# Ejercicios (muy) básicos Python

## Variables, operadores y condicionales

**Problema 1**

Se necesita una solución de software que le permita a un usuario dada su edad actual conocer su edad en el año 2070. Diseñe un algoritmo que satisfaga este requerimiento e impleméntelo en Python.

**Input**: Edad (entero >0), Año actual (entero >0)

**Output**: Edad en el 2070 (entero >0), Salida en pantalla con mensaje informativo

**Problema 2**

Se necesita una solución de software que le permita a un usuario conocer si un número es par o no. Diseñe un algoritmo que satisfaga este requerimiento e impleméntelo en Python.

**Input**: Número entero

**Output**: Bandera indicando si número es par o no (Bool), Salida en pantalla con mensaje informativo

**Problema 3**

Se necesita una solución de software que permita imprimir de una palabra el primer y último carácter. Diseñe un algoritmo que satisfaga este requerimiento e impleméntelo en Python.

**Input**: Palabra (String)

**Output**: Salida en pantalla con mensaje informativo sobre el primer y último carácter.

**Problema 4**

Se necesita una solución de software que permita calcular el descuento de un producto (el precio del producto es un dato ingresado por el usuario) según el día de la semana (dato ingresado por el usuario mediante carácter Lunes: l, Martes: m, Miércoles: M, Jueves: j, Viernes: v, Sábado: s, Domingo: d ) así:

DIA DESCUENTO

Lunes 10%

Martes 5%

Miércoles 3%

Jueves 1%

Viernes 7%

Sábado 0%

Domingo 1%

Diseñe un algoritmo que satisfaga este requerimiento e impleméntelo en Python.

**Input**: Precio (Real positivo), Día de la semana (string codificado como se enuncia).

**Output**: Descuento (Real positivo), Salida en pantalla con mensaje informativo

**Problema 5**

Se necesita una solución de software que haga un programa que en base al día del mes y día de la semana determine que placas tienen pico y placa. Diseñe un algoritmo que satisfaga este requerimiento e impleméntelo en Python.

**Input**: Día del mes (natural entre 1 y 31), Día de la semana (string codificado como Lunes: l, Martes: m, Miércoles: M, Jueves: j, Viernes: v, Sábado: s, Domingo: d).

**Output**: Salida en pantalla con mensaje informativo sobre las placas que tienen pico y placa (Las placas terminadas en 1,3,5,7,9 tiene pico y placa, o las placas terminadas en 0,2,4,6,8 tiene pico y placa).

## Ciclos

**Problema 6**

Se necesita una solución de software que permita identificar si una frase/palabra es palíndroma o no (frase/palabra que se lee igual de izquierda a derecha que de derecha a izquierda). Diseñe un algoritmo que satisfaga este requerimiento e impleméntelo en Python.

**Input**: Palabra (string)

**Output:** Bandera indicando si la palabra es palíndroma o no (Bool), Salida en pantalla con mensaje informativo.

**Problema 7**

En 2019 el país A tiene una población de 35 millones de habitantes y el país B de 19.9 millones. Las tasas de crecimiento anual de la población son de 2% y 3% respectivamente. Desarrollar un algoritmo para informar en que año la población del país B supera a la de A. Impleméntelo en Python.

**Output:** Año (entero) en el cual la población del país B supera a la de A.

**Problema 8**

Desarrolle una solución de software que permita al usuario ingresar un rango especificado por 2 números enteros, tal que el primero sea menor al segundo. Luego cuente el número de múltiplos de un numero entero X introducido por el usuario que existe en el rango. Por ejemplo, si se define el rango [2,21] y X=3 el número de múltiplos es 7, dado que 3; 6; 9; 12; 15; 18 y 21 son múltiplos de 3.

**Input**: Límite inferior (LI) del rango (entero >0) , Límite superior (LS) del rango (LS > LI), número X ( LI<=X<=LS).

**Output:** Bandera indicando si la palabra es palíndroma o no (Bool), Salida en pantalla con mensaje informativo.

**Condicionales:** Verifique e informe al usuario si el rango es valido y si X están en el rango.

**Problema 9**

Desarrolle una solución de software que permita generar la siguiente serie de números. N es un número ingresado por el usuario. Diseñe un algoritmo que satisfaga este requerimiento e impleméntelo en Python.

1

22

333

4444

55555

.

.

.

N veces N

**Input**: N (entero > 0).

**Output:** Serie, Salida en pantalla con serie generada.

**Problema 10**

Juego Adivinando el número. Se necesita desarrollar un juego donde un usuario debe adivinar un número entero positivo entre 1 y 1000 que solo es conocido por el computador. Para cada intento de adivinar por parte del usuario el juego le podrá dar dos indicaciones:

1. Que el número ingresado es menor al número a adivinar.

2. Que el número ingresado es mayor al número a adivinar.

El juego termina cuando el usuario adivina el número.

Diseñe un algoritmo que satisfaga este requerimiento e impleméntelo en Python.

**Input**: Intento de adivinanza por parte del usuario (enteros >0).

**Output:** Mensajes de indicación del juego.

## Funciones y Recursividad

**Problema 11**

Se requiere una solución de software que genere los primeros n números de la serie de Fibonacci. Identifique los inputs y outputs del problema, diseñe un algoritmo que satisfaga la definición de la función e impleméntelo en Python mediante funciones.

**Problema 12**

Se requiere una solución de software que implemente la siguiente función matemática:

Identifique los inputs y outputs del problema, diseñe un algoritmo que satisfaga la definición de la función e impleméntelo en Python.

**Problema 13**

Se requiere una calculadora que implemente las funciones básicas de una calculadora (suma, resta, multiplicación, módulo, potencia y máximo común divisor). La calculadora debe desplegar un menú para las operaciones indicadas. La calculadora le dará la opción al usuario de escoger la operación que desea realizar, luego le pedirá los dos números NATURALES a operar. Si alguno de los números ingresados nos es un número natural debe solicitarlos nuevamente. Diseñe un algoritmo que satisfaga este requerimiento e impleméntelo en Python.

Identifique los inputs y outputs del problema, diseñe las funciones que satisfagan las operaciones e impleméntelo en Python.

**Problema 14**

La princesa Dido tuvo que huir de Roma, perseguida por su propio hermano, y terminó por refugiarse en la costa norte de África, donde le pidió a un rey local que le cediera un terreno en el cual ella pudiera vivir. El rey, con la oportunidad de humillar a una princesa extranjera, accedió a entregarle el terreno que ella deseara, siempre y cuando no fuera mayor a lo abarcado por la piel de un buey. La princesa accedió a la extraña propuesta, y el rey le hizo entregar la piel de buey, ofreciéndose acompañarla para oficializar la entrega del terreno.

Cuál sería su sorpresa cuando la princesa tomó la piel y la cortó en trozos de un milímetro de ancho, formando una larga cuerda con la que trazó un círculo gigantesco. Después del curioso trato, el rey no tuvo otra opción más que entregarle el terreno, y en el Dido fundó la ciudad de Cartago.

Descripción de la Entrada: Su función debe quedarse esperando las dimensiones (largo y ancho, en metros) de la piel de buey –que vamos a asumir cuadrada. Tenga en cuenta que los números de entrada pueden tener decimales, en cuyo caso el separador de decimales será el punto (“.”).

Descripción De La Salida: La salida en consola será el área total, en kilómetros cuadrados que puede abarcar Dido con la piel de buey ingresada, asumiendo que la divide en tiras de un milímetro de grosor para formar una cuerda, y que con esa cuerda forma un círculo perfecto. Sólo es necesario un número, no es necesario que especifique que se trata de kilómetros cuadrados.

**Problema 15**

Construya una función que lea el contenido textual de un archivo de texto, cuente la cantidad de palabras, caracteres, sentencias que existen y las imprima en un archivo nuevo. El nombre del nuevo archivos será {nombreArchivoOriginal}\_salida.txt.