

¿Que es Programación Estructurada?

Un programa estructurado es *esencialmente* aquel que se puede *entender fácilmente*, y la *abstracción* es la herramienta más útil para llevar a cabo dicho cometido. La *abstracción* se basa en la identificación de los elementos a una determinada escala, e ignorando los detalles especificados en escalas inferiores.

La complejidad del software es tan grande que la abstracción es la mejor herramienta de que disponemos para ser capaz de analizarlo.

Sólo podremos emplear la *abstracción* en el análisis del software si utilizamos las estructuras de control de una forma disciplinada. Por ejemplo, consideremos la figura 1, podemos dibujar cajas alrededor de los componentes como se muestra en la figura. Debido a que está construido utilizando unas estructuras de control restringidas, podemos ver el programa a todos los niveles de detalle como componentes abstractos que tienen una **única entrada y una única salida**. Si examinamos cualquier subsistema del programa, éste se encuentra totalmente contenido en otro, es decir, las cajas no se solapan unas con otras. Si miramos en detalle a sus contenidos, no surgen inesperadamente conexiones con otros subsistemas.

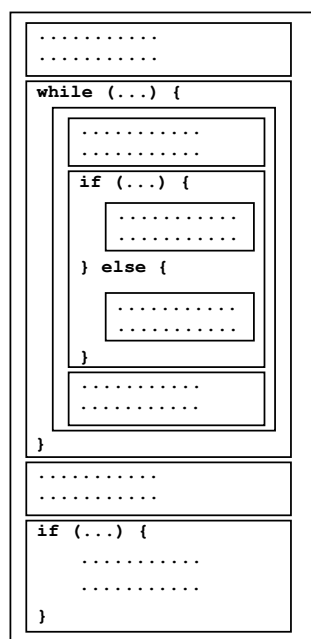


Figura 1

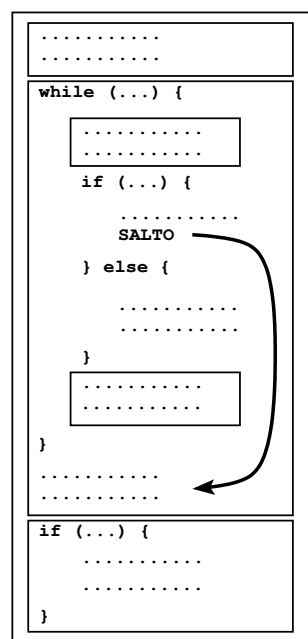


Figura 2

Supongamos que en el anterior programa introducimos una sentencia de salto (goto, exit, break, continue, return, etc) como se muestra en la figura 2. Hemos *arruinado* la estructura, ya que no podemos ver el programa a *diferentes niveles de abstracción*, y **deberemos entenderlo como un todo**. La *instrucción de salto* arruina todos los niveles de abstracción de todos los bloques que están involucrados en el salto, y que ya no tienen una única entrada y una única salida.

Como analogía, comparemos los problemas de entender un *plato de espaguetis* con un plato de *lasaña*. Para entender los *espaguetis*, los deberemos entender como un todo, es decir, no podemos emplear la abstracción. Sin embargo, con la *lasaña* podemos analizarla por capas, y cada capa a su vez volverla a analizar descubriendo los detalles interesantes de la capas sucesivas y comprendiendo cada una de ellas separada del resto.

La idea de abstracción es una herramienta poderosa a la hora de construir software que sea comprensible. Sin embargo, por sí misma es insuficiente. Para crear programas que estén bien estructurados, necesitamos además un método sistemático que nos *conduzca de la especificación del problema a la estructura del programa*. Para ello se utiliza la metodología de programación *descendente* basada en *refinamientos sucesivos*.

Así mismo, la *abstracción* aplicada a las estructuras de datos nos conduce a los *Tipos Abstractos de Datos* y a la *Programación Orientada a Objetos*.