



**FACULTAD DE INGENIERIA Y CIENCIAS EXACTAS**

**Departamento de Tecnología Informática**

# **INTRODUCCIÓN A LA PROGRAMACIÓN**

**Guía de Trabajos Prácticos**

**Segundo Parcial**

V06.12.20

### **Bibliografía sugerida:**

#### **Básica**

- Joyanes Aguilar, Luis: Fundamentos de Programación. Editorial Mc.Graw-Hill. ISBN 978-84-481-6111-8. EAN 9788448161118.

#### **Complementaria**

- Pilgrin, Mark: Inmersión en Python 3 [en línea] Traducido por José Miguel González Aguilera. Disponible gratuitamente (bajo licencia Creative Commons 3.0). [consultas: 07/03/19]  
<http://www.imgaguilera.com/libro/python/traducci%C3%B3n/latex/2016/08/19/inmersion-python.html>
- Marzal Varó, Andrés - Gracia Luengo, Isabel, García Sevilla, Pedro: Introducción a la programación con Python 3 [En línea] Disponible gratuitamente bajo licencia Creative Commons. [consultas: 07/03/19]  
<https://openlibra.com/es/book/introduccion-a-la-programacion-con-python-3>
- \*Van Rossum, Guido. El tutorial de Python. Traducido y empaquetado por la comunidad de Python Argentina. [consultas:31/10/18]. Español:  
<http://docs.python.org.ar/tutorial/pdfs/TutorialPython3.pdf>
- The Python Language Reference. [consultas 07/03/19]. Español:  
<https://docs.python.org/3/reference/>

#### **Autor:**

Ricardo Thompson, Patricia Mazzitelli, Felipe D'Aquino, Verónica Galati

#### **Actualización:**

Verónica Galati

## TRABAJO PRÁCTICO 6: FUNCIONES

Codifique funciones apropiadas para cumplir con los siguientes objetivos:

**Ejercicio 1:** Dados dos parámetros numéricos, calcular y devolver el resultado de la multiplicación de ambos utilizando sólo sumas.

**Ejercicio 2:** Dados dos parámetros enteros A y B, obtener AB (A elevado a la B) mediante multiplicaciones sucesivas, utilizando la función del ejercicio 1.

**Ejercicio 3:** Imprimir una columna de asteriscos, donde su altura se recibe como parámetro.

**Ejercicio 4:** Verificar si un número es par o impar, devolviendo True o False respectivamente.

**Ejercicio 5:** Devolver el máximo entre dos números recibidos como parámetros.

**Ejercicio 6:** Devolver True si el número entero recibido como primer parámetro es múltiplo del segundo, o False en caso contrario. Ejemplo: `esmultiplo(40,8)` devuelve True y `esmultiplo(50,3)` devuelve False.

**Ejercicio 7:** Dado un número entero, calcular su factorial. Ejemplo: `fact(4) = 4*3*2*1 = 24`.

**Ejercicio 8:** Calcular el Máximo Común Divisor de dos enteros no negativos, basándose en las siguientes fórmulas matemáticas:

- $MCD(X,X) = X$
- $MCD(X,Y) = MCD(Y,X)$
- Si  $X > Y \Rightarrow MCD(X,Y) = MCD(X-Y,Y)$ .

Ejemplo: `MCD(40,15)` devuelve 5.

**Ejercicio 9:** Desarrollar la función `signo(n)`, que reciba un número entero y devuelva un 1, -1 o 0 según el valor recibido sea positivo, negativo o nulo.

**Ejercicio 10:** Escribir la función `comparar(a,b)` que reciba como parámetros dos números enteros y devuelva 1 si el primero es mayor que el segundo, 0 si son iguales o -1 si el primero es menor que el segundo. Ejemplo: `comparar(4,2)` devuelve 1, y `comparar(2,4)` devuelve -1.

**Ejercicio 11:** Adaptar el programa que utiliza la fórmula de Newton para calcular la raíz cuadrada de un número positivo N (de la práctica anterior) para que trabaje como una función.

GUÍA DE EJERCICIOS

**Ejercicio 12:** Extraer un dígito de un número entero. La función recibe como parámetros dos números enteros, uno será del que se extraiga el dígito y el otro indica qué cifra se desea obtener. La cifra de la derecha se considera la número 0. Retornar el valor -1 si no existe el dígito solicitado. Ejemplo: `extraerdigito(12345,1)` devuelve 4, y `extraerdigito(12345,8)` devuelve -1.

**Ejercicio 13:** Devolver los últimos N dígitos de un número entero pasado como parámetro. El valor de N también debe ser pasado como parámetro. Devolver el número completo si N es demasiado grande. Ejemplo: `ultimosdigitos(12345,3)` devuelve 345, y `ultimosdigitos(12345,8)` devuelve 12345.

**Ejercicio 14:** Obtener el dígito central de un número entero pasado como parámetro, sólo si la cantidad de dígitos es impar. Si la longitud fuera par devolver -1. Ejemplo: `digitocentral(12345)` devuelve 3, y `digitocentral(123456)` devuelve -1.

**Ejercicio 15:** Escribir una función que reciba como parámetros dos números enteros y devuelva la concatenación de ambos. Por ejemplo `concatenar(123,456)` devuelve 123456.

**Ejercicio 16:** Realizar una función que calcule y devuelva la sumatoria de los términos de la sucesión de Fibonacci entre dos números de término dados, los que se reciben como parámetros.

GUÍA DE EJERCICIOS

TRABAJO PRÁCTICO 7: LISTAS

Desarrollar los siguientes programas o funciones, según corresponda:

**Ejercicio 1:** Escribir una función para ingresar desde el teclado una serie de números entre 1 y 20 y guardarlos en una lista. En caso de ingresar un valor fuera de rango el programa mostrará un mensaje de error y solicitará un nuevo número. Para finalizar la carga se deberá ingresar -1. La función no recibe ningún parámetro, y devuelve la lista cargada (o vacía, si el usuario no ingresó nada) como valor de retorno.

En los siguientes ejercicios utilice la función del ejercicio 1 para ingresar datos en una lista y:

**Ejercicio 2:** Calcular la suma de los números de una lista.

**Ejercicio 3:** Determinar si una lista es capicúa.

**Ejercicio 4:** Invertir aquellos valores ubicados en posiciones impares de una lista.

**Ejercicio 5:** Escribir una función para devolver la posición que ocupa un valor pasado como parámetro, utilizando búsqueda secuencial en una lista desordenada. La función debe devolver -1 si el elemento no se encuentra en la lista.

**Ejercicio 6:** Ídem anterior, utilizando búsqueda binaria sobre una lista ordenada.

**Ejercicio 7:** Una escuela necesita conocer cuántos alumnos cumplen años en cada mes del año, con el propósito de ofrecerles un agasajo especial en su día. Desarrollar un programa que lea el número de legajo y fecha de nacimiento (día, mes y año) de cada uno de los alumnos que concurren a dicha escuela. La carga finaliza con un número de legajo igual a -1. Emitir un informe donde aparezca -mes por mes- cuántos alumnos cumplen años a lo largo del año. Imprimir también una leyenda que indique cuál es el mes con mayor cantidad de cumpleaños.

**Ejercicio 8:** Rellenar una lista con números enteros entre 0 y 100 obtenidos al azar e imprimir el valor mínimo y el lugar que ocupa. Tener en cuenta que el mínimo puede estar repetido, en cuyo caso deberán mostrarse todas las posiciones que ocupe. La carga de datos termina cuando se obtenga un 0 como número al azar, el que no deberá cargarse en la lista.

**Ejercicio 9:** Dado una lista de N números (por ejemplo 5), devolver una lista de N-1 valores booleanos, tal que cada valor de este último arreglo corresponde al resultado de la comparación de los pares de valores consecutivos del primer arreglo. El valor booleano es True si el primer elemento del par es menor o igual que el siguiente, y False si no lo es. Ejemplo:

Lista original:

4	2	8	8	6
---	---	---	---	---

Resultado:

F	T	T	F
---	---	---	---

GUÍA DE EJERCICIOS

**Ejercicio 10:** Cargar dos listas de números A y B. Se solicita construir e imprimir tres nuevas listas que contengan:

- La concatenación de los valores pares de A con los impares de B.\*
- La concatenación de los valores pares de A con el reverso de los valores pares de B. ("valores pares" o "valores impares" se refiere a los elementos propiamente dichos y no a sus posiciones).
- La intercalación de los elementos de A y B.

**Ejercicio 11:** Dada una lista ordenada de números llamada A y un nuevo número N, desarrollar un programa que agregue el elemento N dentro de la lista A, respetando el ordenamiento existente. El programa deberá detectar automáticamente si el ordenamiento es ascendente o descendente antes de realizar la inserción.

**Ejercicio 12:** Leer una lista de números V. Luego se solicita:

- Calcular el producto de los elementos de subíndice par y dividirlo por la suma de los elementos de subíndice impar, sólo si esta suma es distinta de cero. Imprimir la lista leída y el resultado calculado, o un mensaje de error en caso de no poder realizar la operación.
- Generar e imprimir otra lista tal que su primer elemento contenga la suma del primero más el último elemento de la lista V; el segundo elemento contenga la suma del segundo más el penúltimo de V, etc. La nueva lista contendrá la mitad de los elementos de la lista original.
- Imprimir un listado de aquellos elementos de V que cumplan con la condición de tener iguales sus dos elementos laterales (el anterior y el siguiente). Si ninguno cumple esta condición, se imprimirá una leyenda aclaratoria. Considerar que los extremos de la lista se encuentran unidos, de modo que el último elemento se encuentra antes que el primero, y que el primer elemento se encuentra después del último.

**Ejercicio 13:** Realizar un programa que permita ingresar números en una lista, finalizando la lectura con -1. Informar si la secuencia de elementos ingresada es ascendente, descendente, todos sus valores son iguales o se encuentra desordenada.

**Ejercicio 14:** Eliminar de una lista de números enteros los valores que se encuentren en una segunda lista. Imprimir la lista original, la lista de valores a eliminar y la lista resultante.

**Ejercicio 15:** Leer dos listas de números M y N, ambas ordenadas de menor a mayor. Generar e imprimir una tercera lista que resulte de intercalar los elementos de M y N. La nueva lista también debe quedar ordenada, sin utilizar ningún método de ordenamiento.

GUÍA DE EJERCICIOS

**Ejercicio 16:** Desarrollar un programa que genere un número entero al azar de cuatro cifras y proponerle al usuario que lo descubra, ingresando valores repetidamente hasta hallarlo. En cada intento el programa mostrará mensajes indicando si el número ingresado es mayor o menor que el valor secreto. Permitir que el usuario abandone al ingresar -1. Informar la cantidad de intentos realizada al terminar el juego, haciendo que el usuario ingrese su nombre si mejoró la mejor marca de intentos obtenida hasta el momento. Luego mostrar la lista de los 5 mejores puntajes y preguntar si se desea jugar otra vez, reiniciando el juego en caso afirmativo

**Ejercicio 17:** Modificar el programa anterior para que las pistas brindadas por el programa no sean del tipo "es mayor" o "es menor" sino "M dígitos correctos y N dígitos aproximados". Se considera que un dígito es correcto cuando tanto su valor como su posición coinciden con los del número secreto, mientras que un dígito es aproximado cuando coincide el valor pero no su posición. Ejemplos:

Número secreto: 5739

- Intento 1: 1234 -> 1 correcto
- Intento 2: 5678 -> 1 correcto y 1 aproximado
- Intento 3: 9375 -> 4 aproximados

**Ejercicio 18:** Ingresar por teclado un número N y construir una lista llamada SECUENCIAS con N números enteros al azar entre 1 y 20. Esta lista se caracterizará porque sus valores deben encontrarse divididos en secuencias de números separadas por ceros, cuya suma no sea mayor que 20. Para eso se deberá agregar un elemento de valor 0 a fin de separar cada secuencia de la siguiente, cuidando que ninguna secuencia sume más de 20. Agregar un 0 adicional al final de la lista y mostrar la lista obtenida por pantalla. Ejemplo:

Valor de N ingresado: 11

Números al azar generados:

5, 2, 9, 6, 4, 15, 3, 19, 12, 1, 5

Lista SECUENCIAS construida:

5 2 9 0 6 4 0 15 3 0 19 0 12 1 5 0

**Ejercicio 19:** A partir de la lista SECUENCIAS generada en el ejercicio anterior, imprimir la secuencia más larga almacenada en la misma. Si hubiera varias secuencias con la misma longitud máxima deberán mostrarse todas las que correspondan.