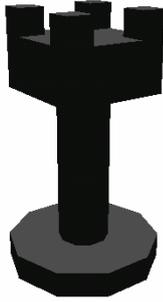
```
Cilindro(3.0,1.0);
glPushMatrix();
  glTranslatef(0.0,1.0,0.0);
  Cilindro(1.0,altura);
glPopMatrix();
```

- **Peon:**



```
glPushMatrix();
  Base(6.0);
  glPushMatrix();
    glTranslatef(0.0,7.0,0.0);
    Cilindro(2.0,0.5);
  glPopMatrix();
  glPushMatrix();
    glTranslatef(0.0,9.0,0.0);
    Esfera(2.0);
  glPopMatrix();
glPopMatrix();
```

- **Torre:**



```

Base(6.0);
glPushMatrix();
  glTranslatef(-2.0,7.0,-2.0);
  Box(4.0,2.0,4.0);
glPopMatrix();
glPushMatrix();
  glTranslatef(-2.0,9.0,-2.0);
  Box(1.0,1.0,1.0);
glPopMatrix();
glPushMatrix();
  glTranslatef(-2.0,9.0,1.0);
  Box(1.0,1.0,1.0);
glPopMatrix();
glPushMatrix();
  glTranslatef(1.0,9.0,1.0);
  Box(1.0,1.0,1.0);
glPopMatrix();
glPushMatrix();
  glTranslatef(1.0,9.0,-2.0);
  Box(1.0,1.0,1.0);
glPopMatrix();

```

- **Caballo:**



```

Base(6.0);
glPushMatrix();
  glTranslatef(-2.0,4.5,-1.5);
  Box(4.0,2.5,3.0);
  glTranslatef(-2.0,2.5,0.0);
  Box(6.0,3.5,3.0);
glPopMatrix();
glPushMatrix();
  glTranslatef(-2.0,8.5,-1.5);
  Esfera(1.0);
  glTranslatef(0.0,0.0,3.0);
  Esfera(1.0);
glPopMatrix();

```

- **Arfil:**



```

Base(6.0);
glPushMatrix();
  glTranslatef(0.0,7.0,0.0);
  glScalef(1.0,1.25,1.0);
  Esfera(2.0);
glPopMatrix();
glPushMatrix();
  glTranslatef(0.0,9.5,0.0);
  Esfera(1.0);
glPopMatrix();

```

- **Reina:**



```
glPushMatrix();
Base(6.0);
glPushMatrix();
glTranslatef(0.0,7.0,0.0);
Cilindro(2.0,2.0);
glTranslatef(0.0,1.0,0.0);
Cilindro(2.5,0.5);
Esfera(2.0);
glTranslatef(0.0,1.5,0.0);
Esfera(1.0);
glPopMatrix();
glPopMatrix();
```

- **Rey:**



```
glPushMatrix();
Base(6.0);
glPushMatrix();
glTranslatef(0.0,7.0,0.0);
Cilindro(2.0,1.0);
glTranslatef(0.0,1.0,0.0);
Cilindro(2.5,1.0);
glTranslatef(0.0,1.0,0.0);
Esfera(2.0);
glPopMatrix();
glPushMatrix();
glTranslatef(-0.5,10.0,-0.25);
Box(1.0,2.5,0.5);
glTranslatef(-0.25,1.0,0.0);
Box(1.5,1.0,0.5);
glPopMatrix();
glPopMatrix();
```

3. Interacción.

La interacción con el modelo se lleva a cabo a través del teclado y del ratón. Con el teclado controlaremos los movimientos de cámara sobre el tablero, control de zoom, etc... En cuanto al juego se controla con el ratón.

Cada conjunto *pieza* + *casilla* del tablero lleva asociado un identificador, de esta forma podemos seleccionar “función *pick*” las distintas piezas y moverlas por el tablero.

Los movimientos de las fichas deberán seguir las reglas del juego. En el apartado 5 explicaremos este paso.

4. Estructuras de datos

La estructura de datos principal consiste en una matriz de tamaño 8x8 en la que se almacena toda la información sobre la posición de las fichas dentro del tablero.

Cada posición de la matriz contiene dos valores: la ficha que se encuentra en esa casilla del tablero (0 si está libre) y su color correspondiente.

5. Control del juego

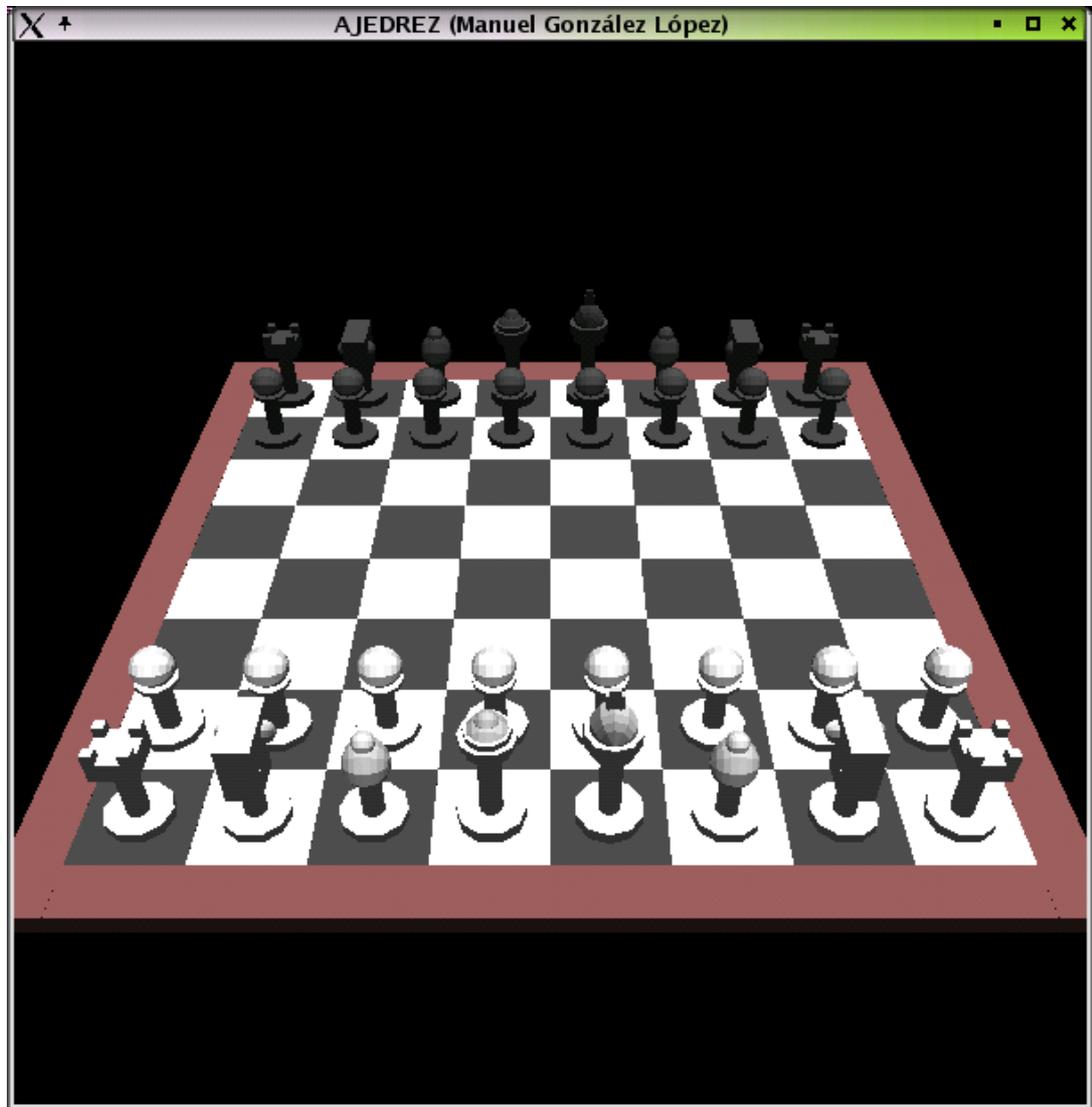
El control del juego forma parte, de cierto modo, de la interacción. Cuando intentamos mover una ficha de lugar debemos comprobar distintas cosas: turno de juego, movimiento correcto, jaque (mate), ... Pasamos a detallar estas partes:

- **Turno de juego:** Sólo necesitamos de una variable global que nos indique si el turno es de las blancas o de las negras. No podremos seleccionar una ficha que sea de color distinto al indicado por el turno.
Como comprobará, para saber a quién le toca jugar, rotamos el tablero de forma que las fichas que deben jugar aparecen en la parte inferior del monitor (más cerca del usuario).
- **Control de movimiento:** El movimiento de una ficha se lleva a cabo en dos fases: primero seleccionamos la pieza que queremos mover y segundo colocamos dicha ficha en su posición final. Entre estos dos pasos es donde se controla si el movimiento es correcto.
El funcionamiento es el siguiente: tenemos una función para cada tipo de pieza que recibe como argumentos la posición inicial en la que se encuentra la ficha que deseamos mover, dicha función devuelve un vector dinámico con todas los posibles movimientos correctos que puede realizar, si la posición final introducida por el usuario coincide con algún elemento de este vector consideramos el movimiento correcto, en caso contrario, el usuario deberá repetir la jugada.
- **Jaque y jaque mate:** Además de comprobar si un movimiento es correcto, también tenemos que verificar cuando un jugador pone en jaque al adversario. La forma de controlar el jaque es sencilla: con la ayuda de los métodos explicado en el control de movimientos, podemos ver si alguna ficha del adversario puede alcanzar nuestro rey, si es así estaremos en posición de jaque. En cuanto al jaque mate el funcionamiento es similar: generamos los posibles movimientos del rey y, para cada uno de ellos, vemos si seguimos estando en jaque. Si no podemos salir del jaque (jaque mate) la partida habrá concluido. NOTA: no podremos realizar ningún movimiento que ponga a nuestra rey en jaque, y si el adversario pone en jaque a nuestro rey, estaremos obligados a salir de este estado antes de continuar la partida.

Nota: Los mensajes indicando el estado del juego como movimientos incorrectos, jaques, etc... se muestran en la consola (intérprete de comandos) desde la que se lanzó el juego. Veremos esto con más detalle en el manual de usuario.

6. Manual de Usuario.

El aspecto de la ventana de juego es el siguiente:



Compilación

Para compilar el programa, ejecute el comando *make*, esto procesará el fichero *makefile* y creará un ejecutable llamada *ajedrez*, para lanzar el programa sólo tendrá que ejecutar dicho comando.

Juego

Empezarán a jugar las fichas blancas, el turno cambia cuando se realiza un movimiento correcto. El movimiento de las fichas sigue las reglas del juego del ajedrez. Para mover una ficha pinche sobre ella (botón izquierdo del ratón) y a continuación seleccione la casilla de destino, si el movimiento es correcto la pieza se moverá, en caso

contrario deberá realizar otra jugada. Si realiza un movimiento no permitido aparecerá un mensaje en la consola indicándolo. Si el adversario consigue poner en jaque a su rey aparecerá el correspondiente mensaje, seguidamente deberá realizar una jugada que le permita salir de este estado antes de continuar con el normal desarrollo del juego. La partida acaba cuando un jugador consigue poner en jaque mate al adversario, en este caso el ganador también se indica en la consola.

Para acceder al menú de juego pulse el botón derecho del ratón, las opciones son las siguientes:

- **Nuevo Juego:** Permite iniciar una nueva partida.
- **Salir:** Permite abandonar el programa, equivalente a pulsar la tecla ESCape.

Con la ayuda del teclado podrá realizar los movimientos de cámara que considere oportunos. A través de las teclas de cursor ←, ↑, →, ↓, podrá controlar las rotaciones en sentido X e Y mientras que con las teclas +, -, se controla el zoom (+ para acercarse al tablero, - para alejarse).
