

## ACTIVIDAD 1: APLICACIÓN PYTHON

### 1. Presentación

Se trata de una aplicación individual en la que los alumnos deberán realizar un conjunto de funciones y procedimientos que conforman una calculadora personalizada y tendrán que resolver una serie de operaciones proporcionadas en un fichero que tendrá que ser abierto y gestionado por la aplicación resultante.

El objetivo principal es adquirir competencias en el uso del lenguaje de programación Python, enfrentándose a un problema complejo y solucionarlo con la aplicación de las principales técnicas de programación, como el uso de funciones, la lectura y procesamiento de ficheros, y la visualización y presentación de resultados.

### 2. Aplicación Python

La aplicación que se pide desarrollar en esta actividad es una calculadora personalizada.

La calculadora debe realizar 12 operaciones personalizadas, con 3 operandos cada una, con una complejidad creciente.

Las operaciones se definen más adelante, y se requiere la generación de, al menos, 12 funciones, una para cada una de las funciones.

Además, se valorará el uso de funciones auxiliares para operaciones que se repitan en más de una operación.

Adicionalmente, se proporciona un fichero (*operaciones\_calculadora\_personalizada.csv*) con 60 operaciones que emplean las operaciones personalizadas en el siguiente formato:

ID	nombre_operador	operando1	operando2	operando3	resultado_esperado
----	-----------------	-----------	-----------	-----------	--------------------

El resultado esperado en la actividad es un script de Python que cargue las funciones necesarias para realizar las operaciones, cargue el fichero *operaciones\_calculadora\_personalizada.csv* y recorra cada una de las operaciones, calcule el resultado, lo compare con el esperado, y decida si éste es correcto o no, guardando el resultado en un fichero que contendrá 60 líneas con los respectivos resultados y que se llamará *resultados.txt*. Además, se deberá realizar una validación de los tipos de los operandos, mostrando un mensaje "VALOR INVÁLIDO" y el resultado = 0

Por ejemplo, para las líneas:

```
1,sumaResta,5,3,2,6
2,sumaResta,3,4,8,1
3,sumaResta,"a",5,5
```

El formato para la salida por pantalla de estas líneas, debería ser:

```
Operación 1 - sumaResta(5,3,2) - resultado: 6 - CORRECTO
Operación 2 - sumaResta(3,4,8) - resultado: -1 - INCORRECTO
Operación 3 - sumaResta("a",5,5) - resultado: 0 - VALOR INVÁLIDO
```

### 2.1. Operaciones

Las operaciones que hay que definir en la aplicación:

- sumaResta
- multiPot
- triangulos
- sectores
- fibo
- ecuaciones2
- distancias
- areaCartesiana
- acortar
- insertar
- barajar
- invertir

sumaResta

Esta operación es la más sencilla. Consiste en sumar el primer operando al segundo y restarle el tercero:

$$\text{sumaResta}(a,b,c) = a + b - c$$

Por ejemplo:

$$\text{sumaResta}(5,3,2) = 5 + 3 - 2 = 6$$

### multiPot

Se multiplican el operando 1 y el operando 2 y se eleva al operando 3:

$$\text{multipot}(a,b,c) = (a \times b)^c$$

Por ejemplo:

$$\text{multiPot}(1,3,2) = (1 \times 3)^2 = 3^2 = 9$$

### triangulos

Esta operación calcula el área total de varios triángulos iguales. El primer operando es la base de los triángulos, el segundo es la altura, y el tercero es el número de triángulos. El resultado de la operación estará truncado a 2 decimales, redondeando hacia arriba.

El área de un triángulo se calcula mediante la fórmula siguiente, siendo a la base del triángulo y b su altura:

$$\text{areaTriangulo}(a,b) = \left( \frac{a \times b}{2} \right)$$

$$\text{triangulos}(a,b,c) = \text{areaTriangulo}(a,b) \times c$$

Por ejemplo:

$$\text{triangulos}(2,3,5) = (2 \times 3 / 2) \times 5 = 3 \times 5 = 15$$

### sectores

Esta operación calcula el área de varios sectores circulares iguales. El primer operando es el radio del círculo base, el segundo operando es el número de sectores iguales en el que deberíamos dividir el

círculo original, y el tercer operando es el número de sectores a sumar. Por defecto, el resultado vendrá con sólo 2 decimales, redondeando hacia arriba.

$$\text{sectores}(a, b, c) = \left( \frac{\pi \times a^2}{b} \right) \times c$$

Por ejemplo:

$$\text{sectores}(2, 3, 5) = (3.14 \times 2^2 / 3) \times 5 = (12.57 / 3) \times 5 = 4.18 \times 5 = 20.94$$

### fibo

Esta operación es una personalización de la secuencia de Fibonacci. Si esta secuencia que se ha visto durante el curso, comienza en (0,1) y los siguientes valores son el resultado de la suma de los 2 últimos, en la operación fibo se darán los dos primeros números de la secuencia (operando 1 y 2) y el tercer operando significa el orden en la secuencia que sigue. El resultado será el elemento enésimo de la secuencia definida, tomando como primer elemento (n=1) el elemento definido en el primer operando.

$$\text{fibo}(a, b, c) = \underbrace{a, b}_{\leftarrow} \underbrace{a + b, \dots}_{\rightarrow} \text{resultado}$$

**c**

Por ejemplo:

$$\text{fibo}(2, 3, 5) = 2, 3, 5, 8, 13 = 13$$

### ecuaciones2

En este operador resolveremos ecuaciones de segundo grado donde cada operador será el coeficiente de cada grado de la ecuación, siendo a el coeficiente de grado 2, b el de grado 1 y c el de grado 0. El resultado de la operación estará truncado a 2 decimales, redondeando hacia arriba.

La ecuación queda así:

$$ax^2 + bx + c = 0$$

La resolución de la ecuación se realiza con la siguiente operación, donde nos quedaremos con la solución positiva de la ecuación:

$$\text{ecuaciones2}(a, b, c) = \frac{-b + \sqrt{b^2 - 4ac}}{2a}$$

Por ejemplo:

$$ecuaciones2(1, 3, 2) = \frac{-3 + \sqrt{3^2 - 4 \times 1 \times 2}}{2 \times 1} = \frac{-3 + \sqrt{9 - 8}}{2} = \frac{-3 + 1}{2} = -1$$

### distancias

En esta operación recibiremos los operadores como un punto en un espacio bidimensional. Es decir, una lista con 2 valores, positivos o negativos, del eje de las x, el primero, y el de las y, para el segundo. La operación consiste en calcular la distancia cartesiana entre el punto del primer operador y el segundo, y sumarle la distancia entre el punto del segundo operador y el del tercero. El resultado de la operación estará truncado a 2 decimales, redondeando hacia arriba.

La fórmula para calcular la distancia entre dos puntos es:

$$distancia(a, b) = \sqrt{(x_b - x_a)^2 + (y_b - y_a)^2}$$

$$distancias(a, b, c) = distancia(a, b) + distancia(b, c)$$

Por ejemplo:

$$distancias((1,2),(-2,3),(3,-1)) = \sqrt{(-2-1)^2 + (3-2)^2} + \sqrt{(3+2)^2 + (-1-3)^2} = \sqrt{9+1} + \sqrt{25+16} = \sqrt{10} + \sqrt{41} = 9.57$$

### areaCartesiana

La operación consiste en devolver el área de un triángulo en coordenadas cartesianas. En esta operación recibiremos los operadores como un punto en un espacio bidimensional. Es decir, una lista con 2 valores, positivos o negativos, del eje de las x, el primero, y el de las y, para el segundo. El resultado de la operación estará truncado a 2 decimales, redondeando hacia arriba.

El área de un triángulo se calcula utilizando la Fórmula de Herón, que utiliza el valor de los lados del triángulo, y la fórmula del semiperímetro:

$$areaHeron(sp, ladoA, ladoB, ladoC) = \sqrt{sp \times (sp - ladoA) \times (sp - ladoB) \times (sp - ladoC)}$$

Y la fórmula del semiperímetro es:

$$sp(ladoA, ladoB, ladoC) = \frac{ladoA + ladoB + ladoC}{2}$$

Los lados los calcularemos como la distancia entre los puntos definidos como vértices, cogidos dos a dos. De esta forma el ladoA será la distancia entre vértices A-B, el ladoB la distancia B-C y el ladoC será la distancia C-A. La fórmula para calcular la distancia entre dos puntos es:

$$\text{distancia}(a,b) = \sqrt{(x_b - x_a)^2 + (y_b - y_a)^2}$$

$$\text{areaCartesiana}(a,b,c) = \text{areaHeron}(sp, \text{distancia}(a,b), \text{distancia}(b,c), \text{distancia}(c,a))$$

Por ejemplo:

$$\text{areaCartesiana}((2,0),(-3,5),(3,2)) = \text{areaHeron}(sp, \text{ladoA}, \text{ladoB}, \text{ladoC}) = 7.5$$

$$sp = 8.01$$

$$\text{ladoA} = 7.07$$

$$\text{ladoB} = 6.71$$

$$\text{lado C} = 2.24$$

### acortar

Pasamos a las operaciones con cadenas o strings de texto. Acortar devolverá el string que se escribe en el primer operando (escrito entre comillas) acortado en un número de caracteres por delante, y otro número por detrás, definidos por el operando 2 y 3 respectivamente.

Por ejemplo:

$$\text{acortar}(\text{"Esto es una frase"}, 5, 6) = \text{"es una"}$$

### insertar

Tanto el operando 1 como el 2 contendrán información en forma de cadena de texto. El tercer operando será un valor numérico que indica la posición de la primera cadena a partir de la cual se insertará la segunda.

Por ejemplo:

$$\text{insertar}(\text{"Esto es una frase"}, \text{"gran "}, 12) = \text{"Esto es una gran frase"}$$

### barajar

Tanto el operando 1 como el 2 contendrán información en forma de cadena de texto. El tercer operando será un valor numérico que indica cada cuantos caracteres de la primera cadena se inserta 1 carácter de la segunda. Si la primera cadena se termina antes de introducir todos los caracteres de la segunda, se concatenarán detrás de la primera los caracteres restantes de la segunda. En el caso contrario, simplemente, se escribirán los restantes caracteres de la primera cadena sin insertar ningún carácter extra.

Por ejemplo:

`barajar("Esto es una frase", "gran" , 2) = "Esgtor eas nuna frase"`

`barajar("gran""Esto es una frase", 1) = "gErsatno es una frase"`

### invertir

El primer operando 1 será una cadena de texto. El segundo será un valor categórico que indica el modo de la operación, con dos valores posibles: "completo" y "palabra". El tercer operando será un valor numérico que indica cada cuantos caracteres se realizará la inversión.

El modo completo consiste en invertir toda la cadena, estableciendo el número de caracteres de inversión. Si tenemos "123456" y 3 caracteres de inversión, contaremos 3 desde el final hacia el inicio y seleccionaremos los 3 caracteres, en este caso "456" y los pondremos al principio del resultado; después seleccionamos los tres siguientes hacia el principio ("123") y los colocamos a continuación, quedando como resultado: "456123".

El modo palabra hace la inversión por palabras, considerando una palabra lo que separa un espacio en blanco (" "). El procedimiento es similar al completo, teniendo en consideración la palabra como si fuera toda la cadena de texto. El resultado serán las palabras invertidas, manteniendo el orden de aparición de las palabras, y separadas por espacios (" "). En la cadena "1234 5678 9012", considerando con el número de caracteres de inversión es 2, se detectarían 3 palabras: "1234", "5678" y "9012" y se invertirían individualmente: "3412", "7856" y "1290" para, después, devolver el resultado separadas por espacios: "3412 7856 1290".

Por ejemplo:

`invertir("Esto es una frase", "completo" , 2) = "sera fna uesto stE"`

`invertir("Esto es una frase", "palabra" , 1) = "otsE se anu esarf"`

### 2.2. Validaciones

Las operaciones recibidas en el fichero '*operaciones\_calculadora\_personalizada.csv*' pueden no estar correctamente definidas.

Por ello es necesario realizar una comprobación de que los operandos recibidos tienen el formato definido en la operación.

Los tipos posibles para esta aplicación son:

- Entero: Son números enteros, positivos o negativos.
- Punto: Es un array de 2 elementos enteros, positivos o negativos, que representan un punto en el espacio cartesiano. El primer valor es el eje de las X y el segundo, el de las Y. En el fichero aparecerán como una cadena de texto, entre comillas, entre paréntesis, y separados por comas.
- String: Es una cadena de caracteres o string. En el fichero aparecerán entre comillas.
- Modo: Es una cadena de caracteres o string que tiene información categórica. Sólo puede contener sólo 2 valores distintos: "completo" y "palabra".

En la siguiente tabla se muestran los tipos de cada operando para cada operación:

Operación	Operando 1	Operando 2	Operando 3
sumaResta	Entero	Entero	Entero
multiPot	Entero	Entero	Entero
triangulos	Entero	Entero	Entero
sectores	Entero	Entero	Entero
fibo	Entero	Entero	Entero
ecuaciones2	Entero	Entero	Entero
distancias	Punto	Punto	Punto
areaCartesiana	Punto	Punto	Punto
acortar	String	Entero	Entero
insertar	String	String	Entero
barajar	String	String	Entero



invertir	String	Modo	Entero
----------	--------	------	--------

### 2.3. Consideraciones sobre la Aplicación

Tal y como se ha comentado anteriormente, se recomienda el uso de funciones y procedimientos para desarrollar el programa.

En una primera lectura del enunciado, aparentemente aparecen 12 funciones que se corresponden con las operaciones que hay que desarrollar.

Es recomendable plantear el problema de manera completa antes de comenzar a desarrollarlo en Python.

Una buena práctica es definir la solución en pseudo-código (no es necesario que sea muy estricto en la notación, pero se debe entender) y un diagrama de cómo funciona la solución.

Otra buena práctica es no duplicar código. Hay que detectar aquellos procesos que se repiten en el programa, aislarlos, convertirlos en una función o procedimiento y utilizarlo cuando toque. Es una manera más clara de generar aplicaciones profesionales que se puedan integrar y desarrollar por otros futuros desarrolladores.

Por último, la clave de todos los programas es que tengan una buena documentación.

Además de la documentación generada en la fase previa a la codificación del proyecto, es necesario realizar comentarios y utilizar la documentación dentro del código del lenguaje de programación.

### Preguntas a responder

Los entregables a desarrollar en esta actividad son:

- Documentar a través de un documento Word, la solución propuesta:
  - Describir el funcionamiento de la solución
  - Dibujar un diagrama con los procesos, funciones y procedimientos que intervienen.
- Documentar a través de un documento Word la solución propuesta en pseudo-código.
- Código de la Aplicación
- Fichero *'resultados.txt'* resultante del procesado del fichero *'operaciones\_calculadora\_personalizada.csv'*