

Manual Técnico

El juego cuenta con 2 archivos distintos, el primero se llama *dam.asm* y es el que lleva el código fuente del juego, el segundo se llama *macro5.asm* y es el encargado de las macros repetitivas del código fuente del juego.

Dam.asm:

En el segmento de datos, se encuentran todas las variables y etiquetas utilizadas para la creación correcta del juego, para un mejor control cada una va etiquetada con su finalidad en el juego.

```
===== SEGMENTO DE DATOS =====
.data
;----- PARA LOS REPORTES -----
bufferentrada db 50 dup('$')
handlerentrada dw ?
bufferinformacion db 200 dup('$')

br db 0ah, 0dh, '<br>', '$'
bj1 db 0ah, 0dh, ' Blancas ', '$'
nj2 db 0ah, 0dh, ' Negras ', '$'
html db 0ah, 0dh, '<html><body><center>', '$'
encabezado db 0ah, 0dh, ' <h1><br><br> Reporte del Juego </h1><br><br> <table>', '$'
ttr1 db 0ah, 0dh, '<tr><td>Jugadores</td><td>Fichas</td><td>Punteo</td></tr>', '$'
ttr db 0ah, 0dh, '<tr>', '$'
ttd db 0ah, 0dh, '<td>', '$'
ttdc db 0ah, 0dh, '</td>', '$'
ttrc db 0ah, 0dh, '</tr>', '$'
encabezado2 db 0ah, 0dh, ' <h3> <br><br> Tablero actual del juego </h3>', '$'

table2 db 0ah, 0dh, ' <table border=1>', '$'
table2c db 0ah, 0dh, ' </table>', '$'
tab db 0ah, 0dh, ' <td></td><td>A</td><td>B</td><td>C</td><td>D</td><td>E</td><td>F</td><td>G</td><td>H</td>'

n1 db 49, '$'
n2 db 50, '$'
n3 db 51, '$'
n4 db 52, '$'
n5 db 53, '$'
n6 db 54, '$'
n7 db 55, '$'
n8 db 56, '$'
aux11 db 0, '$'
siEsRepo11 db 0, '$'
siEsRepo12 db 0, '$'

htmlc db 0ah, 0dh, ' </table></center></body></html>', '$'
```

En el segmento de código, se encuentra el flujo del juego, como principal el despliegue del menú, luego el llenado del tablero y dependiendo de la opción elegida del menú, saltara hasta esa etiqueta que contendrá el flujo de código que corresponde.


```

EscribirArchivo:
    print imprimir
    limpiar bufferInformacion, SIZEOF bufferInformacion,24h
    print ing2
    escribir handlerentrada, html, SIZEOF html
    escribir handlerentrada, encabezado, SIZEOF encabezado
    escribir handlerentrada, ttr1, SIZEOF ttr1
    escribir handlerentrada, ttr, SIZEOF ttr
    escribir handlerentrada, ttd, SIZEOF ttd
    escribir handlerentrada, arregloJugador, SIZEOF arregloJugador
    escribir handlerentrada, ttdc, SIZEOF ttdc
    escribir handlerentrada, ttd, SIZEOF ttd
    escribir handlerentrada, nj2, SIZEOF nj2
    escribir handlerentrada, ttdc, SIZEOF ttdc
    escribir handlerentrada, ttd, SIZEOF ttd
    escribir handlerentrada, punteoJ1, SIZEOF punteoJ1
    escribir handlerentrada, ttdc, SIZEOF ttdc
    escribir handlerentrada, ttrc, SIZEOF ttrc
    escribir handlerentrada, ttr, SIZEOF ttr
    escribir handlerentrada, ttd, SIZEOF ttd
    escribir handlerentrada, arregloJugador2, SIZEOF arregloJugador2
    escribir handlerentrada, ttdc, SIZEOF ttdc
    escribir handlerentrada, ttd, SIZEOF ttd
    escribir handlerentrada, bj1, SIZEOF bj1
    escribir handlerentrada, ttdc, SIZEOF ttdc
    escribir handlerentrada, ttd, SIZEOF ttd
    escribir handlerentrada, punteoJ2, SIZEOF punteoJ2
    escribir handlerentrada, ttdc, SIZEOF ttdc
    escribir handlerentrada, ttrc, SIZEOF ttrc
    escribir handlerentrada, table2c, SIZEOF table2c

    escribir handlerentrada, br, SIZEOF br
    escribir handlerentrada, br, SIZEOF br
    escribir handlerentrada, br, SIZEOF br
    escribir handlerentrada, encabezado2, SIZEOF encabezado2
    escribir handlerentrada, table2, SIZEOF table2

```

Macro5.asm:

Este archivo es muy importante, ya que sin el, no funcionaria el juego.

Se divide en varias macros, todas son importantes.

Las primeras tres macros son las encargadas de recibir cadenas, imprimir cadenas y para limpiar pantalla.

```

;----- MACRO PARA IMPRIMIR -----
print macro buffer
    mov ax, @data
    mov ds, ax
    mov ah, 09h
    mov dx, offset buffer
    int 21h
endm

;----- MACRO PARA LEER CARACTER OBTENIDO -----
getChar macro
    mov ah, 01h
    int 21h
    ;al registro AL se le manda el codigo ascii recibido, en codigo hexadecimal
endm

;----- MACRO PARA LIMPIAR PANTALLA -----
LimpiarConsola macro
    mov ax, 0600h
    mov bh, 89h
    mov cx, 0000h
    mov dx, 184Fh
    int 10h
endm

```

Las 2 macros más extensas son las encargadas de validar los movimientos para los 2 jugadores, ya sean fichas normales o fichas reinas. Estas macros verifican que los jugadores no muevan las fichas de su oponente, tambien que sus fichas se muevan 1 casilla en diagonal a la vez y que las fichas reinas tengan los movimientos requeridos.

```

MacroMueveJ1 macro
LOCAL PintaJugadaJ1, PintaJugadaRJ1, MovInvalido, sale2, Mensaje

    cmp al, 110 ;compara si es n
    je Mueve
    cmp al, 78 ;compara si es N
    je PintaJugadaRJ1
    jmp MensajeError

Mueve:
    cmp si, 1h ;1 en hexa
    je Mueve1
    cmp si, 3h ;3 en hexa
    je Mueve3
    cmp si, 5h ;5 en hexa
    je Mueve5
    cmp si, 7h ;7 en hexa
    je Mueve7
    cmp si, 0Ah ;10 en hexa
    je Mueve10
    cmp si, 0Ch ;12 en hexa
    je Mueve12
    cmp si, 0Eh ;14 en hexa
    je Mueve14
    cmp si, 10h ;16 en hexa
    je Mueve16
    cmp si, 11h ;17 en hexa
    je Mueve17
    cmp si, 13h ;19 en hexa
    je Mueve19
    cmp si, 15h ;21 en hexa
    je Mueve21
    cmp si, 17h ;23 en hexa
    je Mueve23
    cmp si, 1Ah ;26 en hexa

```