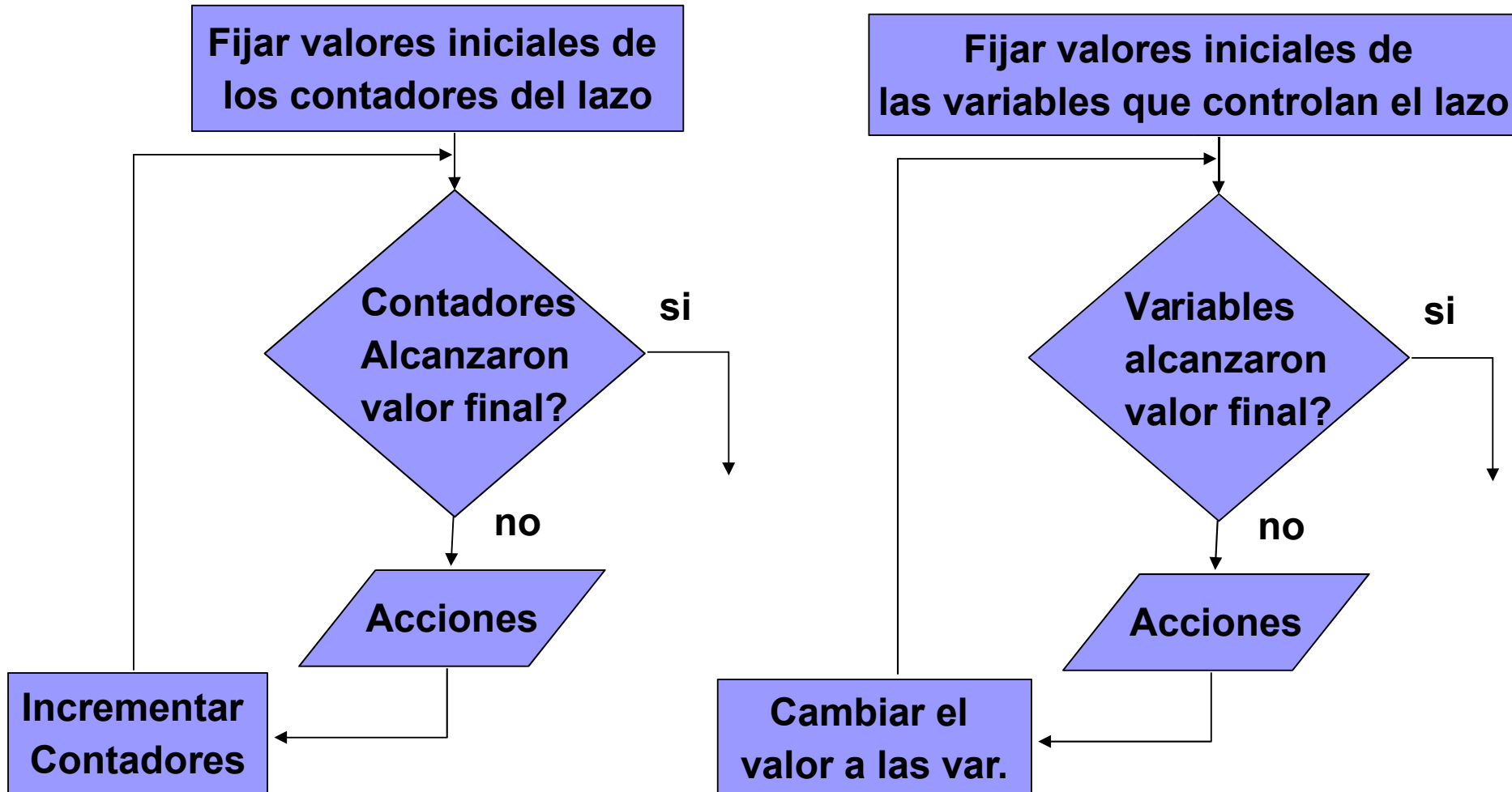


Estructuras de Repetición (Repita para)

Andrés Arcia
Departamento de Computación
Escuela de Ingeniería de Sistemas
Facultad de Ingeniería
Universidad de Los Andes

Estructura de Repetición

“REPITA PARA”



Estructura de Repetición

“REPITA PARA”

Pseudocódigo en español

```
Repita para (expr1; expr 2; expr3)
  S1
  ...
  Sn
fin_rp
```

Código en C

```
for (expr1; expr 2; expr3)
{
  S1
  ...
  Sn
}
```

Estructura de Repetición

“Repita para”

Se utiliza mayormente cuando se conoce de antemano el número de veces que se va a ejecutar el lazo. La estructura REPITA PARA requiere:

- Al menos un contador o condición inicial evaluada cierta para entrar al lazo (exp1).
- El incremento, decremento o expresión de cambio con la cual, cada vez que se termine una repetición, se evaluará la condición de salida (expr3).
- La condición que compruebe la existencia del valor final del contador del lazo (expr2).

Estructura de Repetición

“REPITA PARA”

Repita para (expr1; expr2; expr3)

expr1: Inicializar 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.

Estructura de Repetición

“REPITA PARA”

Nombre de
la variable de
control



Valor final
de la variable
de control



Repita para (i = 1; i <= 10; i = i + 1)



Valor inicial
de la variable
de control

Incremento
de la variable
de control

Estructura de Repetición

“REPITA PARA”

Pseudocódigo en Español	Código en C
<pre>Repita para (I=1; I≤10; I=I+1) Escribir (I) fin_rp</pre>	<pre>for (int I=1; I <= 10; I++) printf("%i", I); // I es la variable de control</pre>
<pre>Repita para (I=0; I <= n-1; I=I+1) Escribir ("Numero?") Leer (numero) fin_rp</pre>	<pre>for (I=0; I <= n-1; I++) { // I es la variable de control printf("numero?\n"); scanf("%i", &numero); }</pre>

Estructura de Repetición

“REPITA PARA”

Ejemplos

```
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("i: %i j: %i",i,j); }  
// Variación simultanea de dos contadores.  
  
for (unsigned counter = 1; counter <= 10; counter++)  
// Variar el contador (counter) de 1 a 10  
// en incrementos de 1  
  
for (i = 100; i >= 1; i--)  
// Variar el contador (i) de 100 a 1  
// en decrementos de 1
```


Estructura de Repetición

“REPITA PARA”

Ejemplos

```
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
```

Estructura de Repetición

“REPITA PARA”: Ejemplo 1

Calcular el seno para los valores de $0.0 \leq x \leq 1.6$ con incrementos de 0.1.

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

void main ()
{
    int i;
    float x;

    for (i = 0; i <= 16; i++)
    {
        x = 0.1 * i;
        printf("x: %f sin(x): %f", x, sin(x));
    }
}
```

Estructura de Repetición

“REPITA PARA”: Ejemplo 2

Sumar todos los enteros pares desde 2 hasta 100

Análisis E-P-S

Entradas: No hay.

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

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

Salidas: Suma de los enteros pares ($AP \in \mathbb{Z}^+$)

Estructura de Repetición

“REPITA PARA”: Ejemplo 2

Algoritmo

0. Inicio
1. AP = 0
2. Repita para (num = 2; num <= 100; num = num + 2)
 AP = AP + num
 fin_RP
3. Escribir (“Suma de enteros pares = ”, AP)
4. Fin

Estructura de Repetición

“REPITA PARA”: Ejemplo 2

Codificación

```
#include <stdio.h>

void main()
{
    unsigned int AP = 0, num;

    for (num = 2; num <= 100; num += 2)
        AP += num;
    printf("Suma de numeros pares %i \n",AP);
}
```

Estructura de Repetición

“REPITA PARA”: Ejemplo 3

Dadas las notas de n estudiantes de las materias de PD-I, Probabilidades y Cálculo 40. Calcular la media de cada estudiante.

Análisis E-P-S

Entradas: Número de estudiantes ($n \in \mathbb{Z}^+$), nota de las tres materias para cada estudiante ($\text{nota1} \in \mathbb{Z}^+$, $\text{nota2} \in \mathbb{Z}^+$, $\text{nota3} \in \mathbb{Z}^+$).

Proceso: Para cada estudiante hacer:

$$\text{Calcular media} = \frac{\text{nota1} + \text{nota2} + \text{nota3}}{3}$$

Salidas: La media para cada estudiante ($\text{media} \in \mathbb{R}$)

Estructura de Repetición

“REPITA PARA”: Ejemplo 3

0. Inicio

1. Escribir ("Introduzca el numero de estudiantes")

2. Leer (n)

3. Repita para ($k = 1$; $k \leq n$; $k = k + 1$)

 Escribir ("Para el estudiante", k)

 Escribir ("Introduzca la nota de PD-I")

 Leer (nota1)

 Escribir ("Introduzca la nota de Probabilidades")

 Leer (nota2)

 Escribir ("Introduzca la nota de Calculo 40")

 Leer (nota3)

$media = (nota1 + nota2 + nota3) / 3$

 Escribir ("Media del estudiante", k, "= ", media)

 fin_RP

4. Fin

Estructura de Repetición

“REPITA PARA”: Ejemplo 3

Codificación

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

#include <stdio.h>

void main ()
{
    unsigned integer n, k, nota1, nota2, nota3;
    float media;

    printf("Introduzca el numero de estudiantes\n");
    scanf("%i", &n);
```


Estructura de Repetición

“REPITA PARA”: Ejemplo 3

```
for (k = 1; k <= n; k++)
{
    printf("Para el estudiante" << k << endl;
    printf("Introduzca la nota de PD-I" << endl;
    scanf("%i",&nota1);
    printf("Introduzca la nota de Probabilidades" <<
endl;
    scanf("%i",&nota2);
    cout << "Introduzca la nota de Calculo 40" << endl;
    scanf("%i",&nota3);
    media = ((float)nota1 + (float)nota2 +
            (float)nota3)/3.0;
    printf("Media del estudiante %i = %i\n",k,media);
}
}
```

Estructura de Repetición

“REPITA PARA”: Ejemplo 4

Calcular independientemente la suma de los números pares e impares comprendidos entre 1 y n.

Análisis E-P-S

Entradas: Valor de $n \in \mathbb{Z}^+$.

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

Si número es par $\text{numPar} = \text{numPar} + \text{numero}$

Si número es impar $\text{numImpar} = \text{numImpar} + \text{numero}$

Salidas: Suma de los números pares ($\text{numPar} \in \mathbb{Z}^+$) y suma de los números impares ($\text{numImpar} \in \mathbb{Z}^+$).

Estructura de Repetición

“REPITA PARA”: Ejemplo 4

Algoritmo

```
0. Inicio
1. Repita
    Escribir ("Introduzca el valor de n")
    Leer (n)
    mientras (n < 1)
2. numero = 0
3. numPar = 0
4. numImpar = 0
```

Estructura de Repetición

“Repita para”: Ejemplo 4

5. Repita para ($j = 1; j \leq n; j = j + 1$)
 numero = numero + 1
 Si (numero mod 2 = 0) entonces
 numPar = numPar + numero
 sino
 numImpar = numImpar + numero
 fin_si
fin_RP
6. Escribir (“Suma de numeros pares”, numPar)
7. Escribir (“Suma de numeros impares”, numImpar)
8. Fin

Estructura de Repetición

“Repita para”: Ejemplo 4

Codificación

```
#include <stdio.h>
void main ()
{
    unsigned int n, j, numero = 0, numPar = 0,
    numImpar = 0;
    do          // validación del valor dado a n
    {
        printf("Introduzca un valor entero mayor que
        cero");
        scanf("%i",n);
    } while (n < 1);
```

Estructura de Repetición

“Repita para”: Ejemplo 4

```
for (j = 1; j <= n; j++)
{
    numero ++;
    if (numero % 2 ==0)
        numPar += numero;
    else
        numImpar += numero;
}
printf("Suma de numeros pares %i",numPar);
printf("Suma de numeros impares %i",numImpar);
}
```

Estructura de Repetición

“Repita para”: Ejemplo 5

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

Análisis E-P-S

Entradas: Valor de $n \in \mathbf{N}$.

Proceso: Calcular

$$\text{factorial} = n \times (n-1) \times (n-2) \times \dots \times 3 \times 2 \times 1$$

Salidas: Valor del $n!$ ($\text{factorial} \in \mathbf{N}$)

Estructura de Repetición

“Repita para”: Ejemplo 5

Algoritmo

0. Inicio

1. Hacer

 Escribir (“Introduzca un numero ≥ 0 ”)

 Leer (n)

 mientras (n < 0)

2. Factorial = 1

3. Repita para (I = 1, I \leq n, I = I + 1)

 Factorial = Factorial * I

 fin_RP

4. Escribir (“n! = ”, Factorial)

5. Fin

Estructura de Repetición

“Repita para”: Ejemplo 5

Codificación

```
#include <stdio.h>

void main ()
{
    int n, Factorial = 1, I;

    do {
        printf("Introduzca un numero entero >= 0");
        scanf("%i", &n);
    } while (n < 0);
    for (I = 1; I <=n; I++)
        Factorial *= I;
    printf("n! = %i", Factorial);
}
```

Ejercicios

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: ");
    scanf("%i", &n);
```

Ejercicios

```
while (contador < n)
{
    printf("Introduzca elemento: ");
    scanf("%i", &numero);
    suma += numero;
    contador++;
}
printf("La suma es: %i", 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, algoritmo y codificación.

Calcular la conversión de grados Celsius a grados Fahrenheit comenzando desde 5° C y decrementando hasta -5° C .

Dadas las notas de n estudiantes correspondientes al segundo examen de PD-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 no satisfactorios (0-9).

Ejercicios

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

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

$$1 + \frac{1}{2} + \frac{1}{3} + \frac{1}{4} + \frac{1}{5} + \dots + \frac{1}{n}$$

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

$$1 + \frac{1}{3} + \frac{1}{5} + \frac{1}{7} + \frac{1}{9} + \dots + \frac{1}{n}$$

Ejercicios

Una persona invierte una cierta cantidad de bolívares en una cuenta de ahorros que produce intereses del 5% (tasa de interés). Suponiendo que todos los intereses quedan en depósito, calcular el imprimir la cantidad de dinero que habrá en la cuenta al final de cada año durante 10 años. Utilice la siguiente fórmula para determinar dichas cantidades.

$$a = p(1 + r)^n$$

donde

p = cantidad original de bolívares invertida

r = tasa de interés anual

n = número de años

Ejercicios

Modifique el programa del ejercicio anterior de modo que repita sus pasos para tasas de interés del 6%, 7%, 8%, 9% y 10%.

Calcular la suma de los 10 primeros términos de la serie:

$$\sum_{j=1}^{10} \frac{1}{3j+1}$$

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

Ejercicios

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.

La suma de los cuadrados de los primeros N enteros positivos.

Una compañía paga a sus empleados 20.000 Bs. por semana mas una comisión por sus ventas en esa semana, la cual viene especificada en la siguiente tabla:

Ejercicios

Ventas	Comisión
Ventas = 5.000 Bs.	No hay comisión
$5.000 \text{ Bs.} \leq \text{ventas} \leq 55.000$	10% de las ventas
Ventas > 55.000 Bs.	5.000 + %8 de las ventas

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

Ejercicios

Características	Prima
Soltero	1.000 Bs.
Casado sin hijo	2.500 Bs.
Casado con hijos	5.000 Bs.

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

Ejercicios

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 utilizando funciones 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.

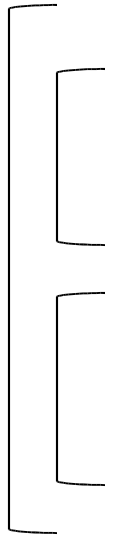
Reglas de anidamiento

La(s) estructura(s) interna(s) debe(n) estar completamente contenida(s) en la estructura externa.

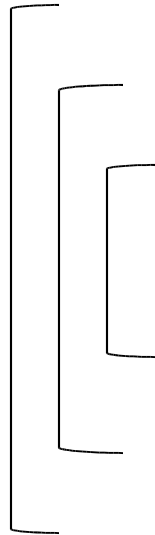
No deben existir solapamientos.

Estructuras de Repetición Anidadas

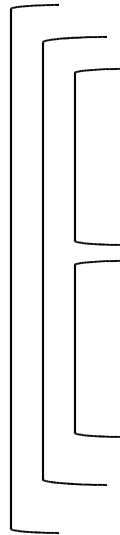
Ejemplos



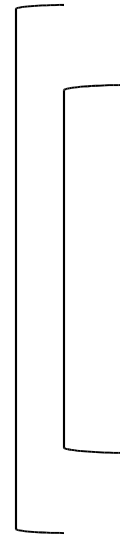
(a)



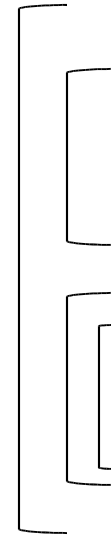
(b)



(c)



(d)



(e)

Estructuras de Repetición Anidadas: Ejemplo 1

Un triángulo rectángulo puede tener lados que sean enteros. El conjunto de tres valores enteros para los lados de un triángulo rectángulo es una tripleta pitagórica. Estos tres lados deben satisfacer la relación de que la suma de los cuadrados de dos de los lados (cateto1, cateto2) es igual al cuadrado de la hipotenusa. Encontrar todas las tripletas pitagóricas tal que $\text{cateto1} \leq 500$, $\text{cateto2} \leq 500$ e $\text{hipotenusa} \leq 500$.

Estructuras de Repetición Anidadas: Ejemplo 1

Algoritmo

0. Inicio

1. Repita para ($i = 1, i \leq 500, i = i + 1$)

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

$z = \text{raiz}(i*i + j*j)$

$z2 = z*z$

Si ($z2 = i*i + j*j$) && ($z2 \leq 500$) entonces

Escribir ("Tripla pitagórica", "(i , j , z)")

fin_RP

fin_RP

2. Fin

Estructuras de Repetición Anidadas: Ejemplo 1

Codificación

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

```
#define MAX 500
```

```
int main()
```

```
{
```

```
    int i, j;
```

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

```
    {
```

```
        {
```

```
            if (sqrt(float(i*i+j*j)) - int(sqrt(float(i*i+j*j))) == 0.0)
```

```
                printf("tripla: %i, %i, %i", i, j, i*i+j*j);
```

```
        }
```

```
    }
```


Estructuras de Repetición Anidadas: Ejemplo 2

Se conoce la población de cada una de las cinco ciudades mas importantes de veintiocho países y se desea identificar e imprimir la población de la ciudad mas grande (en número de habitantes) de cada país.

REALIZAR EL ANALISIS E-P-S

Estructuras de Repetición Anidadas: Ejemplo 2

Algoritmo

```
0. Inicio
1. j = 1
2. Repita mientras (j ≤ 28)
    mayor = 0
    Repita para (k = 1, k ≤ 5, k = k + 1)
        Escribir ("Introduzca población de ciudad",
k)
        Leer (pob_ciudad)
        Si (pob_ciudad > mayor)
            mayor = pob_ciudad
        fin_si
    fin_RP
    Escribir ("Poblacion mayor del pais", j, "=",
mayor)
    j = j + 1
    fin_RM
```

Estructuras de Repetición Anidadas: Ejemplo 2

Codificación

```
#include <stdio.h>

void main () {
    int j = 1, mayor, k;
    while (j ≤ 28)
    {
        mayor = 0;
        for (k = 1; k ≤ 5; k++)
        {
            printf("Introduzca
                    población de
                    ciudad %i:", k);
            scanf("%i", pob_ciudad);
            if (pob_ciudad > mayor)
                mayor = pob_ciudad;
        }
    }
}
```

```
        printf("Poblacion mayor
del pais %i = %i", j,
mayor);
        j++;
    }
}
```

Ejercicios

Calcular el factorial de cada uno de una lista de m números dados.

Calcular el máximo común divisor de n pares de números (x_1, y_1) , (x_2, y_2) , \dots , (x_n, y_n) mediante el algoritmo de Euclides:

Sean los números A y B . El método para hallar el máximo común divisor (mcd) de dos números A y B por el método de Euclides es:

Dividir el número mayor por el menor. Si el residuo de la división es 0, el número menor es el mcd.

Si la división no es exacta, se divide el número menor por el residuo de la división anterior.

Se siguen los pasos anteriores hasta obtener un resto cero. El último divisor es el mcd buscado.