



UNIVERSIDAD  
DE LOS ANDES  
MERIDA VENEZUELA

# *Estructuras de Repetición*

## *Repita para*

Prof. Judith Barrios Albornoz

Departamento de Computación

Escuela de Ingeniería de Sistemas

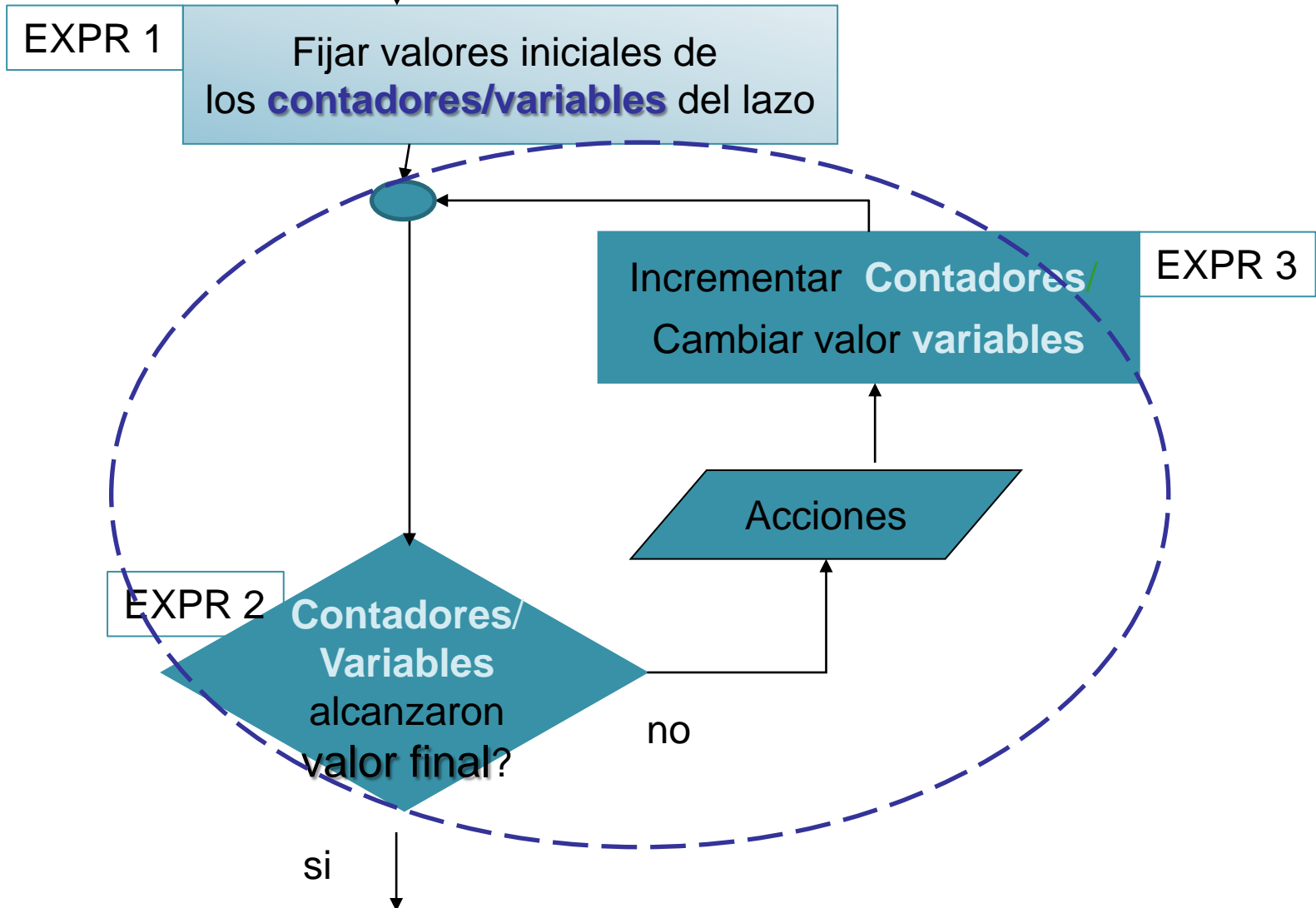
Facultad de Ingeniería

Universidad de Los Andes

Semestre A\_2013

Se utiliza cuando se conoce de antemano el  
número de veces que se va a ejecutar  
(repetir) el lazo

# Repita para



# Repita para

- Se utiliza cuando se conoce de antemano el número de veces que se va a ejecutar (**repetir**) el lazo
- La estructura **REPITA PARA** requiere de tres elementos importantes:
  - Al menos un **contador o condición inicial** que debe ser **cierta** para **entrar** al lazo (**exp1**)
  - El incremento, decremento o **expresión de cambio** con la que cada vez que se termine una repetición, se evalúa la **condición de salida** (**expr3**)
  - La condición que compruebe la **existencia del valor final** del contador del lazo (**expr2**)

# Repita para

*Pseudocódigo en español*

**Repita para** (expr1, expr2, expr3)

$S_1$

....

$S_n$

**frp**

*Código en C/C++*

**for** (expr1; expr2; expr3)

{

$S_1$

....

$S_n$

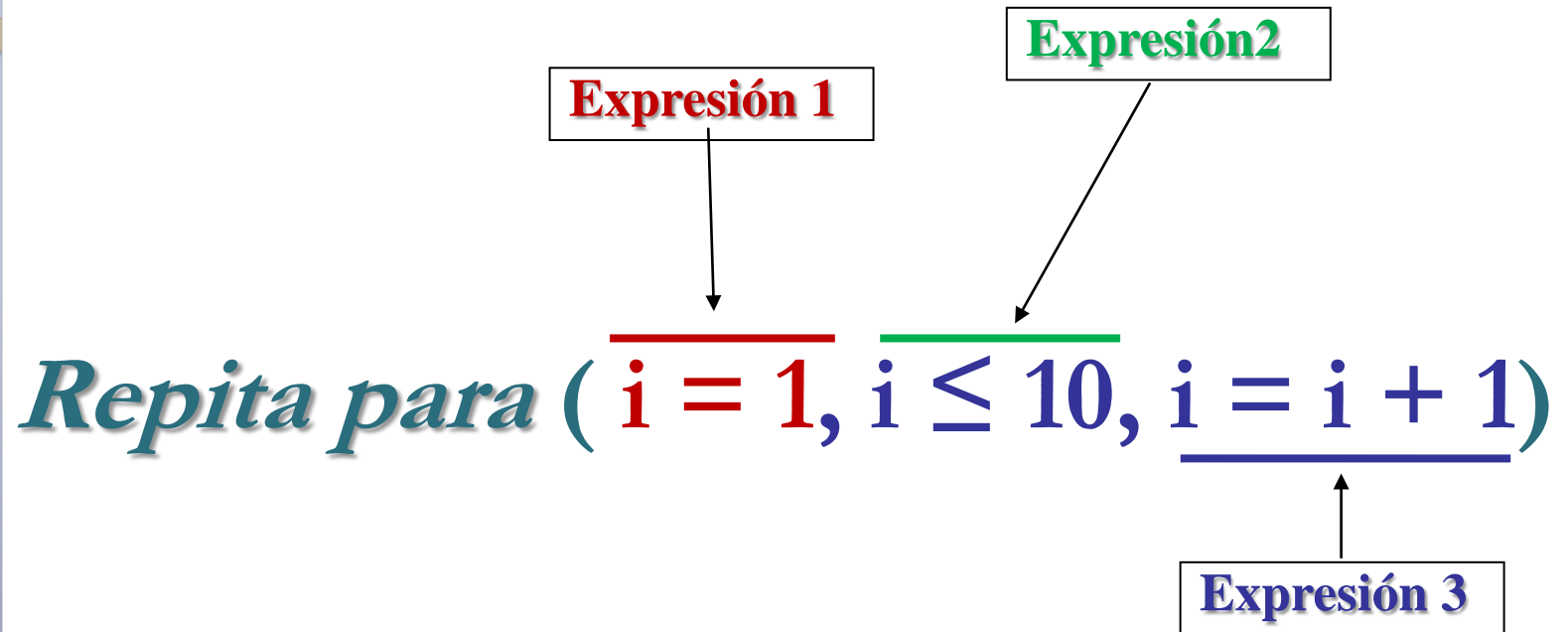
}

# Repita para

- **Repita para (*expr1*, *expr2*, *expr3*)**
  - ***expr1***: **Iniciar el contador**
    - Es evaluada una sola vez, cuando se ejecuta por primera vez el lazo. Normalmente es una sentencia de asignación
  - ***expr2***: **Condición** de **terminación del lazo**
    - Es evaluada cada vez que se ejecuta el repita para
  - ***expr3***: **Modificación del contador**
    - Se ejecuta luego de la última instrucción del lazo

**Nota: Cada una de las tres expresiones es opcional en C/C++**

# Repita para



# Repita para

**Nombre de  
la variable  
de control**

Valor final  
de la  
variable de  
control

Repita para (**i = 1**,  $i \leq 10$ , **i = i + 1**)

Valor inicial  
de la  
variable de  
control

Incremento  
de la  
variable de  
control

# Ejemplo 1

| Pseudocódigo en Español   | Código en C/C++  |
|---|--|
| <b>Repita para</b> (I=1, I $\leq$ 10, I=I+1)<br>Escribir (I)<br><b>frp</b>                      | <b>for</b> (int I=1; I <= 10; I++)<br>printf ("%d\n", I);<br>/*I es la variable de control*/                             |
| <b>Repita para</b> (I=0, I<= n-1, I=I+1)<br>Escribir ("Numero?")<br>Leer (numero)<br><b>frp</b> | <b>for</b> (I=0; I <= n-1; I++)<br>{ /* I variable de control */<br>printf("numero? %d\n");<br>scanf (%f, &numero);<br>} |



# Ejemplos modos de uso de la estructura **For** en C/C++

- `for ( ; ; );`  
*/\* Lazo infinito de una sola línea\*/*
- `for ( ; ; ) { }`  
*/\* Lazo infinito de más de una línea \*/*
- `for (long i = 0, j = 100; i < j; i++, j--)`  
  `{ printf ("%d%d\n", i, j); }`  
*/\* Variación simultanea de dos contadores \*/*
- `for (unsigned conta= 1; conta <= 10; conta++)`  
  */\* Variar el contador (conta) de 1 a 10 \*/*  
  */\* en incrementos de 1 \*/*
- `for (i = 100; i >= 1; i--)`  
*/\* Variar contador (i) de 100 a 1 en decrementos de 1 \*/*

# Ejemplos modos de uso de la estructura **For** en **C/C++**

- **for (I = 7; I <= 77; I +=7)**  
*/\* Variar el contador (I) de 7 a \*/*  
*/\* 77 en incrementos de 7 \*/*
- **for (I = 20; I >= 2; I -=2)**  
*/\* Variar el contador (I) de 20 a 2 \*/*  
*/\* en decrementos de -2 \*/*
- **for (j = 2; j <= 20; j +=3)**  
*/\* Variar el contador (j) a lo largo \*/*  
*/\* de la siguiente secuencia de valores:\*/*  
*/\* 2, 5, 8, 11, 14, 17, 20 \*/*
- **for (j = 99; j >= 0; j -=11)**  
*/\* Variar el contador (j) a lo largo de la \*/*  
*/\* siguiente secuencia de valores: 99, 88, \*/*  
*/\* 77, 66, 55, 44, 33, 22, 11, 0 \*/*

# Ejercicio 1

Sumar todos los enteros pares desde **2**  
hasta **100**

## Análisis E-P-S

Entrada: No se requiere

Proceso: Para cada número par entre el **2** y el  
**100**

- Acumular su valor en el acumulador de pares  
(**AP**)

Salida: Suma de los enteros pares (**AP**  $\in \mathbb{N}$ )

# Diseño - algoritmo

| sumaPares2-100 |   |  |
|----------------|---|--|
| {pre:}         |   | {pos: $ap \geq 0$ }  |
| 1              | $ap = 0$  | <ul style="list-style-type: none"> <li>● <b>num:</b> Natural. Contador de números pares.</li> <li>● <b>ap:</b> Natural. Suma de números pares entre 2 y 100</li> </ul> |
| 2              | <b>repita para</b> ( $num=2, num \leq 100,$<br>$num=num+2$ )<br>$ap = ap + num$<br><b>frp</b> |  |
| 3              | Escribir “Suma de enteros pares entre 2 y 100= “, $ap$  |  |
| 1              | $\rightarrow ap = 4116$   | Caso exitoso   |

# Codificación en C

```
#include <stdio.h>

void main()
{
    unsigned int ap = 0, num;
    for (num = 2; num <= 100; num += 2)
        ap += num;
    printf("Suma de enteros pares entre 2 y 100=%d\n", ap);
}
```

## Ejercicio 2

**Dadas las notas de  $n$  estudiantes de las materias de PRI, Ingeniería de Sistemas (IS) y Cálculo I 0 (CI0). Calcular la media de cada estudiante**

### Análisis E-P-S

**Entrada:** Número de estudiantes ( $ne \in \mathbb{N}$ ), nota de las tres materias para cada estudiante ( $notaPRI \in \mathbb{N}$ ,  $notaIS \in \mathbb{N}$ ,  $notaCI0 \in \mathbb{N}$ ) y  
 $notaPRI, notaIS$  y  $notaCI0 \in [0, 20]$

**Proceso:** Para cada estudiante hacer:

Calcular media = 
$$\frac{notaPRI + notaIS + notaCI0}{3}$$

**Salida:** La media para cada estudiante ( $media \in \mathbb{R}$ )

# Diseño - algoritmo

| media3materias |  |   |
|----------------|--|---|
| {pre:}         |  | {pos: ne $\geq$ 0 }   |
| 1<br>2<br>3    | <p>Escribir “Introduzca el número total de estudiantes que debe ser mayor que 0”</p> <p>Leer <b>ne</b></p> <p><b>repita para</b> (<math>k=1, k \leq ne, k=k+1</math>)</p> <p>    Escribir “Para el estudiante “, k, “Introduzca la nota de Programación 1 entre 0 y 20”</p> <p>    Leer notaPR1</p> <p>    Escribir “Introduzca la nota de Ing. de Sistemas entre 0 y 20”</p> <p>    Leer notaIS</p> <p>    Escribir “Introduzca la nota de Cálculo 10 entre 0 y 20”</p> <p>    Leer notaC10</p> <p>    <math>media = (notaPR1 + notaIS + notaC10) / 3</math></p> <p>    Escribir “Media del estudiante “, k, “ = “, media</p> <p><b>frp</b></p> | <p>● <b>k:</b> Natural. <b>Contador</b> del número de estudiantes.</p> <p>● <b>ne:</b> Natural. Número total de estudiantes <math>&gt; 0</math></p> <p>● <b>notaPR1, notaIS, notaC10:</b> Natural. Notas de cada estudiante en las materias Programación 1 (PR1), Ing. de Sistemas y Cálculo 10.</p> <p>● <b>media:</b> Real. Promedio de las tres notas para cada estudiante</p> |
| 1<br>2         | <p>ne = 0 -&gt;</p> <p>ne = 1, notaPR1 = 10, notaPro = 11, notaC40 = 12 -&gt; media = 11.0</p>   | <p>Caso exitoso</p> <p>Caso exitoso</p>   |

# Codificación en C

```
/* Calcula el promedio de 3 notas para n estudiantes */

#include <stdio.h>
void main ()
{ unsigned integer ne, k, notaPR1, notaIS, notaC10;
  float media;
  printf ("Introduzca el numero total de estudiantes mayor que 0 \n");
  scanf ("%d",&ne);
  for (k = 1; k <= ne; k++)
  { printf("Para el estudiante %d\n", k);
    printf("Introduzca la nota de programacion entre 0 y 20\n");
    scanf("%d",&notaPR1);
    printf("Introduzca nota Ingenieria de Sistemas entre 0 y 20 \n");
    scanf ("%d",&notaIS);
    printf ("Introduzca la nota de Calculo 10 entre 0 y 20 \n");
    scanf ("%d",&notaC10);
    media =((float)notaPR1+(float)notaIS +(float)notaC10)/3.0;
    printf("Media del estudiante %d es %f\n", k, media);
  }
}
```



# Variación solución anterior

- Rehacer el algoritmo y la codificación para que ahora valide los datos de entrada tanto del número de estudiantes como de las notas de cada materia.
- Solo debe aceptar valores válidos por lo que debe solicitarlos hasta que sean válidos

# Ejercicio 3

Calcular de manera separada – independiente - la suma de los números pares e impares comprendidos entre 1 y n

## Análisis E-P-S

Entrada: Valor de  $n \in \mathbb{N}$ .

Proceso: Para todos los números comprendidos entre 1 y n:

Si **número** es par  $sp = sp + \text{número}$

Si **número** es impar  $si = si + \text{número}$

Salida: Suma de los números pares ( $sp \in \mathbb{N}$ ) y suma de los números impares ( $si \in \mathbb{N}$ )

# Diseño - algoritmo

| sumaParesImpares           |  |  |
|----------------------------|--|--|
| {pre: $n \in \mathbb{N}$ } |  | {pos: $n, sp, si \geq 0$ }   |
| 1                          | <b>hacer</b><br>Escribir “Introduzca el valor de $n \geq 1$ ”<br>Leer $n$<br><b>mientras</b> ( $n < 1$ )<br>2 $sp = 0, si = 0$<br>3 <b>repita para</b> ( $j = 1, j \leq n, j = j + 1$ )<br><b>si</b> ( $j \bmod 2 = 0$ ) entonces<br>$sp = sp + j$<br><b>sino</b><br>$si = si + j$<br><b>fsi</b><br><b>frp</b> | ● <b>n</b> : Natural. Número máximo para calcular la suma de pares e impares.<br>● <b>j</b> : Natural. <b>Contador</b> de los números.<br>● <b>sp, si</b> : Natural. Suma de los números pares e impares, respectivamente. |
| 4                          | Escribir “Suma de pares entre 1 y “, $n$ , “ es “, $sp$  |  |
| 5                          | Escribir “Suma de impares es “, $si$   |  |
| 1                          | $n = 1 \rightarrow n = 1, sp = 0, si = 1$  | Caso exitoso   |
| 2                          | $n = 6 \rightarrow n = 6, sp = 12, si = 9$   | Caso exitoso   |

# Codificación en C

```
#include <stdio.h>

void main ()
{ unsigned int n, j, sp = 0, si = 0;
  do /* validación del valor dado a n */
  { printf ("Introduzca el valor de n entre 0 y 1\n");
    scanf ("%d", &n);
  }while (n < 1);
  for (j = 1; j <= n; j++)
  { if (j % 2 == 0)
      sp += j;
    else
      si += j;
  }
  printf ("Suma de pares entre 1 y %d es %f\n", n, sp);
  printf ("Suma de impares es %f\n", si);
}
```

# Ejercicio 4

Dado un número entero **n**, calcular su factorial (**n!**)

$$n! = \begin{cases} 1 & \text{si } n = 0 \\ \prod_{j=1}^n j & \text{si } n > 0 \end{cases}$$

## Análisis E-P-S

**Entrada:** Valor de **n**  $\in \mathbb{N}$ .

**Proceso:** Calcular factorial (si **n = 0** el **factorial = 1**)

---

**factorial = 1 x 2 x 3 x ... x (n-2) x (n-1) x n** si **n > 0**

**Salida:** Valor del **n!** (factorial  $\in \mathbb{N}$ )

# Diseño - algoritmo

| factorial |   |  |
|-----------|---|--|
| {pre:}    |   | {pos: $n \in \mathbb{N}$ }   |
| 1         | <b>hacer</b><br>Escribir “Introduzca un valor $\geq 0$ ”<br>Leer n<br><b>mientras</b> ( $n < 0$ )<br>2 factorial = 1<br>3 <b>repita para</b> ( $i = 1, i \leq n, i = i + 1$ )<br>factorial = factorial * i<br><b>frp</b><br>4 Escribir n, “! = “, factorial | ● <b>n</b> : Natural. Valor para calcular su factorial.<br>● <b>i</b> : Natural. Contador<br>● <b>factorial</b> : Natural. Valor del factorial de n. |
| 1         | $n = 0 \rightarrow \text{factorial} = 1$  | Caso exitoso   |
| 2         | $n = 5 \rightarrow \text{factorial} = 120$  | Caso exitoso   |

# Codificación en C

```
#include <stdio.h>
```

```
void main ()
```

```
{ int n, factorial = 1, i;
```

```
  do
```

```
  { printf ("Introduzca un valor > = 0\n");
```

```
    scanf ("%d", &n);
```

```
  }while (n < 0);
```

```
  for (i = 1; i <= n; i++)
```

```
    factorial *= i;
```

```
  printf ("%d! = %f\n", factorial);
```

```
}
```

# Ejercicios

1. El siguiente programa **lee una lista de números enteros y calcula su suma**. El **tamaño de la lista de números se lee antes** de comenzar a leer los números.

```
#include <stdio.h>
void main ()
{
    int suma, contador, n, numero;
    suma = 0, contador = 0;
    printf ("Introduzca el número de elementos\n");
    scanf ("%d",&n);
    while (contador < n)
    {
        printf ("Introduzca elemento:\n");
        scanf ("%f, &numero);
        suma += numero;
        contador++;
    }
    printf ("La suma es: %f\n", suma);
}
```

**Modifique el programa sustituyendo la estructura **while** por la estructura **FOR****



# + Ejercicios

Para cada uno de los siguientes problemas realizar el análisis E-P-S y diseño:

2. Calcular la conversión de grados Celsius a grados Fahrenheit comenzando desde  $5^{\circ}\text{C}$  hasta  $-5^{\circ}\text{C}$
3. Dadas las **notas** de **n** estudiantes correspondientes al segundo examen de **PR I** en el rango de 0 a 20. Calcular el número de estudiantes **sobresalientes** (16-20), el número de estudiantes **satisfactorios** (10-15) y el número de estudiantes **deficientes** (0-9)

# + Ejercicios

4. Calcular e imprimir la suma de los cuadrados de los 100 primeros números naturales

5. Dado el valor de  $n \geq 1$ . Calcular la suma de la serie

$$S = 1 + \sum_{i=1}^n \frac{1}{2^i}$$

6. Dada una lista de  $n$  números, determinar cual es el mayor

7. Dado un entero positivo  $N$ , calcular:

- La suma de los primeros  $N$  enteros no negativos pares.
- La suma de los primeros  $N$  enteros no negativos impares.
- El producto de los primeros  $N$  enteros positivos pares.
- El producto de los primeros  $N$  enteros positivos impares.

## + Ejercicios

Una compañía paga a sus empleados 1.000 Bs. por semana más una comisión por sus ventas en esa semana, la cual viene especificada en la siguiente tabla:

| ventas  | comisión                                       |
|---|--|
| $< 5000,00 \text{ Bs}$                                  | No hay comisión                                |
| $5.000,00 \leq \text{ventas} \leq 55.000,00 \text{ Bs}$ | 10% del monto total de sus ventas              |
| $> 55.000,00 \text{ Bs}$                                | $5.000,00 + 8\%$ del monto total de sus ventas |

• Suponga que la prima por Seguro de Salud se descuenta del sueldo de un empleado, de acuerdo a las siguientes normas:

|                 |                    |
|-----------------|--------------------|
| Soltero         | prima de 100,00 Bs |
| Casado sin hijo | prima de 200,00 Bs |
| Casado con hijo | prima de 300,00 Bs |

## + del ejercicio 8...

Suponga que el estado civil del empleado viene dado como dato de entrada de la siguiente manera:

- 1 si es soltero
- 2 si es casado

Suponga que la información si el empleado tiene o no hijos viene dada como dato de entrada de la siguiente manera:

- 0 si no tiene hijos
- 1 si tiene hijos

**Dados los datos de  $n$  empleados, donde para cada empleado se tiene: monto de sus ventas en una semana, estado civil y si tiene o no hijos, escribir un programa que lea los datos de cada empleado, determine y escriba:**

- a) La comisión del empleado de acuerdo al monto de sus ventas en una semana
- b) El sueldo neto semanal del empleado
- c) La prima por Seguro de Salud que se descuenta del sueldo del empleado
- d) El sueldo final que recibirá el empleado
- e) Las ventas totales semanal de la compañía

# Estructuras de repetición anidadas



Estructuras de repetición unas dentro de otras

## *Reglas de anidamiento*

- La(s) estructura(s) interna(s) debe(n) estar completamente contenida(s) en la estructura externa
- No deben existir solapamientos

# Ejemplos de anidamiento



(a)



(b)



(c)



(d)



(e)

# Ejercicio

Se conoce la población de cada una de las **cinco** ciudades más importantes de **veintiocho** países y se desea identificar e imprimir la población de la ciudad más grande (según el número de habitantes) de cada país

Tarea: Realizar el análisis **E-P-S**

# Diseño – algoritmo

| CiudadMasGrande |   |   |
|-----------------|---|---|
| {pre:}          |   | {pos: mayor $\geq 0$ }  |
| 1               | j = 1   | <ul style="list-style-type: none"> <li>● <b>mayor:</b> Natural. Mayor población de una ciudad de un país</li> <li>● <b>i, j:</b> Natural. Contadores</li> <li>● <b>pobCiudad:</b> Natural. Número de habitantes de una ciudad de un país</li> </ul> |
| 2               | <b>repita mientras</b> ( j $\leq 28$ )<br>mayor = 0<br><b>repita para</b> ( i = 1, i $\leq 5$ , i = i + 1)<br>Escribir “Introduzca población de la ciudad ”, i<br>Leer pobCiudad<br><b>si</b> (pobCiudad > mayor) <b>entonces</b><br>mayor = pobCiudad<br><b>fsi</b><br><b>frp</b><br>Escribir “Población mayor del país “, j, “ =“, mayor<br>j = j + 1<br><b>frm</b> |   |
| 1               | pobCiudad = 2000000, 1500000, 1300000, 2500000, 1900000 -> mayor = 2500000  | Caso exitoso  |



# Codificación en C

```
#include <stdio.h>

void main ()
{ int j = 1, mayor, i, pobCiudad;
  while (j ≤ 28)
  {   mayor = 0;
      for (i = 1; i ≤ 5; i++)
      {   printf("Introduzca población de ciudad %d\n",i);
          scanf (%d, &pobCiudad);
          if (pobCiudad > mayor) mayor = pobCiudad;
      }
      print ("Poblacion mayor del pais %d es %d\n", j,
mayor);
      j++;
  }
}
```