**FUNDAMENTOS DE**

**PROGRAMACIÓN**

**LABORATORIO N° 12**

**FUNCIONES DEFINIDAS POR EL USUARIO (UDF)**

**CODIGO DEL CURSO:**



|  |  |  |
| --- | --- | --- |
| ***Alumno(s)*** | | ***Nota*** |
|  | |  |
|  | |  |
|  | |  |
| ***Grupo*** |  | |
| ***Ciclo*** |  | |
| ***Fecha de entrega*** |  | |

**I.- OBJETIVOS:**

* Elaborar y diseñar programas en Python que hagan uso de funciones definidas por el usuario

**II.- SEGURIDAD:**

|  |  |
| --- | --- |
|  | **Advertencia:**  **En este laboratorio está prohibida la manipulación del hardware, conexiones eléctricas o de red; así como la ingestión de alimentos o bebidas.** |

**III.- FUNDAMENTO TEÓRICO:**

1. Revisar el texto guía que está en el campus Virtual.

**IV.- NORMAS EMPLEADAS:**

1. No aplica

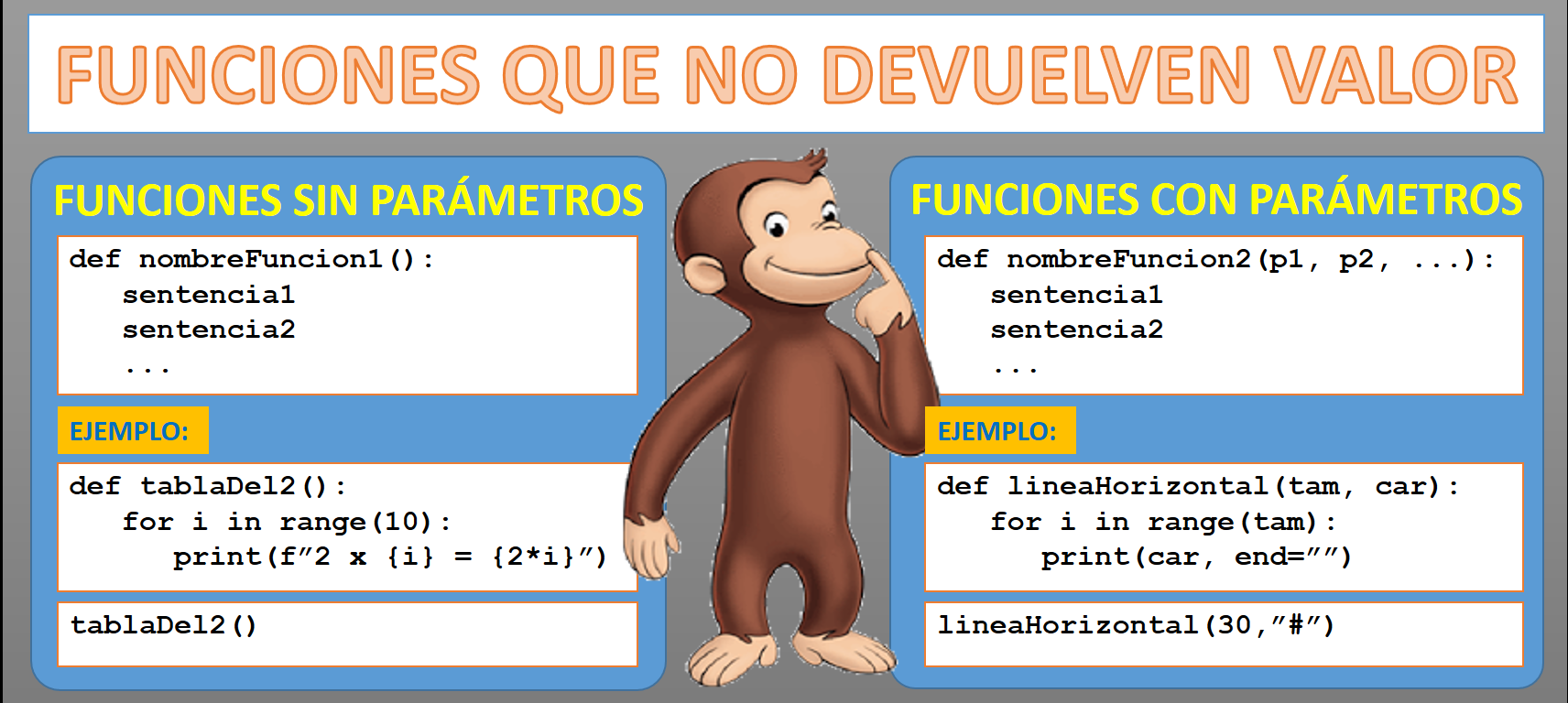
**V.- RECURSOS:**

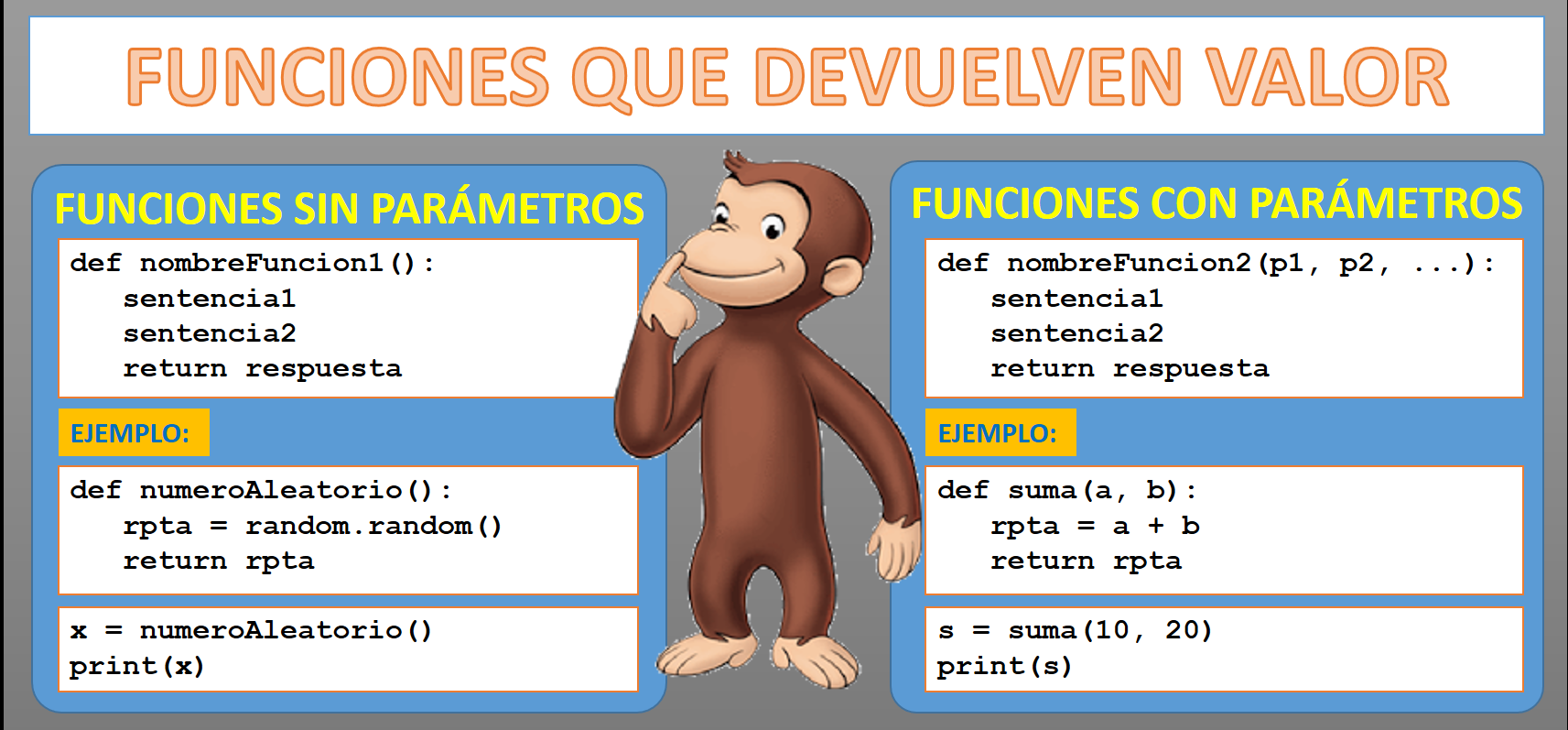
* En este laboratorio cada alumno trabajará con un equipo con Windows 10.

**VI.- METODOLOGÍA PARA EL DESARROLLO DE LA TAREA:**

* El desarrollo del laboratorio es individual.

**VII.- MARCO TEÓRICO:**





**VIII.- PROCEDIMIENTO:**

Crear un nuevo proyecto llamado **“Lab12”**. Crear, dentro de este proyecto, un programa **Python** para cada ejercicio.

**EJERCICIO RESUELTO:**

Implemente un programa denominado **ejemplo01.py** el cual permitirá:

* + Realizar el ingreso de “n” notas.
  + Mostrar un listado de las notas obtenidas
  + Calcular y mostrar el promedio de las notas

**Solución**:

* 1. Implementaremos la estructura general del programa considerando el uso de una función para cada requerimiento especificado en el enunciado:

|  |
| --- |
| **Código en Python** |
| *# Variable global para almacenar las notas* **notas = []**  *# PROGRAMA PRINCIPAL*  **ingresoNotas() #** Realizar el ingreso de “n” notas **listadoNotas() #** Mostrar un listado de las notas obtenidas **calcularPromedio() #** Calcular y mostrar el promedio de las notas |

* 1. Al intentar ejecutar el programa anterior se muestra un **ERROR**, ya que las tres funciones incluidas al final del programa **no existen en Python**. A continuación procederemos a crear la estructura general de estas tres funciones en el programa. Si ejecuta el programa ya no se genera **ERROR**, pero no se visualiza ningun resultado en pantalla

|  |
| --- |
| **Código en Python** |
| *# Variable global para almacenar las notas* notas = []  **def** ingresoNotas():  **pass  def** listadoNotas():  **pass  def** calcularPromedio():  **pass** *# PROGRAMA PRINCIPAL* ingresoNotas() **#** Realizar el ingreso de “n” notas listadoNotas() **#** Mostrar un listado de las notas obtenidas calcularPromedio() **#** Calcular y mostrar el promedio de las notas |

**Nota**: Se hace uso de la sentencia **pass** en cada función la cual representa por ahora el contenido temporal de cada función. Eliminaremos esta función cuando se implemente el código de cada función

* 1. Reemplace el contenido de la función **ingresoNotas**() con el siguiente contenido:

|  |
| --- |
| **Código en Python** |
| **def ingresoNotas**():  **n = int(input('Ingrese numero de notas: '))  for i in range(n):  nota = int(input(f'Ingrese nota {i+1}: '))  notas.append(nota)** |

**Ejecute** el programa y verifique que ahora únicamente puede **ingresar las notas**.

* 1. Reemplace el contenido de la función **listadoNotas**() con el siguiente contenido:

|  |
| --- |
| **Código en Python** |
| **def listadoNotas():  for nota in notas:  print(nota, ' ', end="")  print()** |

**Ejecute** el programa y verifique que ahora puede **ingresar las notas** y tambien se muestra el **listado de las notas** ingresadas.

* 1. Finalmente reemplace el contenido de la tercera función **calcularPromedio**() con el siguiente contenido:

|  |
| --- |
| **Código en Python** |
| **def calcularPromedio():  s = 0  for nota in notas:  s += nota  prom = s // len(notas)  print('Promedio de notas:', prom)** |

**Ejecute** el programa y verifique que ahora puede **ingresar las notas**, se muestra el **listado de las notas** ingresadas y se muestra el resultado del cálculo del **promedio** de las notas obtenidas:

* 1. Codigo del programa resultante:

|  |
| --- |
| **Código en Python** |
| ***# Variable global para almacenar las notas* notas = []  def ingresoNotas():  n = int(input('Ingrese numero de notas: '))  for i in range(n):  nota = int(input(f'Ingrese nota {i+1}: '))  notas.append(nota)  def listadoNotas():  for nota in notas:  print(nota, ' ', end="")  print()  def calcularPromedio():  s = 0  for nota in notas:  s += nota  prom = s // len(notas)  print('Promedio de notas:', prom)  *# PROGRAMA PRINCIPAL* ingresoNotas() *# Realizar el ingreso de “n” notas* listadoNotas() *# Mostrar un listado de las notas obtenidas* calcularPromedio() *# Calcular y mostrar el promedio de las notas*** |

**IX.- PROCEDIMIENTO:**

**EJERCICIOS DE APLICACIÓN:**

1. Elabore una función que tome como argumento dos números enteros y devuelva el mayor.

|  |
| --- |
| **Programa en Python** |
|  |

1. Elabore una función que tome como argumento tres números enteros y devuelva el mayor

|  |
| --- |
| **Programa en Python** |
|  |

1. Elabore una función que calcule la longitud de una lista o una expresión de texto ingresada por teclado.

|  |
| --- |
| **Programa en Python** |
|  |

1. Elabore una función que lea un carácter y devuelva True si es una vocal, de lo contrario que devuelva False

|  |
| --- |
| **Programa en Python** |
|  |

1. Elabore una funcion1() y una función2() que sumen y multipliquen respectivamente todos los números de una lista. Por ejemplo: funcion1([1,2,3,4]) debería imprimir 10 y funcion2([1,2,3,4]) debería devolver 24.

|  |
| --- |
| **Programa en Python** |
|  |

1. Implemente un programa Python, el cual
   1. incluirá una función **linea** con el siguiente encabezado: **linea(car\_izq, car\_centro, car\_der, tam)**, esta función podrá trazar una línea horizontal como se muestra en el siguiente ejemplo, se podrán especificar 3 caracteres para armar el trazado de la línea, carácter izquierdo, carácter central, carácter derecho y tam indica el tamaño total de la línea:

**car\_centro**

**car\_der**

**car\_izq**

+--------------------------------------+

**tam**

Ejemplo de uso:

**linea(’+’, ’-’, ’+’,40)**

|  |
| --- |
| **Código en Python** |
|  |

**OBSERVACIONES:**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**CONCLUSIONES:**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_**

**"Dime y lo olvido, enséñame y lo recuerdo, involúcrame y lo aprendo”. Benjamín Franklin (1706-1790)**