

```

;; Este módulo contiene el loop del juego.
.module    game

;; Cadenas
comienzo_tabla:
    .ascii    "\n\n | JUEGO |\n"
    .ascii    "-----\n"
    .ascii    " | 12345 |\n"
    .asciz    "-----\n"
fin_linea:
    .asciz    " |\n"
comienzo_linea:
    .asciz    " | "
linea_vacia:
    .asciz    " |          |\n"
win:         .asciz    "HAS ACERTADO LA PALABRA\n"
loss:        .asciz    "HAS TERMINADO TUS INTENTOS\n"

palabra_no_en_diccionario:
    .asciz    "\n\nLa palabra no se encuentra en el diccionario."

;; Globales
    .globl    game
    .globl    lee_palabra
    .globl    imprime_cadena
    .globl    compara_palabras
    .globl    imprime_cadena_color
    .globl    imprime_cadena_wordle
    .globl    palabra_en_diccionario

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
; game_loop:
;     Ejecuta un juego de wordle
;
; Entrada: X-dirección de comienzo de la palabra correcta
; Salida: A-Resultado: 0: ningún problema, r: reset, v: vuelta al menú
; Afecta: X,A
;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
game:
    pshs      b,x,y

    ;; Se establece A = -1 para imprimir una tabla vacía
    lda      #-1
    lbr      imprime_tabla
    lda      #0

game_loop:
    cmpa     #6                                ;; s:
    lbeq     game_loss
    pshs     a                                ;; s: a

    ;; Pido palabra
    ldb      #5
    mul
    pshs     y                                ;; s: a - y
    leay     b,y
    pshs     a                                ;; s: a - y - a2
    lbr      lee_palabra

    ;; Compruebo que la palabra se ha leído bien
    cmpa     #0
    lbne     game_end_mal

    puls     a                                ;; s: a - y

    ;; Digo si la palabra estaba en el diccionario
    lbr      palabra_en_diccionario
    puls     y                                ;; s: a
    cmpa     #0
    beq      imprime

```

```

        pshs      x                ;; s: a - x
        ldx      #palabra_no_en_diccionario
        lbsr     imprime_cadena
        puls     x                ;; s: a

imprime:
        ;; Imprime la tabla
        puls     a                ;; s:
        lbsr     imprime_tabla

        ;; Compruebo si la palabra era la correcta
        pshs     a,y              ;; s: y - a

        ldb      #5
        mul
        leay     b,y
        lbsr     compara_palabras
        cmpa     #0

        puls     a,y              ;; s:

        beq      game_win

        ;; Incremento el loop
        inca
        lbra     game_loop

game_end_mal:
        ;; Quedaban bytes de variables en el stack, pero no los necesitamos
        leas     4,s
        puls     b,x,y,pc

game_loss:
        ldx      #loss
        lda      #0
        bra      game_end

game_win:
        ldx      #win
        lda      #2

game_end:
        ;; Imprimo la cadena de resultado
        lbsr     imprime_cadena_color
        lda      #0
        puls     b,x,y,pc

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
; imprime_tabla
;   imprime la tabla de wordle con las palabras escritas anteriormente
; Entrada: X-Palabra correcta
;         Y-Puntero a la lista de palabras introducidas
;         A-Palabra en la que nos encontramos
; Salida: Ninguna
; Afecta: Nada
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
imprime_tabla:
        pshs     b, a                ;; s: a

        ;; Imprimo la cabeza de la tabla
        pshs     x                ;; s: a - x
        ldx      #comienzo_tabla
        lbsr     imprime_cadena
        puls     x                ;; s: a

        lda      #0
bucle_anterior:
        cmpa     ,s
        bgt      bucle_anterior_end
        pshs     a                ;; s: a - a2

        ;; Imprimo el número de línea
        adda     #'1

```

```

    sta      0xFF00
    suba     #'1

;; Imprimo el comienzo de linea
    pshs     x                      ;; s: a - a2 - x
    ldx      #comienzo_linea
    lbsr     imprime_cadena
    puls     x                      ;; s: a - a2

;; Imprimo la palabra
    ldb      #5
    mul
    pshs     y                      ;; s: a - a2 - y
    leay     b,y
    lbsr     imprime_cadena_wordle
    puls     y                      ;; s: a - a2

;; Imprimo el fin de linea
    pshs     x                      ;; s: a - a2 - x
    ldx      #fin_linea
    lbsr     imprime_cadena
    puls     x                      ;; s: a - a2

;; Incremento
    puls     a                      ;; s: a
    inca
    bra      bucle_anterior

bucle_anterior_end:
    ;; Imprimo líneas vacías
    pshs     x                      ;; s: a - x
    ldx      #linea_vacia
bucle_next:
    cmpa     #6
    beq      bucle_next_end
    adda     #'1
    sta      0xFF00
    suba     #'1
    lbsr     imprime_cadena
    inca
    bra      bucle_next

bucle_next_end:
    puls     x                      ;; s: a

    ldb      #10
    stb      0xFF00

    puls     a,b,pc

```

```
.module diccionario

.globl palabras

palabras:
    .ascii "MOSCA"
    .ascii "PULPO"
    .ascii "GANSO"
    .ascii "LLAMA"
    .ascii "HIENA"
    .ascii "LEMUR"
    .ascii "CERDO"
    .ascii "CISNE"
    .ascii "CARPA"
    .ascii "CABRA"
    .ascii "ERIZO"
    .ascii "GALLO"
    .ascii "TIGRE"
    .ascii "CEBRA"
    .ascii "OVEJA"
    .ascii "PERRO"
    .ascii "PANDA"
    .ascii "KOALA"
    .byte 0
```

```
;; Librería con funciones, principalmente de lectura y escritura
.module    aslib

colors:    .ascii    "\33[31m\0\0\0"        ;; Estos \0 son una mickey
           .ascii    "\33[33m\0\0\0"        ;; herramienta sorpresa que
           .ascii    "\33[32m\0\0\0"        ;; nos ayudará más adelante
           .ascii    "\33[37m\0"

bold:      .asciz     "\33[1m"
normal:    .asciz     "\33[0m"
clear:     .asciz     "\33[2J\33[1;1H"

cadena_leer:
           .asciz     "\nPALABRA: "

           .globl     imprime_caracter_color
           .globl     palabra_en_diccionario
           .globl     imprime_cadena_wordle
           .globl     imprime_valor_decimal
           .globl     imprime_cadena_color
           .globl     compara_palabras
           .globl     imprime_palabra
           .globl     imprime_cadena
           .globl     lee_palabra

           .globl     bold
           .globl     normal
           .globl     clear

           .globl     palabras

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
; imprime_valor_decimal:
;     imprime el valor del registro a por pantalla en decimal
; Entrada: A-el valor a imprimir
; Salida: Ninguna
; Afecta: Nada
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
imprime_valor_decimal:
    pshs        a,b

    ;; Divido entre 10 y saco el cociente y resto
    ldb         #10
    pshs        b
    ldb         #0

ivl_divide_bucle:
    cmpa        ,s
    blo         ivl_divide_fin
    incb
    suba        ,s
    bra         ivl_divide_bucle

ivl_divide_fin:
    exg         a,b
    leas        1,s        ;; pull, pero sin hacer pull

    ;; Primera cifra
    adda        #'0
    sta         0xFF00

    ;; Segunda cifra
    addb        #'0
    stb         0xFF00

    puls        a,b,pc

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
; imprime_cadena:
;     imprime por pantalla la cadena apuntada por X
;
```

[illegible]

```

        puls      b
        bne       icw_int_inc
        ;; Si lo son, miro si están en el mismo sitio
        cmpb      ,s
        bne       icw_distinto_sitio
        ;; Si lo están, sumo 6 a A y salto al fin del bucle, si no, solo 1
        adda      #6
        bra       icw_int_fin
icw_distinto_sitio:
        inca
icw_int_inc:
        incb
        leax      1,x
        bra       icw_int_inicio
icw_int_fin:
        ;; Comparo A con 6 para saber si la letra estaba en el mismo sitio
        cmpa      #6
        blo       icw_lo1
        lda       #2
        bra       icw_ext_inc
        ;; Si no, comparo con 0 para ver si estaba en la palabra
icw_lo1:   cmpa      #0
        beq       icw_ext_inc
        lda       #1
icw_ext_inc:
        ;; Imprimo el caracter a color
        ldb       ,y+
        lbsr      imprime_caracter_color
        puls      a,x
        inca
        bra       icw_ext_inicio
icw_ext_fin:
        puls      a,b,y,pc

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
; imprime_palabra                                                    ;
;     imprime por pantalla la palabra de 5 letras a la que apunta X ;
; Entrada: X-palabra                                                ;
; Salida: Ninguna                                                  ;
; Afecta: Nada                                                     ;
;                                                                     ;
;                                                                     ;
imprime_palabra:
        pshs      a,b
        lda       #0
ip_bucle:
        cmpa      #5
        beq       ip_return
        ldb       a,x
        stb       0xFF00
        inca
        bra       ip_bucle
ip_return:
        puls      a,b,pc

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
; imprime_caracter_color:                                           ;
;     imprime por pantalla el caracter en b con el color que se especifique ;
;     a: (0: rojo, 1: amarillo, 2: verde, 3: blanco,                ;
;         >3: mantiene color anterior)                               ;
;                                                                     ;
; Entrada: B-caracter a imprimir                                    ;
;         A-color a usar                                            ;
; Salida: Ninguna                                                  ;
; Afecta: A                                                         ;
;                                                                     ;
;                                                                     ;
imprime_caracter_color:
        pshs      x
        cmpa      #3
        bhi       ic_caracter

```

```

        ldx      #colors
        lsla lsla lsla      ;; Mickey
        leax     a,x
        lbrs     imprime_cadena

ic_caracter:
        stb      0xFF00
        ldx      #colors
        leax     24,x
        lbrs     imprime_cadena
        puls     x,pc

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
; compara_palabras:
;   compara las palabras apuntadas por x e y
;
; Entrada: X-primera cadena
;          Y-segunda cadena
; Salida:  A-resuladado. 0=iguales, 1=distintos
; Afecta:  A
;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
compara_palabras:
        pshs     b, x, y

        lda      #0
        ldb      #0
        ;; Recorro ambas palabras a la vez

cp_bucle:
        cmpb     #5
        beq      cp_ret
        incb

        pshs     a
        lda      ,x+
        cmpa     ,y+
        puls     a
        beq      cp_bucle

        ;; Si en algún caracter las letras son distintas, devuelvo falso
        lda      #1
        bra      cp_ret

cp_ret:
        puls     b, x, y, pc

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
; palabra_en_diccionario:
;   Comprueba si la palabra dada se encuentra en el diccionario
; Entrada: Y-Palabra a comprobar
; Salida:  A-resuladado. 1=en diccionario, 0=fuera de diccionario
; Afecta:  A
;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
palabra_en_diccionario:
        pshs     b,x
        ldx      #palabras

        ;; Recorro las palabras hasta encontrar un \0

ped_bucle:
        ldb      ,x
        cmpb     #0
        beq      ped_return

        ;; Si las palabras son iguales devuelvo 1
        lbrs     compara_palabras
        cmpa     #0
        beq      ped_return
        lda      #1

```



```

        leax      5,x
        bra      ped_bucle
ped_return:
        puls     b,x,pc

;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
; lee_palabra:
;   lee una palabra de 5 letras mayúsculas por pantalla permitiendo
;   borrar la letra anterior con un espacio y la almacena en Y
;
; Entrada: Y-Dirección donde almacenar la cadena
; Salida:  Y-Cadena leída
;         A-Estado de la lectura: 0-correcta, 'v-v, 'r-r
; Afecta:  Y,A
;
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;
lee_palabra:
        pshs     b,x

        lda      #0
        ldb      #0
        ;; Limpio Y para que se imprima todo correctamente
lp_bucle_limpiar:
        cmpa     #6
        beq      lp_bucle_limpiar_fin
        stb      a,y
        inca
        bra      lp_bucle_limpiar
lp_bucle_limpiar_fin:

        ldx      #cadena_leer
        lda      #0

        lbsr     imprime_cadena

        ;; Entro en un bucle para leer las letras hasta que haya 5
lp_bucle_leer:
        cmpa     #5
        beq      lp_return

        ;; Leo caracter
        ldb      0xFF02
        ;; Compruebo que esté entre A y Z
        cmpb     #'A
        blo      lp_invalido
        cmpb     #'Z
        bhi      lp_invalido
        bra      lp_valido

lp_invalido:
        ;; Si es espacio borro
        cmpb     #32
        beq      lp_back

        ;; Si es v o r, return_mal
        cmpb     #'v
        beq      lp_return_mal

        cmpb     #'r
        beq      lp_return_mal

        ;; Si no se cumple ningún caso, ignoro el input
        lbsr     imprime_cadena
        exg      x,y
        lbsr     imprime_cadena
        exg      x,y
        bra      lp_bucle_leer

lp_valido:
        ;; Si es espacio borro
        cmpb     #32
        beq      lp_back

```

```
        stb      a,y
        inca
        bra      lp_bucle_leer
lp_back:
        ;; "Borro" la letra
        ldb      #0
        deca
        stb      a,y
        inca

        ;; Vuelvo a imprimir PALABRA: y lo escrito antes
        lbr      imprime_cadena
        exg      x,y
        lbr      imprime_cadena
        exg      y,x

        ;; Decremento la letra actual si es distinta de 0
        cmpa     #0
        beq      lp_bucle_leer
        deca

        bra      lp_bucle_leer
lp_return_mal:
        exg      a,b
        puls     b,x,pc
lp_return:
        lda      #0
        puls     b,x,pc
```

```

;; Modulo principal.
;; El menú está aquí

        .module    main
        .area      MAIN (ABS)
        .org       0x1000

;; Distintas cadenas
texto_menu:
        .ascii     "\nWORDLE\n"
        .ascii     "1) Ver Diccionario\n"
        .ascii     "2) Jugar\n"
        .asciz     "S) Salir\n"
text_volver_jugar:
        .asciz     "Seguir jugando? [y/N]"
texto_diccionario_inicio:
        .asciz     "\n\nDiccionario:\n"
texto_diccionario_final:
        .asciz     "Numero de palabras = "

;; Importaciones globales
        .globl     imprime_valor_decimal
        .globl     imprime_cadena_color
        .globl     imprime_palabra
        .globl     imprime_cadena
        .globl     palabras
        .globl     normal
        .globl     clear
        .globl     bold
        .globl     game

main:
        lds        #0xFF00

        ;; Pongo el texto en negrita
        ldx        #bold
        lbsr       imprime_cadena

        ldx        #palabras

menu:
        ;; Limpio la pantalla y muestro el menú
        pshs       x
        ldx        #clear
        lbsr       imprime_cadena
        ldx        #texto_menu
        lbsr       imprime_cadena
        puls       x

        ;; Leo la opción en b
        ldb        0xFF02

        ;; Salto a donde corresponda
        cmpb       #'1
        lbeq       diccionario
        cmpb       #'2
        lbeq       jugar
        cmpb       #'S
        lbeq       acabar
        cmpb       #'s
        lbeq       acabar

        ;; Si ninguna opción del menú coincide, vuelve al menu
        bra        menu

diccionario:
        pshs       x

        ;; Imprimo comienzo diccionario
        ldx        #texto_diccionario_inicio
        lbsr       imprime_cadena

        ;; Bucle para imprimir todas las palabras

```

```

        lda        #0
        ldb        #0
        ldx        #palabras
diccionario_bucle:
        ldb        ,x
        cmpb       #0
        beq        diccionario_fin

        ;; Imprimo palabra a palabra
        lbsr       imprime_palabra
        pshs       b
        ldb        #' \n
        stb        0xFF00
        puls       b

        inca
        leax       5,x

        bra        diccionario_bucle
diccionario_fin:
        ;; Imprimo el número de palabras
        ldx        #texto_diccionario_final
        lbsr       imprime_cadena
        lbsr       imprime_valor_decimal
        puls       x

        ldb        #' \n
        stb        0xFF00

        ldb        0xFF02
        bra        menu

jugar:
        ;; Limpio la pantalla
        pshs       x
        ldx        #clear
        lbsr       imprime_cadena
        puls       x

        ;; Empiezo un juego
        ldy        #0xF000
        lbsr       game

        ;; Compruebo si ha terminado bien
        cmpa       #0
        beq        fin_juego

        ;; Volver al menú
        cmpa       #'v
        lbeq       menu

        ;; Reinicio
        bra        jugar

fin_juego:
        ;; Incremento la palabra
        leax       5,x
        lda        ,x
        cmpa       #0
        lbne       volver_a_jugar
        ldx        #palabras
        lbra       volver_a_jugar

volver_a_jugar:
        ;; Pregunto si se vuelve a jugar
        pshs       x
        ldx        #text_volver_jugar
        lbsr       imprime_cadena
        puls       x

        ;; y/n
        lda        0xFF02
        cmpa       #'y

```

```
        lbne      menu

        lbra      jugar

acabar:
        ;; Quito la negrita
        ldx      #normal
        lsr      imprime_cadena

        ldb      #10
        stb      0xFF00

        ;; Fin
        clra
        sta      0xFF01
        .org     0xFFFE
        .word    main
```