



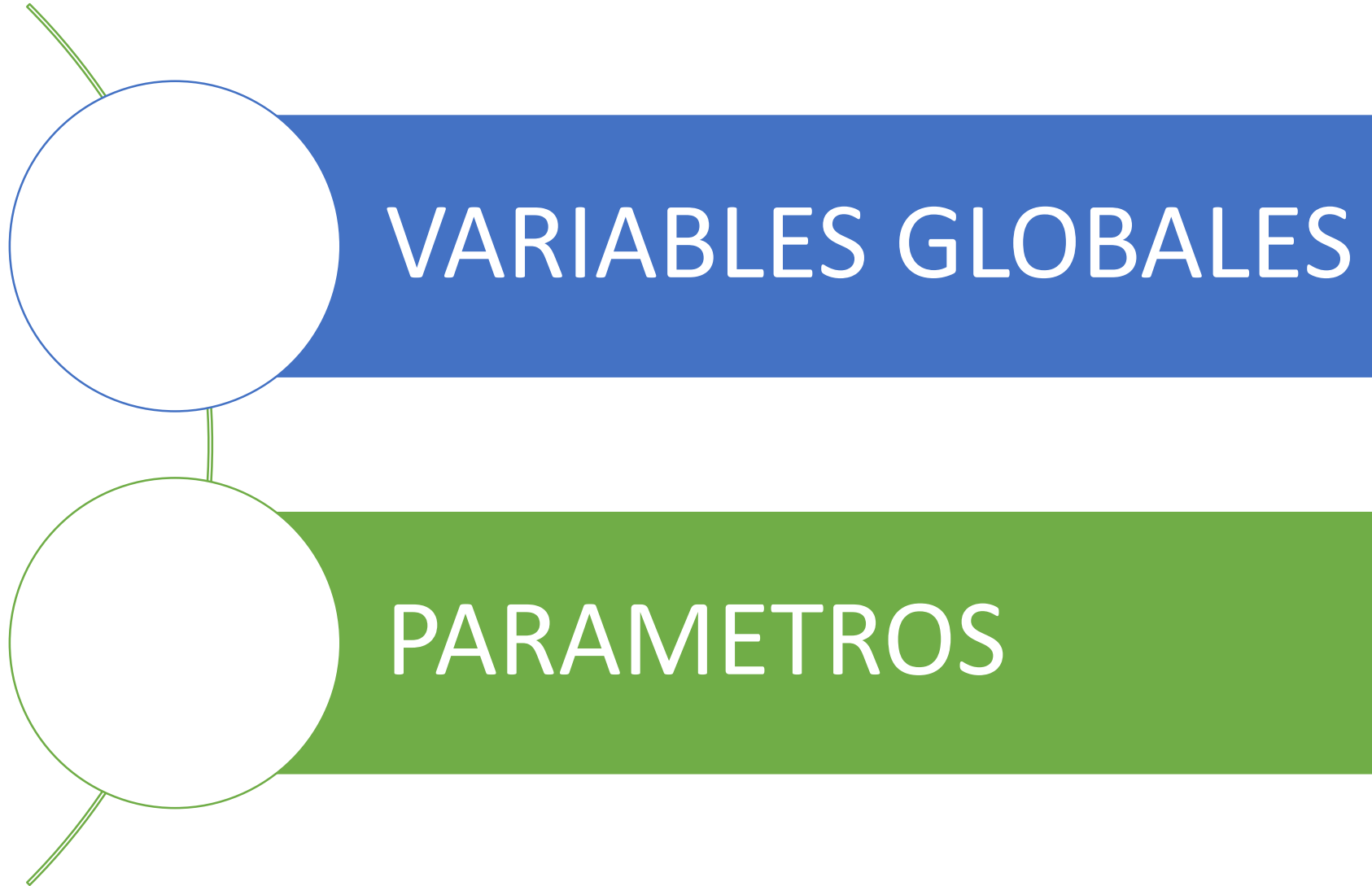
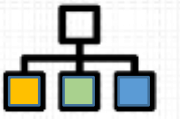
Conceptos de Algoritmos Datos y Programas

CADP – TEMAS

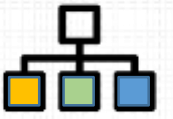


Comunicación entre módulos

CADP – COMUNICACION ENTRE MODULOS



CADP – COMUNICACION ENTRE MODULOS



VARIABLES GLOBALES

```
Program ejemplo1;  
  Var  
    x:integer;  
  
  Procedure uno;  
  Begin  
    x:= x+1;  
    write (x);  
  End;  
  Procedure dos;  
  Begin  
    x:= x MOD 10;  
    write (x);  
  End;  
  var  
    x: integer;  
  
  Begin  
    x:=9;  
    uno;  
    write (x);  
  End.
```



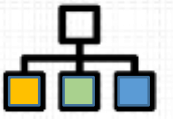
Demasiados identificadores

No se especifica la comunicación entre los módulos

Conflictos de nombres de identificadores utilizados por diferentes programadores.

Posibilidad de perder integridad de los datos, al modificar involuntariamente en un módulo datos de alguna variable que luego deberá utilizar otro módulo.

CADP – COMUNICACION ENTRE MODULOS



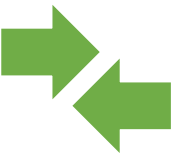
PARAMETROS

La solución a estos problemas ocasionados por el uso de variables globales es una combinación de **ocultamiento de datos (Data Hiding)** y **uso de parámetros**.

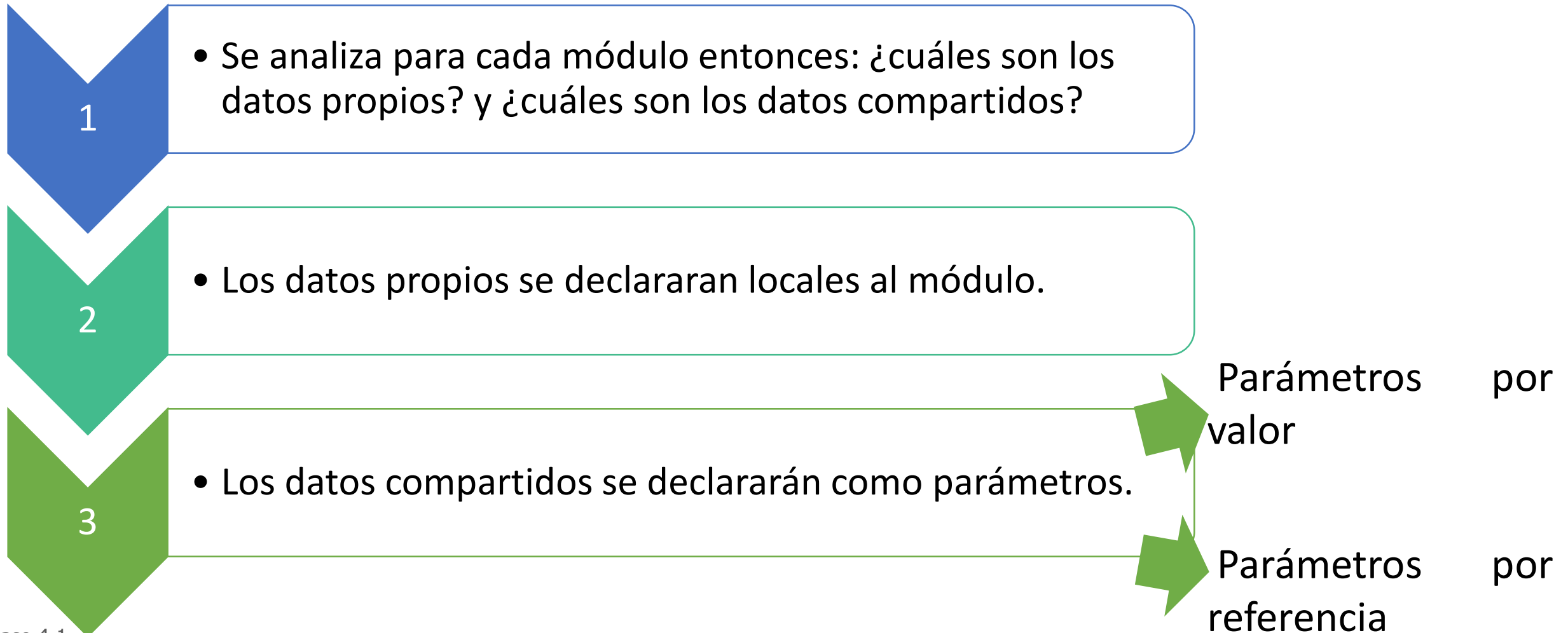
El ocultamiento de datos significa que los datos exclusivos de un módulo NO deben ser "visibles" o utilizables por los demás módulos.

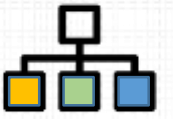
El uso de parámetros significa que los datos compartidos se deben especificar como parámetros que se transmiten entre módulos.

CADP – COMUNICACION ENTRE MODULOS



PARAMETROS – Cómo vamos a trabajar?





PARAMETROS

VALOR

- El módulo recibe un valor, puede realizar operaciones y/o cálculos, pero no producirá ningún cambio ni tampoco tendrá incidencia fuera del módulo.

REFERENCIA

- El módulo recibe una dirección, puede realizar operaciones y/o cálculos, que producirán cambios y tendrán incidencia fuera del módulo.

CADP – COMUNICACION ENTRE MODULOS



PARAMETROS

```
Program ejemplo2;  
  
  Procedure uno (PARAMETRO1; PARAMETRO2);  
    Begin  
      ...  
    End;  
  Procedure dos (PARAMETRO);  
    Begin  
      ...  
    End;  
  var  
    x,y,z: integer;  
  
  Begin  
    ...  
    uno(x,y);  
    dos(z);  
    ...  
  End.
```



Cada módulo indica que necesita recibir

Cada módulo indica que devuelve

No existe el problema donde un se pueda modificar el valor sin darse cuenta.