



Facultad de Ciencias
de la **Administración**

Licenciatura en Sistemas



Algoritmos y Programación

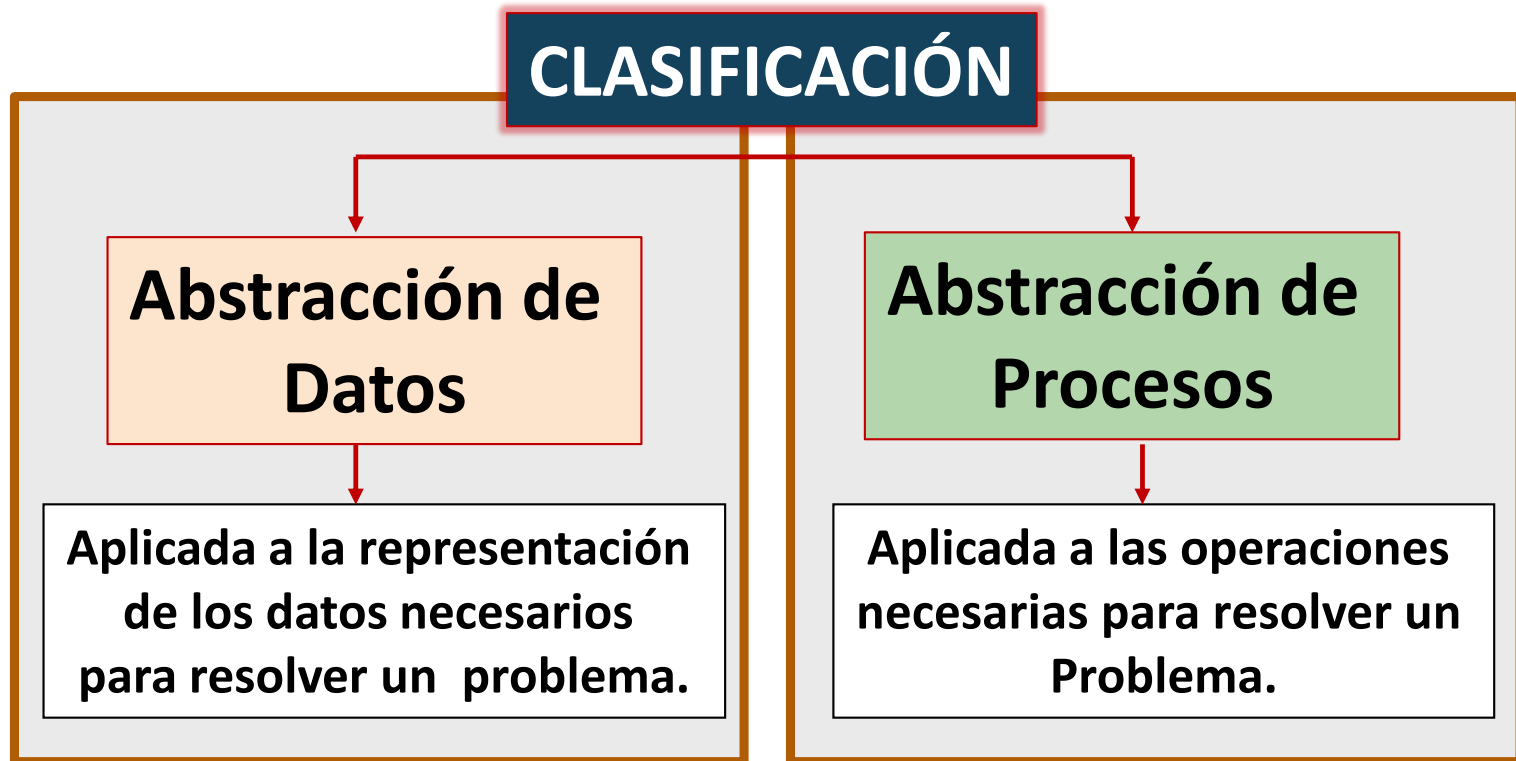
***Modularización -
Subprogramas***

Abstracción

- ❑ Mecanismo fundamental del razonamiento humano para manejar la complejidad de las cosas.
- ❑ Consiste en prestar atención solamente a los aspectos importantes de un tema dejando de lado los detalles que pueden considerarse secundarios en una aplicación particular.
- ❑ “Simplificar” para entender la realidad. Proceso de generalización al reducir el contenido de una información.

Abstracción

En Informática:



Abstracción de Procesos

Modularización

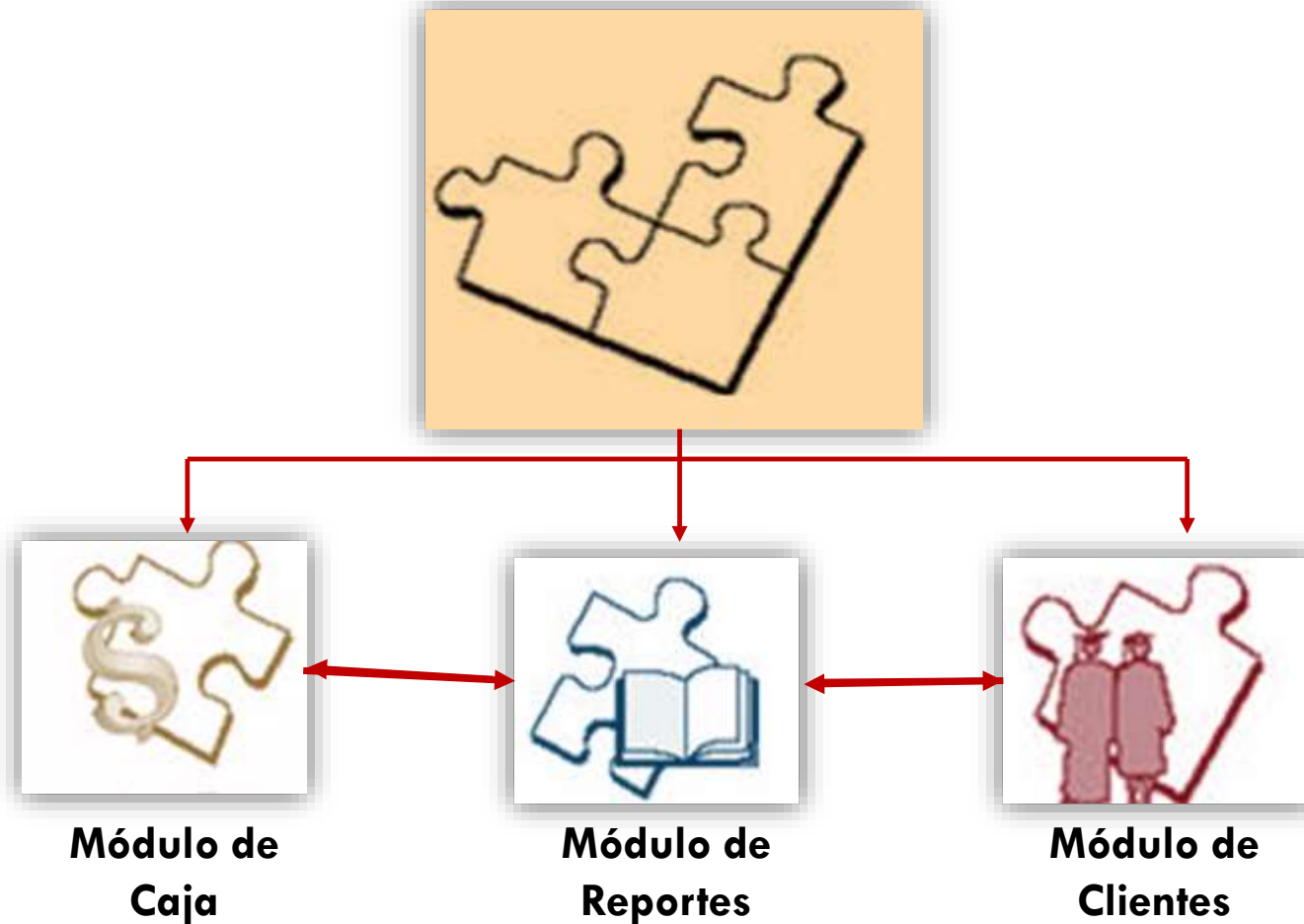
Veamos algunas maneras de definir este término:

1. Significa dividir un problema en partes funcionalmente independientes que encapsulen operaciones y datos.
2. Significa descomponer un programa en un número pequeño de abstracciones coherentes que pertenecen al dominio del problema y ocultan la complejidad interna.
3. Es dividir un programa en módulos o partes que pueden ser compilados separadamente o nó, pero existiendo conexiones o acoplamiento entre los módulos.

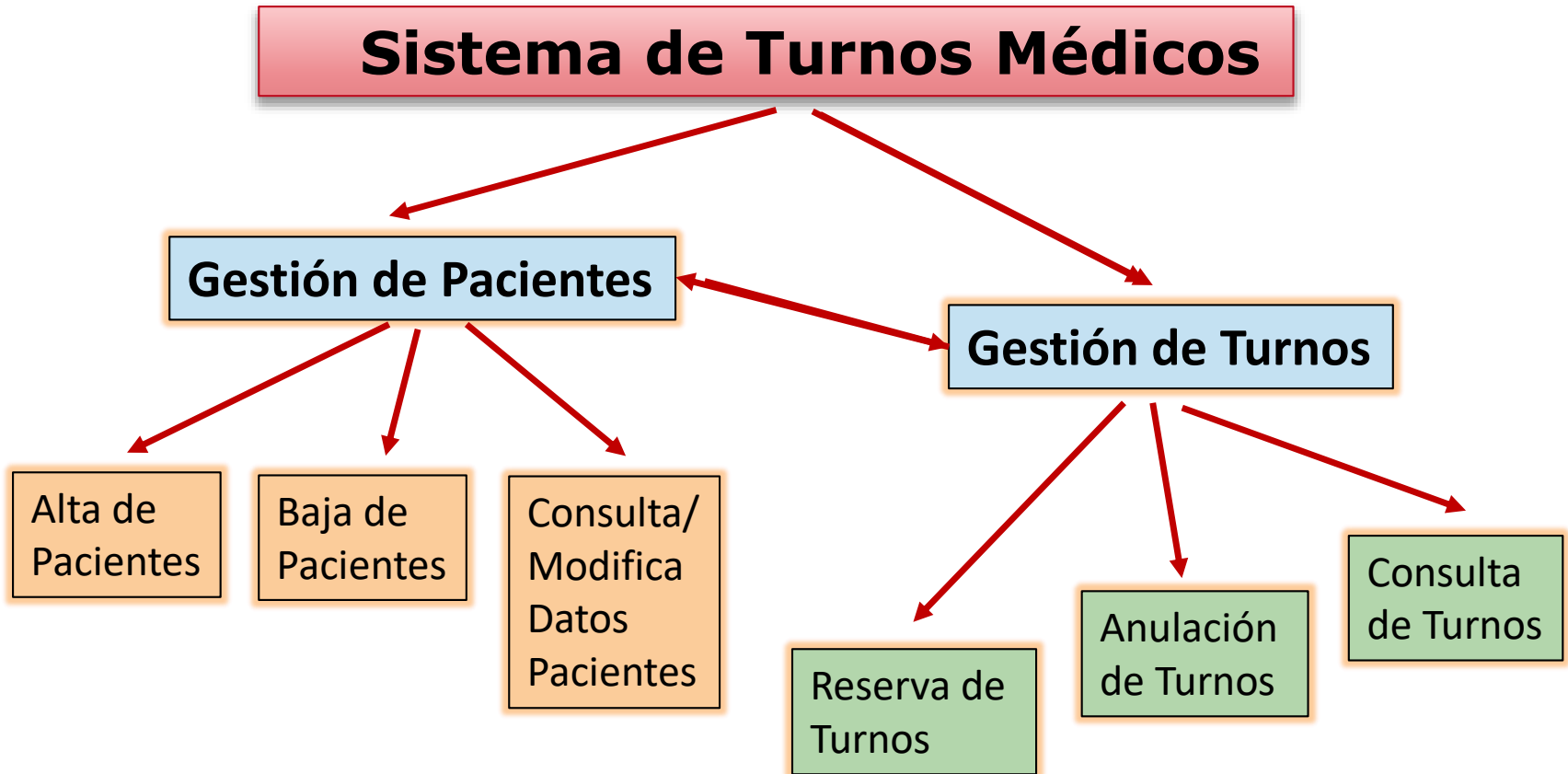


Modularización

Sistema de Facturación



Modularización



También un sistema puede visualizarse como un esquema donde cada módulo es ejecutado por un módulo “master”, según los requerimientos de proceso.

Modularización

VENTAJAS:

1. Mayor Productividad:

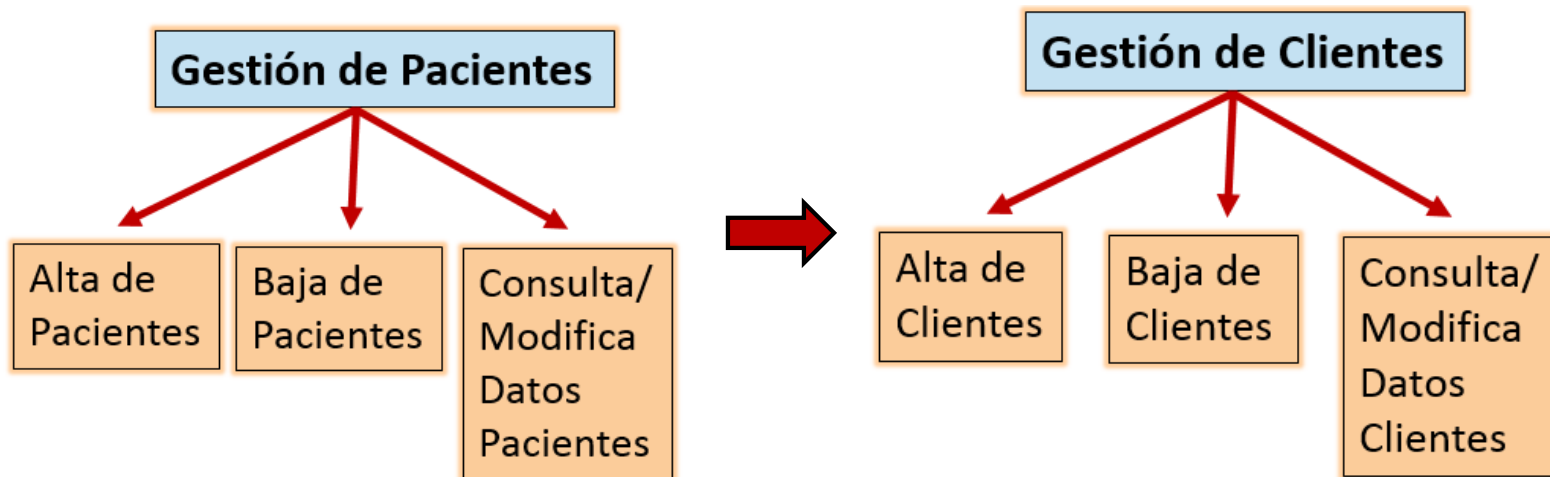
Al dividir un sistema de software en módulos funcionalmente independientes, un equipo de desarrollo puede trabajar simultáneamente en varios módulos Incrementando así la productividad (es decir reduciendo el tiempo de desarrollo global del sistema).

Modularización

VENTAJAS:

2. Reusabilidad:

Es un objetivo fundamental de la Ingeniería de software. Significa poder contar con la posibilidad de utilizar en forma repetida un producto de software ya desarrollado.



Modularización

VENTAJAS:

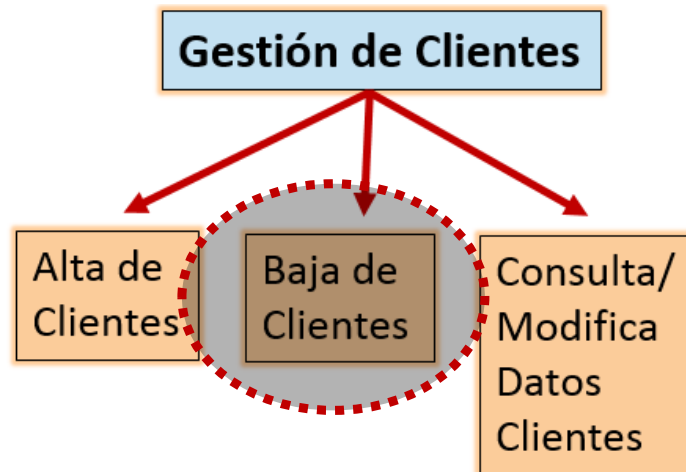
3. Facilidad de Mantenimiento:

La división del problema en módulos permite aislar más fácilmente los posibles errores que se puedan producir. Esto implica menor tiempo de corrección.



¡ No puedo dar de Baja un cliente !

**Sólo
Afecta:**



Modularización

VENTAJAS:

4. Facilidad de Crecimiento del Sistema:

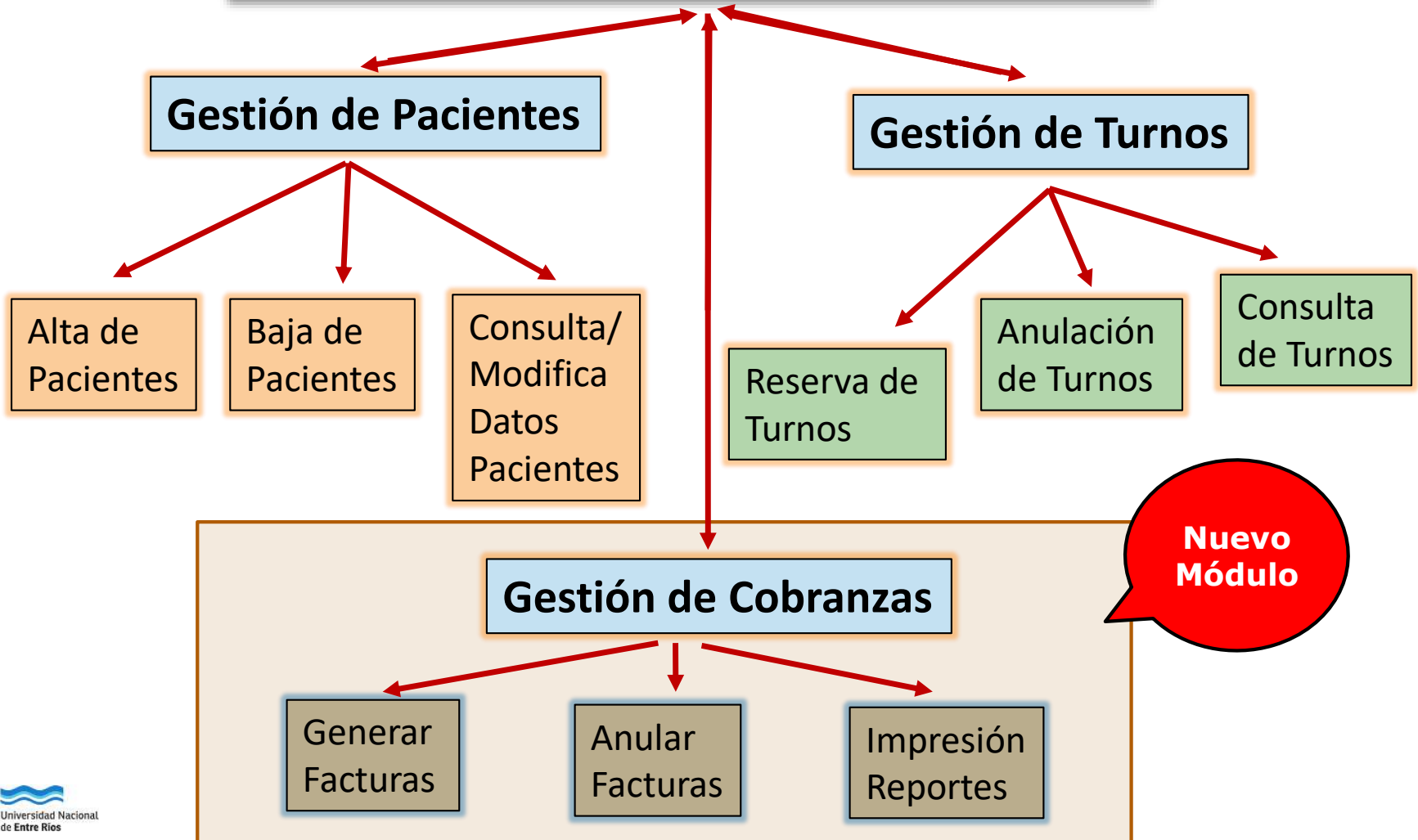
A menudo es necesario actualizar los sistemas, incorporando nuevas funcionalidades que surgen de nuevos requerimientos.

Un desarrollo modular permite disminuir los riesgos y costos que surgen del hecho de incorporar nuevas prestaciones a un Sistema en funcionamiento.



Modularización

Sistema de Turnos Médicos



Modularización

VENTAJAS - Resumen:

1. Mayor Productividad

2. Reusabilidad

3. Facilidad de Mantenimiento

4. Facilidad de Crecimiento del Sistema



**¡ Mayor
Legibilidad !**

Modularización

EN RESUMEN:

- ❑ Forma de programación que consiste en dividir un programa en módulos o subprogramas con el fin de hacerlo más legible y manejable.
- ❑ Así, un problema complejo debe ser dividido en varios subproblemas más simples, y estos a su vez en otros subproblemas más simples, hasta obtener subproblemas lo suficientemente simples como para poder ser resueltos fácilmente con algún lenguaje de programación.
- ❑ Ésta técnica se llama refinamiento sucesivo, divide y vencerás ó análisis descendente (Top-Down).

Modularización

La modularización se puede realizar:

- a) Dividiendo un problema en partes separadas (subprogramas) que se compilan por separados conformando cada uno de ellos una unidad.

- b) Dividiendo un programa en partes mas pequeñas (**procedimientos**) que forman parte del mismo programa.



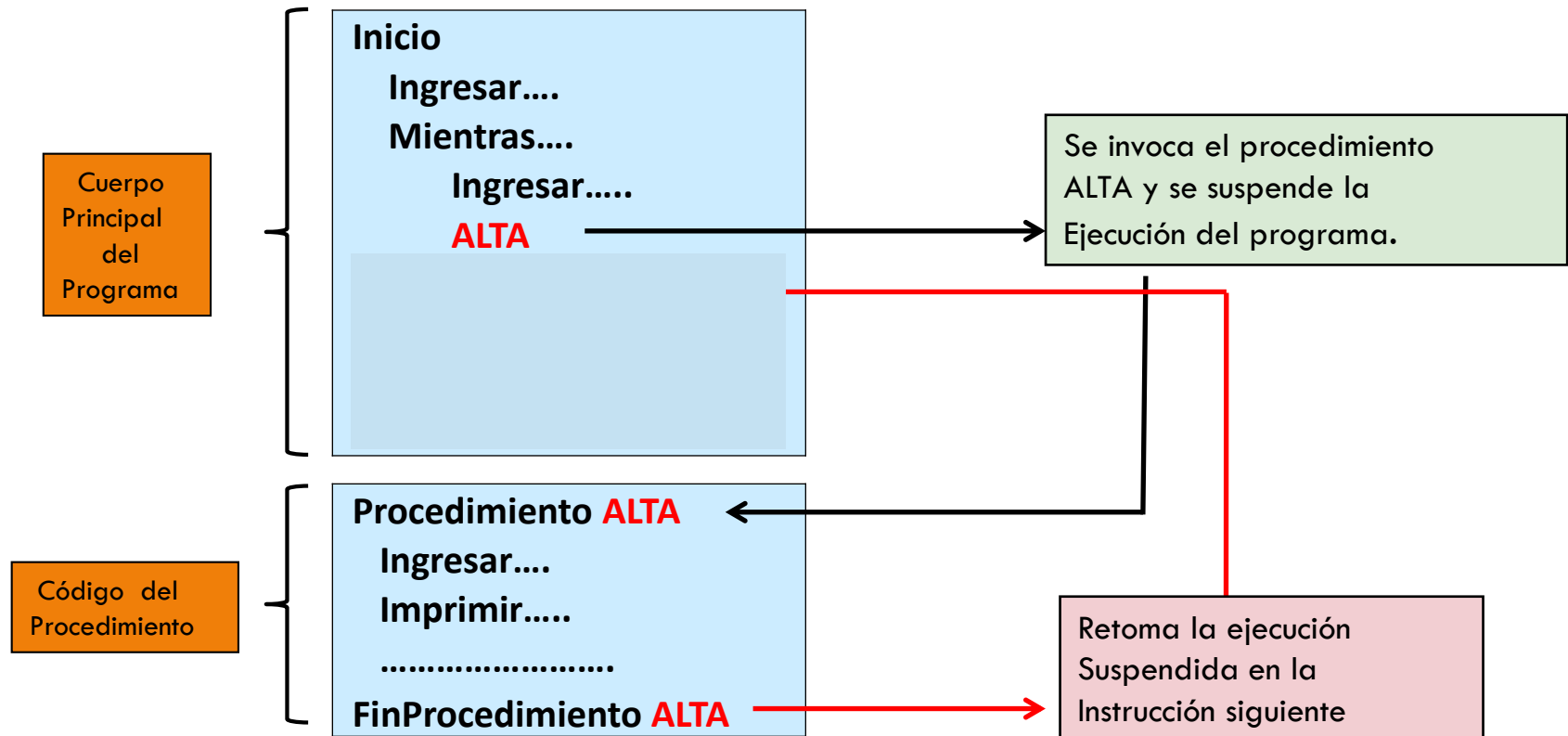
Modularización

Características generales:

- El procedimiento tiene que ser invocado.
- Cada procedimiento tiene un solo punto de entrada.
- La ejecución de la unidad que llama a un procedimiento es suspendida durante la ejecución del procedimiento.
- El control siempre retorna a la instrucción siguiente a la invocación cuando el procedimiento finaliza.



Subprogramas



Subprogramas

- Son construcciones de bloques (partes) de programas y constituyen uno de los conceptos mas importantes en el diseño de los lenguajes de programación.
 - Centrarse en el “que” para posteriormente ver el “como”.

Categorías:

1.Procedimientos.

Conjunto de sentencias algorítmicas que se ejecutan y son convocadas por un simple llamado o sentencia “call”



2. Funciones.

Esta tema lo veremos mas adelante

Subprogramas

EJEMPLO:

Se requiere procesar los datos de las inasistencias del personal de una empresa durante un semestre a fin de proceder a la liquidación de una bonificación. La empresa tiene 80 empleados, cuyos números de legajos están codificados numéricamente en forma correlativa a partir del número 1.

Se requiere codificar un programa que contemple:

- a) Ingresar los nombres de los 80 empleados
- b) Procesar las asistencias en que incurrieron los empleados, para lo cual se ingresa por teclado:
 - Número de legajo (rango de 1 a 80)
 - Mes en que se verificó la/s inasistencia/s
 - Cantidad de inasistencia del mes
- c) Imprimir un listado que contenga:

Nombre Empleado – Cantidad de Inasistencias en el semestre

Subprogramas

¿ Cómo lo resolvíamos ?



Program Busco

Type

PERSO = Record

NOM : String;

INAS : Integer

end;

Var

EMP : array [1..80] of PERSO;

i, NL,CAIN,MES : Integer;

Inicio

Mostrar "Ingreso Nombres Empleados"

Ingresar NL

Mientras NL <> 0

Ingresar EMP(NL).NOM

EMP(NL).INAS := 0

Ingresar NL

Finmientras

Mostrar "Proceso de Inasistencias"

Ingresar NL

Mientras NL <> 0

Ingresar CAIN, MES

EMP(NL).INAS := EMP(NL).INAS + CAIN

Ingresar NL

Finmientras

Mostrar "Impresión de Listado"

Para i = 1, 80, 1

Imprimir EMP(i).NOM, EMP(i).INAS

FinPara

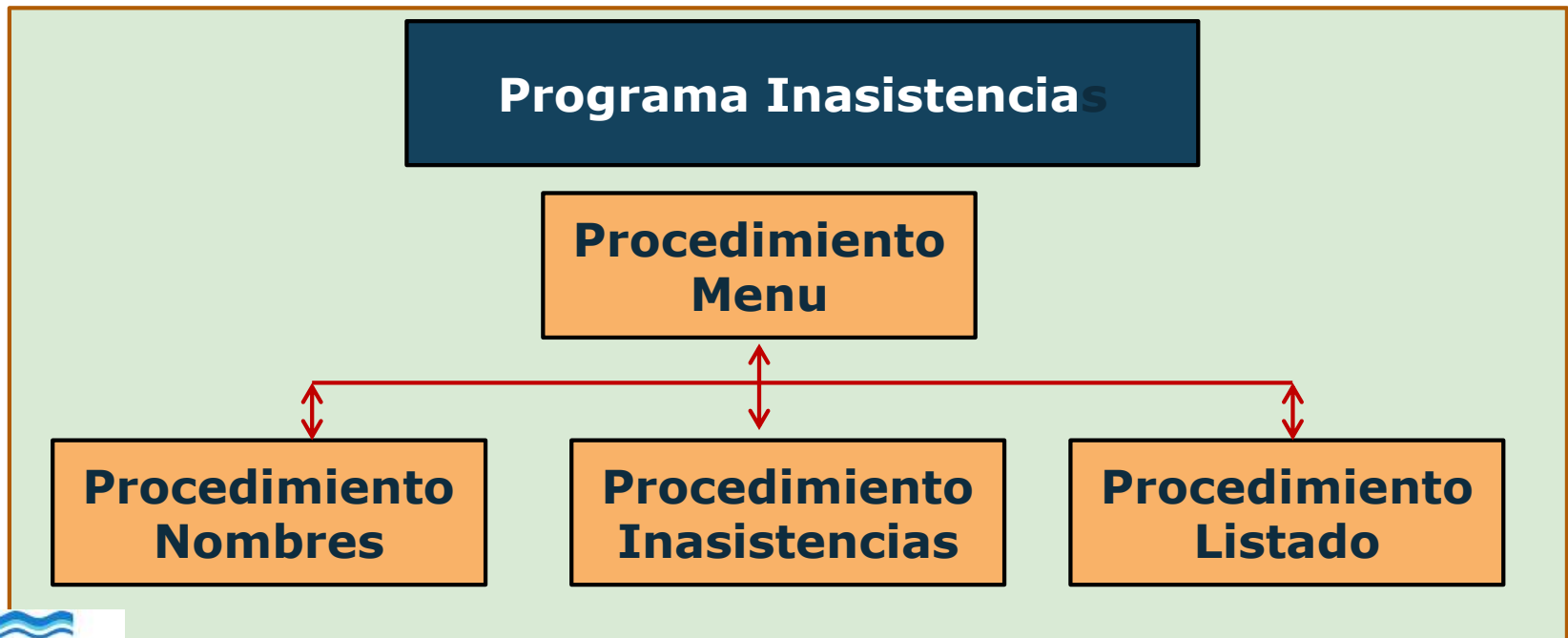
Fin



Subprogramas

ESTRATEGIA MODULAR:

- Separar el código en procedimientos , uno para cada requerimiento.
- Codificar un procedimiento (tipo Menú de opciones) que permita acceder a cada uno de los procedimientos anteriores, a petición del operador.
- Posibilitar que cada módulo se pueda ejecutar varias



Módulo Principal

Inicio

Para i = 1, 80, 1

EMP(i).INAS = 0

Finpara

MENU

Mientras OPCION <> 0

Según OPCION

1 : NOMBRE

2 : INASISTENCIAS

3 : LISTADO

Otro Valor:

Mostrar: "Código Incorrecto"

Finsegún

MENU

Finmientras

Fin

Procedimiento MENU

Mostrar "Seleccione alguna Opción"

" 1 = Ingresar Nombres "

" 2 = Procesar Inasistencias "

" 3 = Imprimir listado "

" 0 = Fin de Programa "

Ingresar OPCION

FinProcedimiento MENU

Subprogramas

Procedimiento NOMBRE

Mostrar "Ingrese Nombres Empleados, FIN=0"

Ingresar NL

Mientras NL <> 0

Ingresar EMP(NL).NOM

Ingresar NL

Finmientras

FinProcedimiento NOMBRE

Procedimiento INASISTENCIAS

Mostrar "Proceso de Inasistencias, FIN=0"

Ingresar NL

Mientras NL <> 0

Ingresar CAIN

EMP(NL).INAS := EMP(NL).INAS + CAIN

Ingresar NL

Finmientras

Fin-Procedimiento INASISTENCIAS

Procedimiento LISTADO

Mostrar "Impresión de Listado"

Para i = 1, 80, 1

Imprimir EMP(i).NOM, EMP(i).INAS

FinPara

Fin-Procedimiento LISTADO

Subprogramas

Características deseables de los módulos:



COHESIÓN

Que cada procedimiento del sistema se refiera a un único proceso o entidad. Se logra alta cohesión cuando cada procedimiento realiza una única tarea.

A **mayor** cohesión, mejor: el procedimiento será más sencillo de diseñar, programar, probar y mantener.



ACOPLAMIENTO

Mide el grado de relacionamiento de un procedimiento con los demás.

En el diseño estructurado, se logra bajo acoplamiento reduciendo las interacciones entre procedimientos.

A **menor** acoplamiento, mejor: el procedimiento en cuestión será más sencillo de diseñar, programar, probar y mantener.

Subprogramas

Definición de Objetos de Datos:

¡ Es posible especificar declaraciones en un procedimiento !



Ejemplo:

```
Program GRANDE
Var
  N1, N2 : Integer;
Begin
  Calculo
Fin
```

Variables Globales

Visibles en todo el programa y todos los procedimientos.

```
Procedure Calculo
Var
  Res : Integer;
Begin
  Ingresar N1, N2
  RES := N1 + N2
  Imprimir RES
Fin-Procedure Calculo
```

Variables Locales

Visibles solamente en el procedimiento donde está definida.



Subprogramas

Luego:

```
Program GRANDE  
  Var  
    N1, N2 : Integer;  
  Begin  
    Calculo  
    RES := RES + 1 ←  
  Fin
```

```
Procedure Calculo  
  Var  
    Res : Integer;  
  Begin  
    Ingresar N1, N2  
    RES := N1 + N2  
    Imprimir RES  
Fin-Procedure Calculo
```

¡ ERROR !
Variable no reconocida



Entidades: Variables (Conceptos)

Concepto:

Es una abstracción de un “pull” de celdas de memoria.

Es un objeto cuyo valor almacenado cambia durante la ejecución del programa.

Atributos:

Características que las define e identifica.

Nombre.

Alcance y Visibilidad.

Tipo.

Tiempo de vida.

Valor.

Locación.

Alcance y Visibilidad

Es la porción del programa (rango de sentencia) dentro de la cual una entidad es visible.

Subprogramas

En Resumen:

Las variables se pueden clasificar en:

- **Variables locales.**
 - Se declaran dentro del módulo (procedimiento o función).
 - Se utilizan (son visibles) solamente dentro del módulo.
 - Están vivas solamente durante la ejecución del módulo.
 - Se crean en el momento en que el módulo es invocado.
- **Variables globales:**
 - Se declaran fuera de los módulos/procedimientos.
 - Pueden ser usadas (son visibles) por el cuerpo principal y por cualquier módulo.



Subprogramas

EJERCICIO



Subprogramas – Ejercicio Resuelto

Una mutual de empleados debe procesar los datos de sus asociados para lo cual, el analista ha definido la codificación de los siguientes procesos:

ALTA

Se deben ingresar los siguientes datos de todos los asociados:

- Número de Asociado (ordenados en forma correlativa a partir del número 1 al 255)
- Nombre del Titular
- Saldo que debe por compras realizadas en distintos comercios.

PROCESO

Ingresar las operaciones que se realizaron con fecha posterior al saldo registrado en el punto anterior y actualizar el saldo que debe.

- Número de Asociado
- Código de operación [1=Pago, 2=Compra]
- Importe

CONSULTA

Modulo de consulta. Se ingresa el número de asociado y se deben mostrar el número de asociado, el apellido y nombre y el saldo que debe.

ESTADISTICO

d) Mostrar por pantalla un cuadro estadístico con los siguientes datos:

- Importe total adeudado por los Asociados
- Nombre del Asociado que mas importe adeuda

Se debe codificar un módulo por cada proceso, los que deben ser invocados por un menú de opciones.

Subprogramas - Ejercicio

Program Mutual

Type

Asociados = Record

NOM : String (40);

SAL : Real

end;

Var

Socio : array [1..255] of Asociados;

NUM, COC, COP, OPCION, i : Integer;

IMP, : Real;

Inicio

MENU

Mientras OPCION <> 0

Según OPCION

1 : ALTA

2 : PROCESO

3 : CONSULTA

4 : ESTADISTICO

OTHER : Mostrar "ErrorCodigo"

FinSegún

MENU

FinMientras

FIN

Procedure MENU

Mostrar "Digite Código Operación"

" 1 = Ingresar Datos de Socios"

" 2 = Procesa Operaciones"

" 3 = Consulta"

" 4 = Imprime Estadístico"

" 0 = Fin de Tarea"

Ingresar OPCION

Fin-Procedure MENU

Procedure ALTA

Ingresar NUM

Mientras NUM <> 0

Si NUM < 0 or > 255

Mostrar "Error Nro.Asociado"

Sino

Ingresar Socio[NUM].NOM

Ingresar Socio[NUM].SAL

FinSi

Ingresar NUM

FinMientras

Fin-Procedure ALTAS

Subprogramas – Ejercicio Resuelto

Procedure PROCESO

Ingresar NUM

Mientras NUM \neq 0

Si NUM < 0 or > 255

Mostrar "Error Nro.Asociado"

Sino

Ingresar COP, IMP

Si COP = 1

Socio[NUM].SAL := Socio[NUM].SAL – IMP

Sino

Socio[NUM].SAL := Socio[NUM].SAL + IMP

FinSi

FinSi

Ingresar NUM

FinMientras

Fin-Procedure PROCESO

Procedure CONSULTA

Ingresar NUM

Mientras NUM \neq 0

Si NUM < 0 or > 255

Mostrar "Error Nro.Asociado"

Sino

Mostrar NUM

Mostrar Socio[NUM].NOM

Mostrar Socio[NUM].SAL

FinSi

Ingresar NUM

FinMientras

Fin-Procedure CONSULTA

Subprogramas - Ejercicio

Procedure ESTADISTICO

Var

ACU, MAX : real;

NumMax : integer;

Inicio

ACU := 0

MAX := Min-Valor

Para i = 1, 255, 1

ACU := ACU + Socio[i].SAL

Si Socio[i].SAL > MAX

MAX := Socio[i].SAL

NumMax := i

FinSi

FinPara

Imprimir " Total Saldos : ", ACU

Imprimir "Socio > Saldo: " Socio[NumMax].NOM

Fin-Procedure ESTADISTICO

Subprogramas



Facultad de Ciencias
UNER de la **Administración**



Algoritmos y Programación

FIN



Universidad Nacional
de Entre Ríos