



Fundamentos de la Programación y Métodos Numéricos

CLASE 2. **Modularización**



Modularización

Un problema complejo puede ser resuelto de manera más fácil y eficiente si se divide en problemas más pequeños.

El problema se divide en "subproblemas".

Se conocen con el nombre de *subalgoritmos*.

Los subalgoritmos o módulos, se escriben sólo una vez, luego es posible hacer referencia a ellos ("llamarlos") desde diferentes puntos.

La ventaja obvia es que nos permite **reutilización** y evita la duplicación de códigos.

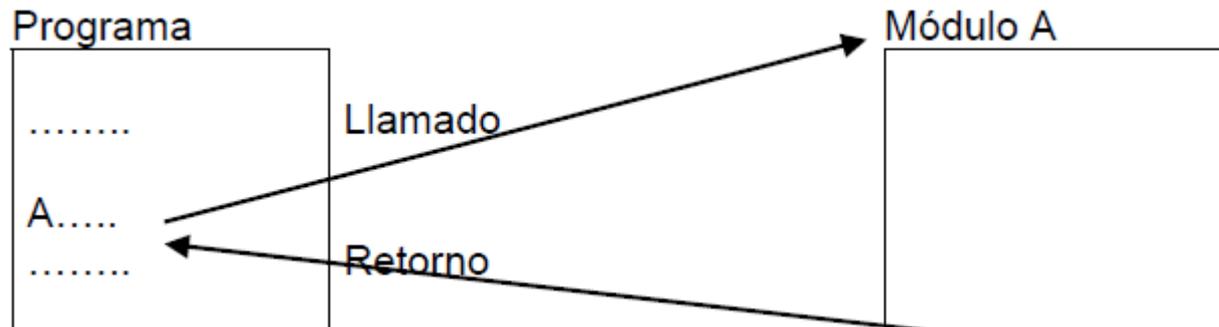
Modularización

Cada módulo debe tener un nombre que lo identifique.

Puede tener una serie de parámetros asociados.

El nombre del módulo es utilizado para la invocación del mismo.

Cuando se invoca a un modulo se transfiere el control al mismo y una vez finalizada la última instrucción del módulo el control retornará a la siguiente instrucción del programa o subprograma que lo llamó.





Modularización

En programación existen dos tipos de variables, las llamadas **locales** y las variables **globales**.

Variables Locales: Son aquellas que se encuentran dentro de un módulo.

Variables Globales: Son las que se definen o están declaradas en el algoritmo principal

Comparación: Una variable local (de un subprograma) no tiene ningún significado en el algoritmo principal y otros subprogramas. Si un subprograma asigna un valor a una de sus variables locales, este valor no es accesible en otros subprogramas, es decir, no pueden utilizar este valor. Las variables globales tienen la ventaja de compartir información de diferentes subprogramas.



Funciones y Procedimientos

Una **función** es una expresión que toma uno o más valores llamados argumentos o parámetros y produce un valor que se llama resultado. Este resultado es además, único.

La llamada a una función será siempre:

variable = nombre_funcion (lista_parámetros)