

Fundamentos de Programación

Curso 2024-25 - Primer examen parcial, 13 de enero de 2025

MotoFP

Contexto

Se ha organizado un nuevo mundial de motociclismo llamado MotoFP. Se dispone de un archivo en formato CSV codificado en UTF-8 con información de los resultados de todas las carreras. En cada línea del archivo se recoge la siguiente información: la fecha y hora de la carrera, nombre del circuito, país, si el asfalto estuvo mojado en algún momento de la carrera ("Mojado") o no ("Seco"), la duración de la vuelta rápida en segundos y los tres nombres de pilotos y escuderías en el podio al final de la carrera (es decir, los pilotos y escuderías que quedaron en primer, segundo y tercer lugar). Las primeras líneas son las que se muestran a continuación:

```
Fecha y Hora,Circuito,País,Asfalto,Tiempo de la Vuelta Rápida,Nombre Ganador,Marca Ganador,Nombre  
Segundo Clasificado,Marca Segundo Clasificado,Nombre Tercer Clasificado,Marca Tercer Clasificado  
2023-03-01 14:00,Le Mans,Francia,Seco,114.938,Francesco Bagnaia,Ducati,Johann Zarco,Honda,Fabio  
Quartararo,Yamaha  
2023-03-12 17:00,Motegi,Japón,Seco,101.52,Fabio Quartararo,Yamaha,Brad Binder,KTM,Francesco  
Bagnaia,Ducati
```

Utilice obligatoriamente las `NamedTuple` **CarreraFP** y **Piloto** que se definen a continuación:

```
from typing import NamedTuple
```

```
Piloto=NamedTuple("Piloto", [("nombre", str),("escuderia", str)])
```

```
CarreraFP=NamedTuple("CarreraFP",[  
    ("fecha_hora",datetime),  
    ("circuito",str),  
    ("pais",str),  
    ("seco",bool), # True si el asfalto estuvo seco, False si estuvo mojado  
    ("tiempo",float),  
    ("podio", list[Piloto])])
```

Observaciones

- Los datos en el csv no están ordenados cronológicamente ni con ningún otro criterio.
- La cadena de formato para parsear el campo `fecha_hora` es `"%Y-%m-%d H:M"`.
- El número de días entre dos fechas, `fecha_inicio` y `fecha_fin`, se calcula como `(fecha_fin - fecha_inicio).days`.

Estructura de las carpetas del proyecto

- **/src:** Contiene los diferentes módulos de Python que conforman el proyecto.
 - **motoFP.py:** Contiene funciones para explotar los datos de las carreras de MotoFP.
 - **motoFP_test.py:** Contiene funciones de test para probar las funciones del módulo motoFP.py. En este módulo está el main.
 - Puede añadir otros módulos para funciones auxiliares si así lo desea.
- **/data:** Contiene el dataset del proyecto.
 - **mundial_motofp.csv:** Archivo con los resultados de las carreras de MotoFP.
- **/doc:** Contiene la documentación del proyecto.
 - **identificacion.md:** Archivo con los datos del alumno que realiza el examen.
 - **FP2425-Parcial1-Sesión 1-Enunciado.pdf:** Enunciado del examen.

Ejercicios

Se pide implementar las siguientes funciones (en el módulo motoFP.py) y sus funciones de test correspondientes con los parámetros adecuados (en el módulo motoFP_test.py). Las puntuaciones indicadas para cada ejercicio incluyen la realización de dichos tests. Tenga en cuenta que se pueden definir funciones auxiliares cuando se considere necesario.

Para cada función solicitada, se le proporciona el prototipo de la función (donde puede ver los parámetros, sus tipos y el tipo de salida), así como una posible salida del test correspondiente.

Ejercicio 1 (1 punto)

```
def lee_carreras(filename: str) -> list[CarreraFP]
```

Recibe la ruta de un fichero CSV y devuelve una lista de tuplas de tipo CarreraFP conteniendo todos los datos almacenados en el fichero.

Test lee_carreras

Número de carreras leídas: 200

Las dos primeras son:

```
CarreraFP(fecha_hora=datetime.datetime(2023, 3, 1, 14, 0),
circuito='Le Mans', pais='Francia', seco=True, tiempo=114.938,
podio=[Piloto(nombre='Francesco Bagnaia', escuderia='Ducati'),
Piloto(nombre='Johann Zarco', escuderia='Honda'), Piloto(nombre='Fabio
Quartararo', escuderia='Yamaha')])
CarreraFP(fecha_hora=datetime.datetime(2023, 3, 12, 17, 0),
circuito='Motegi', pais='Japón', seco=True, tiempo=101.52,
podio=[Piloto(nombre='Fabio Quartararo', escuderia='Yamaha'),
Piloto(nombre='Brad Binder', escuderia='KTM'), Piloto(nombre='Francesco
```

```
Bagnaia', escuderia='Ducati']])
```

Las dos últimas son:

```
CarreraFP(fecha_hora=datetime.datetime(2030, 8, 10, 9, 0),
circuito='Valencia', pais='España', seco=True, tiempo=102.827,
podio=[Piloto(nombre='Jorge Martin', escuderia='Ducati'),
Piloto(nombre='Marc Marquez', escuderia='Honda'), Piloto(nombre='Pedro
Acosta', escuderia='KTM')])
```

```
CarreraFP(fecha_hora=datetime.datetime(2030, 8, 31, 12, 0),
circuito='Phillip Island', pais='Australia', seco=True, tiempo=119.885,
podio=[Piloto(nombre='Johann Zarco', escuderia='Honda'),
Piloto(nombre='Fabio Quartararo', escuderia='Yamaha'),
Piloto(nombre='Maverick Viñales', escuderia='Aprilia')])
```

Ejercicio 2 (1,5 puntos)

```
def maximo_dias_sin_ganar(carreras: list[CarreraFP], nombre_piloto: str)
-> int:
```

Devuelve el tiempo máximo (en días) que nombre_piloto estuvo sin ganar una carrera. Es decir, el número máximo de días transcurridos entre dos carreras ganadas por el piloto. Si el piloto no ha ganado al menos dos carreras, la función debe devolver None.

Test maximo_dias_sin_ganar

Para Marc Marquez: 424

Para Jorge Martin: 629

Para Freddy Mercuri: None

Ejercicio 3 (1,5 puntos)

```
def piloto_mas_podios_por_circuito(carreras: list[CarreraFP]) ->
dict[str,str]:
```

Devuelve un diccionario que a cada circuito le hace corresponder el nombre del piloto que más veces ha estado en el podio en ese circuito.

Test piloto_mas_podios_por_circuito

```
{'Le Mans': 'Johann Zarco',
'Motegi': 'Joan Mir',
'Sepang': 'Maverick Viñales',
'Mugello': 'Marc Marquez',
'Circuit of the Americas': 'Marc Marquez',
'Aragón': 'Aleix Espargaró',
'Sachsenring': 'Pedro Acosta',
'Barcelona-Catalunya': 'Pedro Acosta',
'Misano': 'Aron Canet',
'Red Bull Ring': 'Pedro Acosta',
'Valencia': 'Pedro Acosta',
```

```
'Termas de Río Hondo': 'Johann Zarco',  
'Silverstone': 'Fabio Quartararo',  
'Losail International Circuit': 'Joan Mir',  
'Phillip Island': 'Joan Mir',  
'Jerez': 'Jorge Martin',  
'Assen': 'Pedro Acosta'}
```

Ejercicio 4 (2 puntos)

```
def escuderias_con_solo_un_piloto(carreras: list[CarreraFP]) ->  
list[str]:
```

Devuelve una lista con las escuderías que solo tienen un piloto.

```
Test escuderias_con_solo_un_piloto  
['Suzuki', 'KALEX']
```

Ejercicio 5 (2 puntos)

```
def piloto_racha_mas_larga_victorias_consecutivas(carreras:  
list[CarreraFP], año: int|None = None) -> tuple[str, int]:
```

Devuelve el nombre del piloto que ha obtenido la racha de victorias consecutivas más larga ese año y la longitud de esa racha. Si año es None entonces el resultado será sin restricción de año. Una racha es una secuencia de resultados iguales y seguidos, por ejemplo, si G es gana y N no gana, y los resultados de un piloto son GNN**GGG**NNNNGG, la racha más larga de victorias es de tres (señalada en negrita).

```
Test piloto_racha_mas_larga_victorias_consecutivas  
Para año=2024 ('Johann Zarco', 2)  
Para año=None ('Pedro Acosta', 3)
```

Ejercicio 6 (2 puntos)

```
def ultimos_ganadores_por_circuito(carreras: list[CarreraFP], n: int,  
estado: str) -> dict[str, list[str]]:
```

Devuelve un diccionario que a cada circuito le hace corresponder una lista con los nombres de los pilotos ganadores de las últimas n carreras disputadas en el circuito con el estado dado como parámetro. El estado puede ser "Seco" o "Mojado". Las listas de los nombres de los pilotos del diccionario estarán ordenadas por la fecha de la carrera que ganaron, de más reciente a más antigua.

```
Test ultimos_ganadores_por_circuito  
Para n=2 y estado="Seco"  
{ 'Le Mans': ['Enea Bastianini', 'Francesco Bagnaia'],  
  'Motegi': ['Andrea Iannone', 'Andrea Iannone'],
```

```
'Valencia': ['Jorge Martin', 'Andrea Iannone'],
'Sepang': ['Maverick Viñales', 'Pedro Acosta'],
'Misano': ['Pedro Acosta', 'Aron Canet'],
'Mugello': ['Johann Zarco', 'Fabio Quartararo'],
'Phillip Island': ['Johann Zarco', 'Maverick Viñales'],
'Circuit of the Americas': ['Johann Zarco', 'Francesco Bagnaia'],
'Aragón': ['Johann Zarco', 'Marc Marquez'],
'Sachsenring': ['Francesco Bagnaia', 'Maverick Viñales'],
'Red Bull Ring': ['Andrea Iannone', 'Fabio Quartararo'],
'Termas de Río Hondo': ['Maverick Viñales', 'Joan Mir'],
'Silverstone': ['Fabio Quartararo', 'Maverick Viñales'],
'Losail International Circuit': ['Jorge Martin', 'Marc Marquez'],
'Jerez': ['Aron Canet', 'Aleix Espargaro'],
'Assen': ['Aron Canet', 'Johann Zarco'],
'Barcelona-Catalunya': ['Pedro Acosta', 'Maverick Viñales']}]}
```

Para n=2 y estado="Mojado"

```
{ 'Le Mans': ['Johann Zarco', 'Brad Binder'],
  'Barcelona-Catalunya': ['Jack Miller'],
  'Sepang': ['Francesco Bagnaia', 'Enea Bastianini'],
  'Misano': ['Jack Miller', 'Jorge Martin'],
  'Jerez': ['Enea Bastianini', 'Jorge Martin'],
  'Phillip Island': ['Francesco Bagnaia'],
  'Losail International Circuit': ['Enea Bastianini', 'Jack Miller'],
  'Silverstone': ['Marc Marquez', 'Fabio Quartararo'],
  'Assen': ['Marc Marquez', 'Fabio Quartararo'],
  'Circuit of the Americas': ['Marc Marquez', 'Pedro Acosta'],
  'Sachsenring': ['Brad Binder'],
  'Termas de Río Hondo': ['Johann Zarco', 'Johann Zarco'],
  'Aragón': ['Pedro Acosta'],
  'Valencia': ['Pedro Acosta', 'Aleix Espargaro'],
  'Red Bull Ring': ['Jorge Martin', 'Andrea Iannone'],
  'Mugello': ['Fabio Quartararo', 'Brad Binder']}]}
```