



1. Problemas a resolver

En la primera parte de esta tarea se presentan dos problemas en donde deberás aplicar los contenidos vistos en el módulo 3 del curso. Estos problemas te servirán de práctica para desarrollar la segunda parte de esa tarea que es compuesta un ejercicio, cual debe ser entregado en buzón habilitado en Canvas hasta el 10 de mayo de 2018 a las 23:59 hrs. Sin embargo, es muy importante que logres resolver **todos** los ejercicios para prepararte bien para el examen final.

1.1. Problema 1

Escriba una función que reciba un string con una operación matemática básica (+, −, *, /) y devuelve el resultado de esa operación. Por ejemplo si recibe "4.5 + 35" debe devolver 39,5.

1.2. Problema 2

Escriba la función `secuencia_creciente_maxima()`, la cual recibe un arreglo de enteros y retorna la cantidad de elementos que componen la máxima secuencia de números donde se cumple $a_i < a_{i+1}$ para todo elemento de la secuencia. Por ejemplo si la función recibe el arreglo 4, 2, 3, 4, 7, 9, 11, 5, 2, 3, 4, 1, 2, 7, 5, 11, 13 debe retornar 6, ya que la secuencia más larga donde se cumple lo indicado es la formada por los números 2, 3, 4, 7, 9, 11.

1.3. Problema 3

Escribe una función `imprimir_matriz(matriz)` que reciba una lista de listas (parámetro `matriz`) con la representación de una matriz de m filas por n columnas, e imprima la matriz completa en consola, mostrando los elementos de cada fila separados por un espacio. Al imprimir, se debe incluir una fila de encabezado con los números de las columnas, y una columna que muestre los números de fila.

Por ejemplo, la siguiente lista:

```
[[1,2,3,4,5], [6,7,8,9,10], [11,12,13,14,15]]
```

Debiera imprimirse de la siguiente manera:

```
1 2 3 4 5
1: 1 2 3 4 5
2: 6 7 8 9 10
3: 11 12 13 14 15
```

Hints: La función `print` se puede usar en Python 3 de manera que no ponga un salto de línea al final de la impresión, de la siguiente forma:

```
print("1 2 3 4 5", end="")
```

El segundo parámetro `end` recibe el caracter que se desea imprimir al final de la línea. Si se pone un string vacío, `print` no imprime ningún caracter en esta posición.

Otra manera de imprimir la matriz es crear un string inicialmente vacío, y luego concatenarle todos los caracteres de cada fila (usar operador + con strings), y concatenar `n` para poner saltos de línea entre filas de la matriz.

1.4. Problema 4 (Evaluado)

Un histograma es una representación gráfica de las frecuencias de ciertos datos, distribuidos en alguna cantidad de categorías. Por ejemplo para los siguientes datos:

0, 2, 4, 5, 3, 12, 1, 4, 5, 121, 78, 12, 56, 1, 7, 9, 12, 13, 25, 24, 28, 6, 65, 91, 14, 75, 10

Podríamos definir las categorías $x \leq 20$, $20 < x \leq 50$ y $50 < x$, con lo que se produce el siguiente histograma:

X	Frecuencia
$x \leq 20$	18
$20 < x \leq 50$	3
$50 < x$	6

Usted cuenta con un archivo de texto llamado `precios_modelos.txt`, que consiste en una lista de pares `modelo - precio` (un par por línea) de los cuales `modelo` es el nombre de un modelo de automóvil y `precio` es el precio al que fue vendido por una automotora de automóviles usados. Tenga en cuenta que pueden existir distintos registros para un mismo modelo, y el precio no es necesariamente el mismo.

A continuación se muestra un ejemplo del archivo `precios_modelos.txt`:

```
Morning 4000000
Defender 12000000
Swift 5400000
Morning 3890000
Swift 4100000
Defender 15250000
```

Se le pide implementar un programa que le ofrezca al usuario un menú con todos los modelos que contiene el documento y le permita seleccionar uno de ellos. Cuando el usuario seleccione un modelo, el programa debe mostrar un histograma de los precios del modelo seleccionado. Además, su programa debe imprimir el histograma en un archivo llamado `histograma.txt`.

Para determinar las categorías del histograma debe encontrar el menor y el mayor de los precios a los que el modelo fue vendido, y luego separar el intervalo `menor precio - mayor precio` en 5 sub-intervalos del mismo tamaño.

Todos los modelos en el archivo tienen al menos 2 valores de compra distintos entre sí.

HINT: puede usar las funciones `min(nombre_de_lista)` y `max(nombre_de_lista)` para encontrar el valor mínimo y máximo en una lista.

A continuación se muestra un ejemplo del programa en funcionamiento:

```
(1) Baleno
(2) Swift
(3) Yaris
(4) Morning
(5) Defender
(6) Rio
(7) Rubicon
(8) Amarok
(9) Accent
```

```
Seleccione un modelo: 3
```

6381156 - 6654116 | xxxx
6654116 - 6927076 | xxxx
6927076 - 7200036 | xxxxx
7200036 - 7472996 | xxxxxxxx
7472996 - 7745956 | xxxx

1.5. Rúbrica de Puntaje

La siguiente rúbrica de puntaje solamente aplica al ejercicio final, el cual es el único que se considera en la evaluación de la tarea. La nota se obtiene sumando el punto base al puntaje obtenido.

- [1 punto] Abrir y leer correctamente el archivo `precios_modelos.txt`.
- [0.5 puntos] Para cada línea del archivo, extraer el nombre del modelo y el precio de venta.
- [1 punto] Se le muestra correctamente al usuario el menú de modelos, los modelos no deben repetirse.
- [0.5 puntos] Se obtienen todos los precios del modelo seleccionado.
- [0.5 puntos] Se calculan los 5 intervalos para el histograma.
- [1.5 puntos] Se calcula la frecuencia de cada intervalo correctamente.
- [1 punto] Se escribe el resultado en el archivo `histograma.txt`.

Se permiten puntajes parciales de acuerdo al logro en cada ítem $\times 0,25$, $\times 0,5$, $\times 0,75$.

2. Evaluación y Entrega

Fecha de entrega: 10 de mayo de 2018 hasta las 23:59, según la hora del sistema canvasUANDES, oficial en toda la universidad. Por ningún motivo se aceptarán tareas atrasadas.

Formato de entrega: Subir **un solo archivo comprimido en formato zip** que contenga el programa Python del problema evaluado (archivo `p4.py`), al buzón en la página de la tarea en el sitio web del curso en canvasUANDES <https://uandes.instructures.com>. El nombre de archivo debe ser “TareaM2-RUT.zip”, reemplazando “RUT” por su número de RUT sin puntos ni guión. Por ejemplo, si su RUT es 18.123.122-K, el nombre de archivo debe ser “TareaM2-18123122k.zip”.

Aquellas tareas que no funcionen correctamente a tal punto que no sea posible ejecutarlas, o produzcan errores que impidan la ejecución, serán calificadas con nota máxima 3,9.

Debe asegurarse de que su tarea sea descargable de forma íntegra desde canvasUANDES (el archivo `.zip` con el `p4.py` en su interior).

3. Consideraciones

El trabajo en las tareas es estrictamente individual, por lo que cuide su tarea para que no sea copiada **parcial o íntegramente** por otros. Cualquier copia será penalizada, recibiendo el mismo castigo tanto quien copia como quien permite que le copien. También es considerada copia cualquier ayuda externa recibida directamente en la tarea, sin importar si proviene de un alumno del curso, de la universidad, o de otro lugar. El castigo será establecido por el Consejo de la Facultad, siendo como mínimo un 1,0 de promedio en el curso.