

## Lista de Ejercicios

### Funciones, Condicionales, iteración y recursividad

1. Escriba un programa que lea un número entero N y muestre los primeros N elementos de una secuencia de Fibonacci en la pantalla como el siguiente:

```
=====
          SECUENCIA DE FIBONACCI
=====
Cuántos números quieres mostrar en la secuencia de Fibonacci? 15

0 -> 1 -> 1 -> 2 -> 3 -> 5 -> 8 -> 13 -> 21 -> 34 -> 55 -> 89 -> 144 -> 233 -> 377
```

2. Escriba una función que recibiendo la longitud de los 3 lados de un triángulo nos devuelva que tipo de triángulo es:
  - **Escaleno**: todos los lados distintos
  - **Equilátero**: todo los lados iguales
  - **Isósceles**: solo 2 lados iguales
3. Escriba una función que recibiendo del usuario un número hasta el 1000 lo convierta en número romano y muestre el resultado por pantalla.  
Para la solución observe que a los números enteros en la lista **N** corresponden respectivamente los números romanos en la lista **R**  
**N: 1000, 900, 500, 400, 100, 90, 50, 40, 10, 9, 5, 4, 1**  
**R: M, CM, D, CD, C, XC, L, XL, X, IX, V, IV, I**
4. Escribir una función que tome tres calificaciones de un estudiante como parámetro y una letra. Si la letra es **A**, la función debe calcular la media aritmética de las calificaciones del estudiante, si es **P**, debe calcular el **promedio ponderado, con pesos 5, 3 y 2**
5. Escribir una función no recursiva que tome un número entero **positivo impar N** y devuelva el factorial doble de ese número. El factorial doble se define como el producto de todos los números naturales impares desde 1 hasta algún número natural impar N.  
Por ejemplo el doble factorial de 5 es:  $5! = 1 * 3 * 5 = 15$
6. Escribir una función que tome un número entero mayor que zero y retorna la suma de todos sus dígitos.  
Por ejemplo, el número 370 corresponderá al valor 10 ( $3+7+0$ ). Si el parámetro leído no es un número entero mayor que zero, el programa terminará con el mensaje "**Número Inválido**"

7. Carrera de caballos: la pista tiene 1000 metros y corren 4 caballos. Cada metro es una iteración. En cada iteración y por cada caballo el mismo puede avanzar de 1 a 5 metros (**import random y use la función randint**). Agregar al final de la iteración una demora del programa de 1 segundo (**import time y use la función sleep()**) que permita ir viendo por pantalla en que metro está cada caballo dentro de la pista. Cuando el último caballo llegue a los 1000 metros, mostrar como quedó conformado el podio.



8. Imagine un Buffer de impresión con una capacidad máxima para almacenar 20 documentos, simule la gestión de la impresora en caso de falla a través de una cola circular del tipo **FIFO (First In First Out)**, asuma que los documentos que superan la capacidad máxima se pierden. La selección del usuario será **A, I o S**, A para agregar documento, I imprimir y S salir del programa. En el caso de la operación Agregar, el usuario debe informar el nombre del documento que ingresará a la cola.