

Práctica 1: Sencillo Juego en Ensamblador

Periféricos e Interfaces
David Morales Sáez

Introducción

En esta práctica se nos ha solicitado hacer un juego para la máquina Intel 8086. Este juego no es más que el clásico Arkanoid, un juego cuyo fin es destruir todas las barras con una bola. El espacio de juego está acotado tanto a la izquierda y a la derecha como por arriba. Si la bola “cayese” por abajo, la vida se reduciría. Tenemos al principio 3 vidas y, si perdemos todas las vidas y volvemos a perder, la partida se acaba. La meta es destruir todas las barras sin perder todas las vidas.

Desarrollo

Para el control de la barra y la bola, utilizamos 5 variables, 1 para saber la posición de la barra y 2 para la bola. También utilizaremos otras 2 variables para conocer la dirección de ésta. El algoritmo del programa sería el siguiente:

```
MOSTRAMOS PANTALLA DE NOMBRE
OBTENEMOS EL NOMBRE Y LO GUARDAMOS
INICIALIZAMOS LOS VALORES Y PINTAMOS LAS BARRAS
1. ACTUALIZAMOS VIDAS Y SE MUESTRAN
  SI VIDAS ES -1
    GOTO 5
  LEEMOS DEL TECLADO
    SI PULSAMOS LA IZQUIERDA
      SI NO NOS PASAMOS DEL BORDE, NOS MOVEMOS
      EN CASO CONTRARIO, NO SE HACE NADA
    SI PULSAMOS LA DERECHA
      SI NO NOS PASAMOS DEL BORDE, NOS MOVEMOS
      EN CASO CONTRARIO, NO SE HACE NADA
    SI PULSAMOS 'ENTER'
      SE INICIA EL JUEGO (GOTO 2)
    SI PULSAMOS 'ESCAPE'
      SE PIERDE UNA VIDA
  GOTO 1
2. ACTUALIZAMOS PUNTOS Y SE MUESTRAN
  SI LOS PUNTOS = 36
    GOTO 6
  LEEMOS DEL TECLADO
    SI PULSAMOS LA IZQUIERDA
      SI NO NOS PASAMOS DEL BORDE, NOS MOVEMOS
      EN CASO CONTRARIO, NO SE HACE NADA
    GOTO 3
    SI PULSAMOS LA DERECHA
      SI NO NOS PASAMOS DEL BORDE, NOS MOVEMOS
      EN CASO CONTRARIO, NO SE HACE NADA
    GOTO 3
    SI PULSAMOS 'ESCAPE'
      SE PIERDE UNA VIDA
    GOTO 1
  GOTO 2
3. LEEMOS LA SIGUIENTE POSICIÓN DE LA BOLA
  SI ES UN BLOQUE
    SE ELIMINA EL BLOQUE
    SE SUMA 1 A LA PUNTUACIÓN
    GOTO 4
  SI ES EL FONDO
    GOTO 4
4. MOVEMOS LA BOLA
  GOTO 2
5. MOSTRAMOS 'GAME OVER' Y LA PUNTUACION
  GOTO 7
6. MOSTRAMOS 'HAS GANADO'
7. FIN
```

Las interrupciones que se han usado han sido las siguientes:

INT 10H

AL = 03H, AH = 0

Modo de pantalla: modo gráfico

AH = 06H

Impresión de un rectángulo en la pantalla

AH = 08H

Leemos el carácter y los atributos de la posición del cursor

AH = 09H

Escribimos un carácter con sus atributos en la posición del

cursor

AH = 0AH

Escribimos un carácter en la posición del cursor

AX = 1003H

Eliminamos el parpadeo

AH = 2

Ponemos el cursor en la posición deseada

INT 16H

AH = 00H

Leemos la tecla pulsada

INT 21H

AH = 09H

Imprimimos una cadena de caracteres

AH = 0BH

Comprobamos si se ha pulsado algo

AH = 4CH

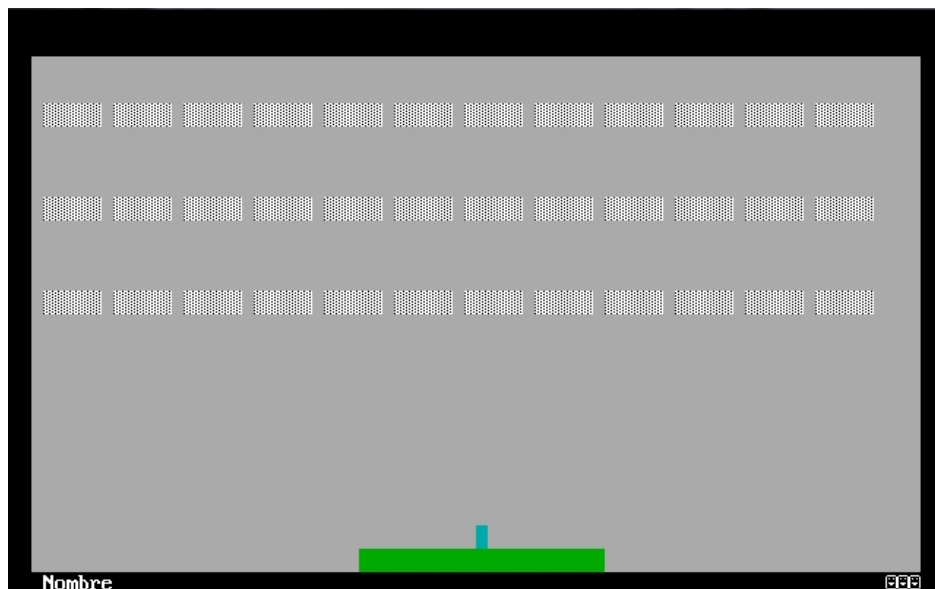
Finalizamos el programa

Descripción de los resultados

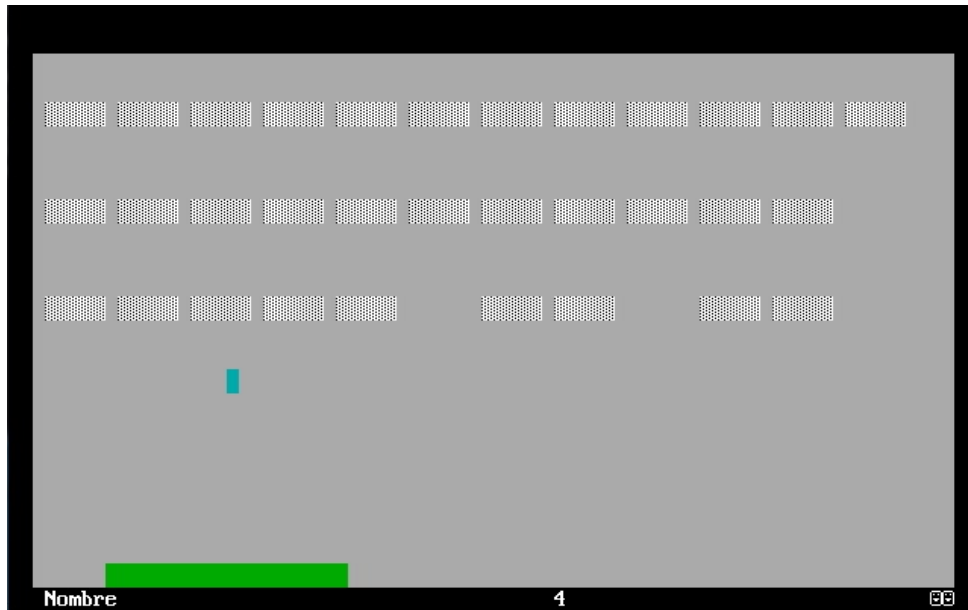
Una vez que se inicie el juego, se le solicitará al jugador un nombre, que será mostrado en la pantalla a la vez que va escribiendo. Una vez escrito, ha de pulsar 'ENTER' y se cargará la pantalla de juego.



En la pantalla de juego, podemos ver que en la esquina inferior izquierda está el nombre, en el centro está la puntuación y a la derecha están las vidas. Antes de iniciar el juego, el jugador puede colocar la barra en el lugar que desee, para poder acertar allá donde quiera. También hay que comentar que, si movemos la barra a la izquierda, la bola se dirigirá hacia arriba y a la izquierda, y si la movemos a la derecha, la bola irá hacia la derecha.



Una vez que empecemos a jugar, la bola irá moviéndose como es debido, cambiando de rumbo cada vez que choque, ya sea con una pared, con un bloque o con la barra). En el caso de perder una vida, volveríamos a la situación inicial, donde podemos escoger la dirección de la bola y la posición de la barra.



En el caso de perder, se mostrará en la pantalla un mensaje indicándonos la puntuación final. Si ganamos, se nos felicitará por la victoria.





Hay que comentar que hay un ligero retraso con el movimiento de la barra, ya que tarda más en obtener la pulsación del teclado que en mover la bola.

Conclusiones

Debido a que estamos creando un programa en ensamblador para la máquina Intel 8086, me he visto obligado a utilizar el emulador DOSBO, ya que el rendimiento del emulador proporcionado por el propio Sistema Operativo (Windows) es muy inferior debido a la política de recursos. Aparte, esta práctica me ha permitido ver bien la influencia de la organización del código respecto a su ejecución, ya que los programas hechos con anterioridad no requerían una sincronización tan ajustada para su uso como este juego (se ve claramente con el movimiento de la barra).