



**Programación Declarativa**  
**Ingeniería Informática**  
**Especialidad de Computación**  
**Cuarto curso. Primer cuatrimestre**

**Escuela Politécnica Superior de Córdoba**  
**Universidad de Córdoba**

**Curso académico: 2014 - 2015**

---

**Práctica número 3.- Iteración, recursión y funciones usadas como parámetros o devueltas como resultados**

**1. Números amigos**

- Dos números naturales son **amigos** si la suma de los divisores de uno es igual al otro número y viceversa.
- El menor par de números amigos es el formado por el 220 y 284:
  - Suma de los divisores de 220 (excepto 220):  
 $1 + 2 + 4 + 5 + 10 + 20 + 11 + 22 + 44 + 55 + 110 = 284$
  - Suma de los divisores de 284 (excepto 284):  
 $1 + 2 + 4 + 71 + 142 = 220$
- Otros números amigos son (1184 y 1210) (6232 y 6368), (2620 y 2924)...
- a. Codifica una función **recursiva** denominada **suma-divisores** para calcular la suma de los divisores de un número natural (excepto el propio número)
- b. Utiliza la función **suma-divisores** para codificar un predicado denominado **amigos?** que permita comprobar si dos números son o no amigos.
- c. Utiliza el predicado **amigos?** para codificar un predicado denominado **perfecto?** que permita comprobar si un número es perfecto, es decir, es igual a la suma de sus divisores inferiores a él.  
Por ejemplo: el número 28 es perfecto porque  
 $28 = 14 + 7 + 4 + 2 + 1$ 
  - Nota: si se utiliza el par (28 y 28) y se comprueba que son "amigos" entonces 28 será "perfecto".

2. (\*) Un número es **primo** si no tiene divisores propios menores que su raíz cuadrada. Codifica dos predicados que determinen si un número es primo no:
- a. El primer predicado se denominará **primo-iterativo?** y utilizará la forma especial "**do**" para crear una función iterativa.
  - b. El segundo predicado se denominará **primo-recursivo?** y será una función recursiva

**3. (\*) El número áureo**

- El número áureo se define como

$$\varphi = \frac{1 + \sqrt{5}}{2} = 1,618033989.....$$

- El número áureo también se puede calcular mediante la siguiente suma infinita

$$\varphi = \sqrt{1 + \sqrt{1 + \sqrt{1 + \sqrt{1 + \sqrt{1 + \sqrt{1 + \dots}}}}}}}$$

- Codifica una función **iterativa** denominada “**suma-aureo**” que permita calcular el número áureo usando la suma anterior. La función recibirá como parámetro el número de sumandos.

#### 4. (\*) Número e

- Considérese el término general de una sucesión numérica que converge al número e: 2.718281...

$$a_n = (1 + 1/n)^n$$

- Codifíquense las siguientes funciones:
  - **término-número-e**
    - Calculará el término n-ésimo de la sucesión numérica.
    - Recibirá como parámetro el valor de **n**.
  - **límite-sucesión-número-e-iterativa**
    - Se debe codificar una función **iterativa** que permita calcular el límite de la sucesión numérica que converge al número e.
    - La función debe recibir como argumento la **cota de error**, que permitirá terminar la función cuando dos elementos consecutivos de la sucesión disten menos que dicha cota de error.

#### 5. (\*) Límite de una sucesión numérica convergente

- Codifica una función **iterativa** denominada “**límite-iterativa**” que permita calcular una aproximación al límite de **cualquier** sucesión numérica convergente.
- La función debe recibir como argumentos a:
  - Una **función** que represente el término general de la sucesión numérica convergente.
  - La **cota de error**, que permitirá terminar la función cuando dos elementos consecutivos de la sucesión disten menos que dicha cota de error.
- ¿Cómo se llamaría a la función “límite-iterativa” si se desea calcular el límite de la sucesión numérica cuyo término general es  $a_n = (1 + 1/n)^n$  con una cota de error de 0.001?

#### 6. (\*) Aproximaciones al número $\pi$ :

- **Leibniz** propuso la siguiente serie numérica para calcular una aproximación a  $\pi/4$ :

$$\frac{\pi}{4} = \sum_{n=1}^{\infty} f(n) = 1 - \frac{1}{3} + \frac{1}{5} - \frac{1}{7} + \dots$$

- Escribe una función denominada **término-Leibniz** que reciba un número entero **n** y calcule el “n-ésimo” término de la serie de Leibniz.
- Escribe una función **iterativa**, denominada **Leibniz-pi-1**, que reciba como parámetro el número de términos **n** de la serie propuesta por Leibniz que se deseen sumar.
- Escribe una función **iterativa**, denominada **Leibniz-pi-2**, que reciba como parámetro una **cota de error** y termine cuando la diferencia entre dos términos consecutivos de la sucesión sea inferior a dicha cota.

- **Wallis** propuso utilizar la siguiente serie para calcular una aproximación a  $\pi/4$ :

$$\frac{\pi}{4} = \frac{2}{3} \times \frac{4}{3} \times \frac{4}{5} \times \frac{6}{5} \times \frac{6}{7} \times \frac{8}{7} \times \frac{8}{9} \times \dots$$

- Codifica una función denominada **factor-Wallis** que reciba como parámetro un número natural **n** y devuelva como resultado el n-ésimo factor de la sucesión de Wallis.

Por ejemplo:

(factor-Wallis 1) ==> 2/3

(factor-Wallis 2) ==> 4/3

...

(factor-Wallis 5) ==> 6 / 7

- b. Escribe una función **iterativa** denominada **Wallis-iterativa** que reciba como parámetro un **número** natural que indicará cuántos factores se han de multiplicar.
- c. Escribe función **recursiva de cola** denominada **Wallis-recursiva** que reciba como parámetro una cota de error, de forma que la función terminará su ejecución cuando el factor que se vaya a multiplicar esté comprendido entre los siguientes valores:

$$1 - \text{cota} < \text{factor} < 1 + \text{cota}$$

- o **Observación:** la sucesión de Wallis converge "muy lentamente".

- **Fracción continua**

- o Se puede obtener una aproximación al número  $\pi$  usando la siguiente fracción continua:

$$\frac{4}{\pi} = 1 + \frac{1}{3 + \frac{4}{5 + \frac{9}{7 + \dots}}}$$

- o Codifica una función **iterativa** que permita calcular una aproximación a  $4/\pi$  usando la fracción continua anterior. La función recibirá como parámetro el número de fracciones continuas que debe calcular.

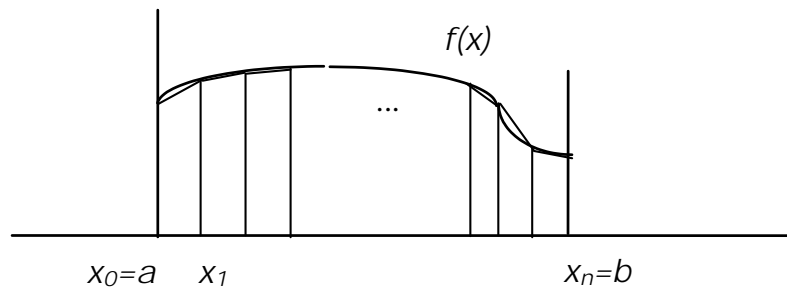
7. Codifica una función **iterativa**, denominada **integral**, que reciba cuatro parámetros:

- Los dos extremos de un intervalo: **a** y **b**
- Una función **f** que sea positiva en el intervalo **[a,b]**
- Un número "n"

y devuelva la aproximación a la integral según el **método de los trapecios**.

$$\int_a^b f(x) dx = \sum_{i=0}^{n-1} \left( \frac{f(x_i) + f(x_{i+1})}{2} \right) * h$$

donde  $h = (b - a) / n$  y  $x_i = a + i * h$



- ¿Cómo se calcularía el área de la función  $f(x) = 3x^2 + 1$  definida en el intervalo  $[1,3]$ ?

8. (\*) **Simpson** propuso un método para calcular la aproximación a la integral de una función en un intervalo  $[a,b]$ :

$$\int_a^b f(x) = (y_0 + 4y_1 + 2y_2 + 4y_3 + 2y_4 + \dots + 2y_{n-2} + 4y_{n-1} + y_n) \frac{h}{3}$$

donde

- $n$  es un número es par
- $h = (b - a) / n$

$$- y_k = f(a + h k)$$

- a. Codifica una función denominada **término-Simpson** que calcule el k-ésimo término de la sucesión de Simpson a partir de los siguientes parámetros:
    - o La función **f**
    - o El extremo inferior del intervalo **a**
    - o El incremento **h**
    - o El número natural **k**
  - b. Utiliza la función **término-Simpson** para codificar una función **iterativa** denominada **Simpson-iterativo** que calcule la aproximación a la integral de una función **f** en un intervalo **[a, b]** utilizando **n** términos.
  - c. ¿Qué valor se obtiene al calcular el área de la función  $f(x) = 3x^2$  en el intervalo  $[0, 1]$  si se utiliza el valor  $n = 100$ ?
9. (\*) El algoritmo de **Euclides** permite calcular el máximo común divisor (M.C.D.) de dos números naturales:

*Dados "a" y "b", dos números naturales, donde "a >= b",  
si  $a = c b + r$  entonces  $M.C.D.(a, b) = M.C.D.(b, r)$ .*

- El algoritmo concluirá cuando el segundo argumento sea cero, siendo el máximo común divisor el primer argumento. Si "a" es menor que "b", se calculará el M.C.D.(b,a).
- Ejemplo: cálculo del máximo común divisor de 630 y 198

<b>a</b>	<b>630</b>	198	36	<b>18</b>
<b>b</b>	<b>198</b>	36	18	0
<b>r</b>	36	18	0	

$$M.C.D.(630,198) = 18$$

- a. Codifica una función **iterativa**, denominada **mcd-iterativo**, que permita calcular el máximo común divisor de dos números.
  - b. Codifica una función **recursiva**, denominada **mcd-recursivo**, que permita calcular el máximo común divisor de dos números.
10. (\*) Codifica una función denominada **incremento-funcional** que reciba una función **f** como parámetro y devuelva como resultado la función que calcularía la siguiente expresión

$$\frac{f(x+1) - 2f(x) + f(x-1)}{4}$$

¿Cómo se invocaría la función **incremento-funcional**? Pon un ejemplo.

11. (\*) Codifica una función denominada **diferencia-simétrica** que reciba como parámetros dos funciones **f** y **g** y devuelva como resultado la función que calcularía la siguiente expresión:

$$|f(x) - g(x)|$$

¿Cómo se invocaría la función **diferencia-simétrica**? Pon un ejemplo.