



**Universidad
Europea**

Memoria realización ejercicio de las damas españolas

Arturo Alba Sánchez-Mayoral

Nº expediente: 21838473

Repo GitHub: <https://github.com/ArtySaurio/DamasExtr>



MODIFICACIONES AL CÓDIGO DE LAS DAMAS ESPAÑOLAS

Lo primero que hay que realizar es definir, mediante la función define, las damas negras y las damas blancas. También se les asigna un número para poder llamarlos posteriormente en el código.

```
9    #define COLUMNAS 8
10   #define DAMAB 4
11   #define DAMAN 5
```

Después de declarar la definición de las damas blancas y negras habrá que entrar al switch dentro de la función de símbolos. De esta manera se castearán 4 por las damas blancas y un 5 por las damas negras.

```
271     switch(i)
272     {
273         case 0:
274             return ' ';
275         case 1:
276             return 'v';
277         case 2:
278             return 'B';
279         case 3:
280             return 'N';
281         case 4:
282             return 'X';
283         case 5:
284             return 'O';
285     }
```

Asignaremos a las damas blancas como "X" y las damas negras "O".

También no hay que olvidarse de crear una condición para detectar si la pieza que se selecciona es o una ficha blanca o una ficha negra. Esto se debe a que antes de implementar las damas solo había que comprobar 2 tipos de piezas, habiendo añadido 2 piezas mas (damas negras y damas blancas) siendo un total de 4 piezas.

```
113     //Comprueba si es un desplazamiento normal, una sola posicion
114     if(turnada == FICHAS_BLANCAS || turnada == FICHAS_NEGRAS){
115         if(filaOrigen - filaDestino == -1 || filaOrigen - filaDestino == 1){
116             if(colOrigen - colDestino == 1 || colOrigen - colDestino == -1){
117                 intercambio(matrix, filaOrigen, colOrigen, filaDestino, colDestino);
118                 return 0;
119             }
120         }
121     }
```

Una vez arreglado la definición de las damas y su inclusión en el switch de simbolos, habrá que realizar cambios en la función “intercambio”. Habrá que programar el momento cuando una ficha blanca normal se convierte en una dama. Esto lo haremos teniendo en cuenta que las fichas negras deberán llegar al borde del tablero (fila 0) y que las fichas blancas deberán llegar al otro borde del tablero (fila 7).

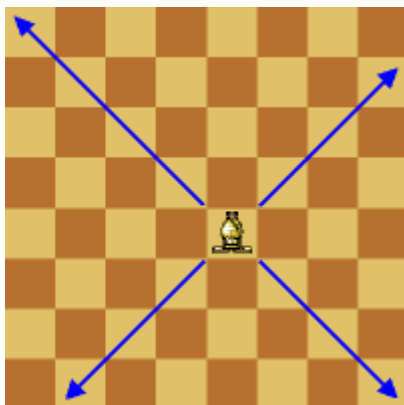
```

252 void intercambio(int matrix[FILAS][COLUMNAS], int filaOrigen, int colOrigen, int filaDestino, int colDestino){
253     int temp;
254
255     if ((matrix[filaOrigen][colOrigen]==FICHAS_BLANCAS) && (filaDestino==7)){
256         temp = matrix[filaOrigen][colOrigen];
257         matrix[filaOrigen][colOrigen] = matrix[filaDestino][colDestino];
258         matrix[filaDestino][colDestino] = DAMAB;
259     }else if ((matrix[filaOrigen][colOrigen]==FICHAS_NEGRAS) && (filaDestino==0)){
260         temp = matrix[filaOrigen][colOrigen];
261         matrix[filaOrigen][colOrigen] = matrix[filaDestino][colDestino];
262         matrix[filaDestino][colDestino] = DAMAN;
263     }else{
264         temp = matrix[filaOrigen][colOrigen];
265         matrix[filaOrigen][colOrigen] = matrix[filaDestino][colDestino];
266         matrix[filaDestino][colDestino] = temp;
267     }
268 }

```

Ahora habrá que codificar el movimiento de las damas previamente definidas. Investigando el funcionamiento de las damas españolas, descubrí que el tipo de movimiento de las damas es bastante similar al del alfil en el ajedrez. Como en el movimiento del alfil, el movimiento de las damas se tendrá que codificar teniendo en cuenta 4 opciones; diagonal arriba izquierda, diagonal arriba derecha, diagonal abajo izquierda y diagonal abajo derecha.

Hablando de código, deberemos crear un sistema para primero comprobar que la pieza que se quiere mover es una dama mediante un if. Después se creará una variable, esta variable almacenará el tipo de movimiento que tendrá (de los 4 movimientos diagonales definidos anteriormente).



```

127 if(turnada == DAMAB || turnada == DAMAN){
128     if((abs(filaOrigen - filaDestino) == (abs(colOrigen - colDestino)))){
129         int diago; //Creamos variable para ver a la diagonal donde se moverá la dama
130         if((filaOrigen > filaDestino) && (colOrigen > colDestino)){//desplazamiento diagonal arriba-izquierda
131             diago = 1;
132         }else if((filaOrigen < filaDestino) && (colOrigen < colDestino)){//Desplazamiento diagonal abajo-derecha
133             diago = 2;
134         }else if((filaOrigen < filaDestino) && (colOrigen > colDestino)){//Desplazamiento diagonal abajo-izquierda
135             diago = 3;
136         }else if((filaOrigen > filaDestino) && (colOrigen < colDestino)){//Desplazamiento diagonal arriba-derecha
137             diago = 4;
138         }

```

En cada tipo de movimiento diagonal, se verá mediante un for, la trayectoria que la dama querrá recorrer. De esta manera detectará si hay alguna otra ficha en medio, habrá que codificar que, en esta ocasión, se comerá la ficha de en medio, dejando ese espacio vacío (1) y después se utilizará la función “intercambio” para poner la dama en el sitio donde desee el usuario.

```
174         }else if(diago == 4){ //desplazamiento diagonal arriba-derecha
175             for(int i = filaOrigen, j= colOrigen; (filaOrigen != filaDestino) && (colOrigen != colDestino); i--,j++){//Recorre
176                 if (((matrix[i][j] == FICHAS_NEGRAS)|| (matrix[i][j] == DAMAN))&&(matrix[i - 1][j + 1] == 1)){//Verificamos si
177                     matrix[i][j] = 1; //Se come a la ficha y deja una posición vacía
178                     intercambio(matrix, filaOrigen, colOrigen, i - 1, j + 1); //Realizamos el intercambio
179                     return 0;
180                 }else if (((matrix[i][j] == FICHAS Blancas)|| (matrix[i][j] == DAMAB))&&(matrix[i - 1][j + 1] == 1)){//Verifica
181                     matrix[i][j] = 1; //Se come a la ficha y deja una posición vacía
182                     intercambio(matrix, filaOrigen, colOrigen, i - 1, j + 1); //Realizamos el intercambio
183                     return 0;
184                 }
185             }
186         }
```

Para terminar creamos, dentro de la función “main”, un sistema que detecte si la ficha que el jugador va mover cuando sea su turno. De esta manera podrá utilizar correctamente el programa los movimientos posibles con la pieza seleccionada. También, por supuesto, habrá que asegurarse de que el jugador no pueda mover una pieza fuera del tablero.

```
335         if(matrix[filaOrigen - 1][colOrigen - 'a'] == FICHAS_NEGRAS){
336             turnada = FICHAS_NEGRAS;
337         }else if(matrix[filaOrigen - 1][colOrigen - 'a'] == DAMAN){
338             turnada = DAMAN;
339         }else{
340             turnada = -1; //Produce un turnada erroneo en caso de fallo
341         }
```

Exactamente la misma función se realizaría para las fichas blancas.