SOLUCIONES A LAS ACTIVIDADES DE LA PRÁCTICA 5: INTERRUPCIONES BIOS DEL TECLADO

NOTA: las soluciones a las actividades son propuestas orientativas ya que a la hora de programar no existe una única solución; plagiar la solución no tiene ningún sentido, utilícela para corregir algún error o mejora su versión

Las soluciones a las actividades que figuran en este documento utilizan la biblioteca de macros que adjuntamos seguidamente:

Biblioteca de macros

```
;-----------------------------------------------
; macros.inc
;-----------------------------------------------
; - Biblioteca de macros.
;-----------------------------------------------

;-----------------------------------------------
; @iniciarDS: carga @DATA en DS
; sin parámetros
; no devuelve nada
; registros: AX, DS
;-----------------------------------------------
@iniciarDS          MACRO
                    mov ax,@DATA
                    mov ds,ax
                    ENDM

;-----------------------------------------------
; @fincodigo: invoca servicio 4Ch-INT21h
; sin parámetros
; devuelve: código de retorno en AL
; registros: AX
;-----------------------------------------------
@fincodigo          MACRO
                    mov ax,4C00h
                    int 21h
                    ENDM

;-----------------------------------------------
; @mostrarcadena: invoca servicio 09h-INT21h
; parámetros:
; pcadena -> puntero de cadena finalizada en '$'
; devuelve: 24h en AL
; registros: AX, DX
;-----------------------------------------------
@mostrarcadena      MACRO pcadena

                    IF TYPE (pcadena) EQ 2     ;; si pcadena word a puntero near
                           mov dx, pcadena
                    ELSEIF TYPE (pcadena) EQ 4 ;; si pcadena doble a puntero far
                           lds dx, pcadena
                    ELSE
                           .ERR
                           %OUT parametro ilegal
                    ENDIF
```

```
                    mov ah,09h
                    int 21h

                    ENDM

;-------------------------------------------------
; @entero2hex: obtiene 2 dígitos hexadecimales de
; un entero de tamaño byte
; parámetros:
; entrada -> operando de tamaño byte
; salida -> buffer de memoria de tamaño word
; devuelve: 2 dígitos hexadecimales en el buffer
; registros: AX, CL
;-------------------------------------------------
@entero2hex         MACRO entrada, salida

                    ;;si entrada byte y salida word
                    IF (TYPE entrada EQ 1) AND (TYPE salida EQ 2)
                           MOV    AL, entrada
                           MOV    AH, AL
                           AND    AL, 0Fh       ;;dígito de menor peso
                           MOV    byte ptr salida, AL
                           MOV    CL, 4
                           SHR    AH, CL        ;;dígito de mayor peso
                           MOV    byte ptr salida[1], AH
                    ELSE
                           .ERR
                           %OUT parametro ilegal
                    ENDIF
                    ENDM

;-------------------------------------------------
; @digito2ASCII: obtiene el ASCII de un dígito
; parámetros:
; entrada -> operando de tamaño byte
; salida -> posición de memoria
; devuelve: el ASCII del dígito
; registros: AL
;-------------------------------------------------
@digito2ASCII MACRO entrada, salida
                    local numero, escribir

                    ;;si entrada y salida bytes
                    IF (TYPE entrada EQ 1) AND (TYPE salida EQ 1)
                           MOV    AL, entrada
                           CMP    AL, 10
                           JB     numero
                           ADD    AL, 'A'-10
                           JMP    escribir
numero:                    ADD    AL, 030h
escribir:                  MOV    byte ptr salida, AL
                    ELSE
                           .ERR
                           %OUT parametro ilegal
                    ENDIF
                    ENDM
```

A) Acceso al teclado por sondeo.

```
;-----------------------------------------------
; pr5-ac01.asm
;-----------------------------------------------
; - Manejo del teclado.
; - Acceso al teclado por programa. Se desactivan
;   las interrupciones de manera que la rutina de
;   servicio de teclado no captura las teclas.
;   Se sondea por programa el puerto 64h. Si hay
;   pulsación o liberación de tecla se indica en LSB = 1.
;   Si no se actúa sobre el teclado, se sigue sondeando.
;   Si es el caso, se lee el carácter en el
;   puerto 60h. Cuando es pulsación MSB = 0.
;   Cuando es liberación, MSB = 1.
;   El programa termina cuando se pulsa la tecla
;   del carácter 'S' (scan code 1Fh)
;
;   Antes de finalizar, se activan las interrupciones
;   de nuevo.
;
; - Utiliza la biblioteca de macros para la
;   conversión de entero byte a hexadecimal
;-----------------------------------------------

              TITLE         pr5-ac01
              DOSSEG
              .MODEL        SMALL

              INCLUDE       macros.inc

              .STACK        100h

              .DATA
dato          DB 0
wbuffer       LABEL word
buffer        DB 0, 0
cadenaASCII   DB 0, 0, 10, 13, '$'
pcadenaASCII  DW cadenaASCII


              .CODE
inicio:       MOV    AX, @DATA
              MOV    DS, AX

              CLI

bucle:        IN     AL, 064h

              TEST   AL, 01h
              JZ     bucle

              IN     AL, 060h

              MOV    dato, AL      ;salvo el scan code

              @entero2hex dato, wbuffer
```

```
                @digito2ASCII buffer[0], cadenaASCII[1]
                @digito2ASCII buffer[1], cadenaASCII[0]

                @mostrarcadena pcadenaASCII

                CMP     dato, 01Fh    ;comparo con scan code de 'S'
                JZ      salir

volver:         JMP     bucle

salir:          STI

                MOV     AX, 4C00h
                INT     21h

                END     inicio
```

B) Acceso al teclado usando los servicios BIOS.

```
;-------------------------------------------------
; pr5-ac02.asm
;-------------------------------------------------
; - Manejo del teclado.
; - El programa sondea el buffer de teclado
;   utilizando el servicio 11h de la INT 16h.
;   (No es lo mismo sondear el controlador de
;   teclado que el buffer de teclado).
;   Si no se ha pulsado ninguna tecla, se
;   emite un mensaje de buffer vacío y
;   se sigue sondeando. Si se ha pulsado una tecla,
;   se lee el buffer de teclado con el servicio 10h
;   de la INT 16h y se presenta en pantalla el nú-
;   mero de tecla en hexadecimal y el carácter
;   ASCII. Luego se lee el estado del teclado con
;   el servicio 12h de la INT 16h y se presenta en
;   pantalla en hexadecimal.
;   Se vuelve a sondear hasta que aparezca el
;   carácter 'S'.
;
; - Utiliza la biblioteca de macros para la
;   conversión de entero byte a hexadecimal
;-------------------------------------------------

                TITLE         pr5-ac02
                DOSSEG
                .MODEL        SMALL

                INCLUDE       macros.inc

                .STACK        100h

                .DATA
dato            DB     0
wbuffer         LABEL word
buffer DB       0, 0
cadena DB       0, 0, '$'
```

```
mensaje1      DB     "Pulse el teclado ('S' termina el programa)",13,10,'$'
mensaje2      DB     "Buffer de teclado vacío",13,'$'
mensaje3      DB     "Código ASCII: ",'$'
mensaje4      DB     "   Scan code: ",'$'
mensaje5      DB     "   Estado: ",'$'
cr$           DB     " ",10,13,'$'

pcadena       DW     cadena
pmensaje1     DW     mensaje1
pmensaje2     DW     mensaje2
pmensaje3     DW     mensaje3
pmensaje4     DW     mensaje4
pmensaje5     DW     mensaje5
pcr$          DW     cr$


              .CODE
inicio:       MOV    AX, @DATA
              MOV    DS, AX

              @mostrarcadena pmensaje1

                                    ; estado del buffer de teclado
bucle:        MOV    AH, 11h        ; servicio no bloqueante teclado 107 teclas
              INT    16h

              JNZ    vertecla

              @mostrarcadena pmensaje2

              JMP    bucle
                                    ; lectura del buffer de teclado
vertecla:     MOV    AH, 10h        ; servicio no bloqueante teclado 107 teclas
              INT    16h

              MOV    dato, AH       ; salvo el scan code en dato
              MOV    BL, AL         ; salvo el ASCII en BL

              @mostrarcadena pmensaje3

              MOV    DL, BL
              MOV    AH, 02h
              INT    21h

              @mostrarcadena pmensaje4

              @entero2hex dato, wbuffer
              @digito2ASCII buffer[0], cadena[1]
              @digito2ASCII buffer[1], cadena[0]

              @mostrarcadena pcadena
              @mostrarcadena pmensaje5

              MOV    AH, 12h        ; idem teclado 107 teclas
              INT    16h
```

```
                MOV    dato, AL            ; salvo el estado code en dato

                @entero2hex dato, wbuffer
                @digito2ASCII buffer[0], cadena[1]
                @digito2ASCII buffer[1], cadena[0]

                @mostrarcadena pcadena
                @mostrarcadena pcr$

                CMP    BL, 'S'
                JNZ    saltolargo

                MOV    AX, 4C00h
                INT    21h

saltolargo:     JMP    bucle
                END    inicio
```