

19. Módulos

Nuestro videojuego cada vez tiene mejor aspecto y estamos muy cerca de poder crear un videojuego comercial como los que se crearon para las consolas de 8 y 16 bits en los 90.

La mayor complicación que tenemos ahora mismo es que nuestro código no para de crecer y cada vez nos resulta más difícil localizar una determinada línea de código, añadir una nueva función o simplemente entender el código que habíamos escrito.

El archivo de código puede llegar a tener varios miles de líneas si queremos crear un buen videojuego, y esto es impracticable. Un videojuego de tal envergadura requeriría el trabajo de varias personas, ¿Y cómo iban a trabajar varias personas en paralelo sobre un mismo archivo de código?

En este tema veremos cómo es posible separar cada uno de los elementos de nuestro videojuego en un archivo .prg independiente. Veremos las ventajas que supone a nivel de legibilidad, facilidad de modificación y aplicación e incluso reutilización en otros videojuegos. Por supuesto, también descubriremos alguna desventaja.

19.1 Concepto de módulo

Un módulo de código en Benu es un archivo .prg que ofrece una funcionalidad determinada. En un módulo podemos incluir un determinado proceso, una serie de variables GLOBAL, una declaración de TYPES, etc.

Al separar un videojuego en módulos siempre debe resultar un módulo principal. En nuestro caso nuestro módulo principal será el archivo videojuego.prg que manteníamos hasta ahora, sólo que sus procesos (Bloques PROCESS - END), su declaración de TYPES y sus variables GLOBAL ahora pasarán a formar parte de archivos .prg distintos con nombres representativos, como por ejemplo enemigo.prg, disparo.prg, global.prg y type.prg.

Una vez hecho esto sólo nos resta incluir en videojuego.prg una serie de instrucciones que indican que los nuevos archivos .prg también forman parte de nuestro videojuego. Para ello utilizaremos sentencias INCLUDE, cuya sintaxis es la siguiente:

```
INCLUDE "nombre_del_archivo.prg" ;
```

```
...
```

Por cada archivo .prg que queramos incluir deberemos añadir una sentencia INCLUDE como la anterior.

Las sentencias INCLUDE se ubican justo debajo de la sentencia PROGRAM, que era la primera línea de todo videojuego en Benu. Realiza la tarea de separar tu videojuego en módulos, no debería tomarte demasiado tiempo y sin duda te será de muchísima utilidad mantener esta organización en módulos durante el resto de trabajo que nos queda por hacer.

Si realizas correctamente la tarea anterior, el módulo principal videojuego.prg quedará de la siguiente forma:

```
PROGRAM nombre_del_videojuego;  
  
INCLUDE "modulo1.prg";  
INCLUDE "modulo2.prg";  
...  
  
BEGIN  
    set_mode (...);  
    load_fpg (...);  
    <resto de instrucciones del proceso principal>  
  
END
```

Recuerda que en el Tema 8 hablamos de la jerarquía de procesos en Benu. Ahora con los módulos queda claro cómo el proceso principal es el primer proceso que se invoca y a partir de él se van generando los hijos, nietos, etc. Más adelante veremos que para realizar proyectos complejos es más que conveniente mantener un diagrama que ilustre la relación jerárquica de los módulos/procesos implicados en nuestro videojuego.