

```

;; 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:
    push    a,b

    ;; Divido entre 10 y saco el cociente y resto
    ldb     #10
    push    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
;
;

```

```
; Entrada: X-dirección de comienzo de la cadena  
; Salida: Ninguna  
; Afecta: X  
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;  
imprime_cadena:  
    pshs        a,x  
ic_sgte:  
    lda         ,x+  
    cmpa        #0  
    beq         ic_return  
    sta         0xFF00  
    bra         ic_sgte  
ic_return:  
    puls        a,x,pc  
  
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;  
; imprime_cadena_color:  
;     imprime por pantalla la cadena apuntada por X en el color  
;     especificado en a  
; Entrada: X-dirección de comienzo de la cadena  
;          A-color a usar (0 - 3) rojo, amarillo, verde, blanco  
; Salida: Ninguna  
; Afecta: A  
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;  
imprime_cadena_color:  
    pshs        x  
    ldx         #colors  
    lsll lsll lsll      ;; Mickey herramienta usada aquí,  
    leax        a,x      ;; explicada en el pdf.  
    lbsr        imprime_cadena  
    puls        x  
  
    lbsr        imprime_cadena  
  
    pshs        x  
    ldx         #colors+24   ;; 8 * 3 = 24  
    lbsr        imprime_cadena  
    puls        x,pc  
  
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;  
; imprime_cadena_wordle:  
;     imprime la cadena de texto en y comparándola con la  
;     cadena en x con los colores de wordle  
; Entrada: X, Y -Cadenas  
; Salida: Ninguna  
; Afecta: Nada  
;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;;  
imprime_cadena_wordle:  
    pshs        a,b,y  
    lda         #0  
  
    ;; Bucle para recorrer la palabra sugerida ( Y )  
icw_ext_inicio:  
    cmpa        #5  
    beq         icw_ext_fin  
    pshs        a,x  
  
    lda         #0  
    ldb         #0  
    ;; Bucle para recorrer la palabra correcta ( X )  
icw_int_inicio:  
    cmpb        #5  
    beq         icw_int_fin  
    pshs        b  
  
    ;; Comparo si son iguales  
    ldb         ,x  
    cmpb        ,y
```

```

        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
        lbr      imprime_cadena

ic_caracter:
        stb      0xFF00
        ldx      #colors
        leax     24,x
        lbr      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
        lbr      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
```