

Turno 4

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. Mientras desarrolle su programa, puede obviamente hacer pruebas con cadenas asignadas en forma fija, pero **asegúrese** que al entregar su trabajo, la carga de la cadena a procesar se haga **obligatoriamente** desde el archivo indicado.
- 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 caracter a caracter (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 (en todos los casos en los que se refieran letras, debe entenderse que nos referimos a letras del idioma español, en minúsculas o en mayúsculas, y acentuadas o no acentuadas si se trata de vocales):

- Determinar la cantidad de palabras que tienen más dígitos que vocales, pero solo una consonante. Por ejemplo, en el texto: **"R23A** es un robot similar a R2D2 y **C256** pero muy distinto de U57E3." hay dos palabras que cumplen: **"R23A"** y **"C256"**.
- Determinar la longitud de la palabra más corta entre aquellas que tienen al menos una "s" o al menos una "c" (minúsculas o mayúsculas). Por ejemplo, en el texto: **"Antes de anclar cae el oca**so." Hay cuatro

palabras que cumplen ("Antes", "anclar", "cae" y "ocaso"), y la menor longitud entre esas palabras es de 3 caracteres (en la palabra "cae").

3. Determinar el porcentaje entero que representan las palabras que no tienen dígitos sobre el total del palabras del texto. Por ejemplo, en el texto "Estamos en la zona X3 o en la T5." hay 9 palabras en total, y 7 de ellas no tienen dígitos. Por lo tanto, el porcentaje entero pedido es del 77 por ciento. Aclaración: al calcular el porcentaje, haga primero la multiplicación y luego la división.
4. Determinar cuántas palabras empiezan con la expresión "vi" (con cualquiera de sus letras en minúscula o mayúscula) y tengan además una "n" en cualquier lugar. Por ejemplo, en el texto: "Avieso el viernes envia su mensaje a AVEIT." hay una palabra que cumple: "viernes".

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).