

Turno 5

Preparación y consignas generales:

- Para comenzar, cree un proyecto en PyCharm y dentro del mismo cree un archivo **.py** cuyo nombre tenga el formato **legajo-apellido** (donde "legajo" es su número de legajo y "apellido" es su apellido).
- En la misma carpeta de ese proyecto, descargue y guarde el archivo **"entrada.txt"** que se provee con este enunciado.
- Se le pedirá que procese una cadena de caracteres cuya carga debe hacerse **obligatoriamente** desde el archivo de texto **"entrada.txt"** según técnicas que fueron explicadas en fichas y en clases prácticas.
- El texto que cargue desde ese archivo finaliza con "." y cada palabra de ese texto está separada de las demás por un (y solo un) espacio en blanco. No hay saltos de línea en el archivo.
- El programa **debe** incluir una función principal para lanzar el programa desde el script principal.
- El programa **debe** tener control de ejecución del script principal con la variable **__name__**.
- El programa **debe** tener **al menos** una función simple desarrollada por el estudiante con parámetros y con retorno de resultados.
- El programa **debe** procesar el texto carácter a carácter (a razón de uno por vuelta de ciclo, **con un único ciclo** para todo el proceso).
- El programa que entreguen **NO debe** usar un menú de opciones ni ningún tipo de carga por teclado en ninguna parte del programa por ninguna razón. El texto **debe** ser levantado estrictamente desde el archivo **"entrada.txt"**.
- La secuencia y el formato de las instrucciones de salida por pantalla de su programa, debe ser **obligatoriamente y tal cual** la que se indica a continuación. **No cambie los mensajes, no cambie los caracteres en cada mensaje, no cambie la forma de cada mensaje, ni cambie el orden de cada print():**

```
print("Primer resultado:", r1)
print("Segundo resultado:", r2)
print("Tercer resultado:", r3)
print("Cuarto resultado:", r4)
```
- Copie y pegue las cuatro instrucciones anteriores **tal como están** al final de su función principal. No agregue **ninguna** otra llamada a **print()** en **ninguna** parte de su programa. Guarde los resultados que vaya calculando en las variables r1 (para el primer resultado pedido), y en r2, r3 y r4 respectivamente (para los resultados que siguen).

Enunciado:

Se pide desarrollar un programa en Python que permita procesar un texto completo contenido en una variable de tipo cadena de caracteres (cargado desde el archivo "entrada.txt" de acuerdo a todo lo expresado en la sección anterior), que haga lo siguiente:

- Determinar la cantidad de palabras de más de dos caracteres, que tienen letras (minúsculas o mayúsculas) en las primeras dos posiciones, y luego solo siguen caracteres que representen dígitos pares. Por ejemplo, en el texto: "Los cuadrantes **AC284** y DE983 son peligrosos pero el sector 8U248 es seguro." hay solo una palabra que cumple: "**AC284**". La palabra "DE983" tiene un dígito impar y no cumple. La palabra "8U248" tampoco cumple porque uno de sus dos primeros caracteres es un dígito. Y las palabras "el" y "es" no son válidas porque tienen menos de tres caracteres (aunque los dos que tienen son letras, no tienen ningún dígito).
- Determinar la longitud (en cantidad de caracteres) de la palabra más corta entre aquellas que tienen más de tres caracteres de largo. Por ejemplo, en el texto: "Por fin fuimos campeones del **mundo**." la menor longitud entre las palabras que tienen más de tres caracteres es 5 (en la palabra "**mundo**").

3. Determinar el **porcentaje entero** (respecto del total de palabras del texto) de las palabras que tienen más vocales que consonantes. Por ejemplo, en el texto "**Ahora esa tarea** resulta imposible y torpe." hay tres palabras que cumplen: "**Ahora**", "**esa**" y "**tarea**". Como hay siete palabras en total en el texto, el **porcentaje entero** pedido es del 42 por ciento (aclaración: en el cálculo del porcentaje haga primero la multiplicación y luego la división).
4. Determinar cuántas palabras incluyen la expresión "tu" (con cualquiera de sus letras en minúscula o mayúscula) pero de tal forma que la palabra no tenga además ninguna "b" (minúscula o mayúscula) ni ningún dígito. Por ejemplo, en el texto: "Los **tucumanos** tumbaron al ritmo de 2tu y quedaron en el atium." hay una palabra que cumple: "**tucumanos**". La palabra "tumbaron" no cuenta porque si bien tiene la expresión "tu", tiene también una "b". La palabra "2tu" tampoco cuenta porque si bien tiene "tu", también tiene un dígito. Y "atium" no cuenta porque no tiene "tu".

Criterios generales de evaluación.

- Planteo sin carga por teclado en ninguna parte: 0 puntos (**reprueba si no cumple**).
- Instrucciones de salida tal cual se indicó: 0 puntos (**reprueba si no cumple**).
- Nombre del archivo fuente correcto: 0 puntos (-1 si no cumple).
- Apertura correcta del archivo "entrada.txt": 0 puntos (-1 si no cumple).
- Planteo en base a un único ciclo: máximo 0 puntos (-2 si no cumple).
- Inclusión correcta de una función principal: **máximo 2 puntos**.
- Inclusión correcta de al menos una función con parámetros y retorno: **máximo 2 puntos**.
- Control correcto de ejecución del script principal: **máximo 1 punto**.
- Resultado correcto del ítem 1: **máximo: 3 puntos**.
- Resultado correcto del ítem 2: **máximo: 4 puntos**.
- Resultado correcto del ítem 3: **máximo: 5 puntos**.
- Resultado correcto del ítem 4: **máximo 6 puntos**.
- Para aprobar el parcial, el estudiante debe llegar a un mínimo de alrededor de 12 puntos (un porcentaje de al menos 55% del puntaje máximo de 22 que puede ser alcanzado).