

Exame de programación en Python, Diciembre 2020

Nome:

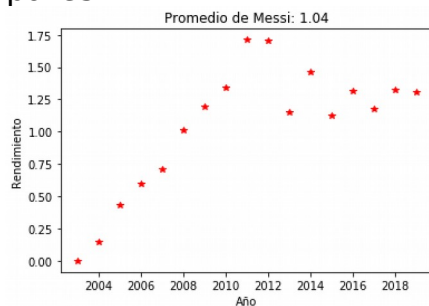
Maradona ha sido uno de los mejores futbolistas del mundo. Se retiró en 1997 con 259 goles y 163 asistencias en un total de 491 partidos de liga oficial. Los datos de Messi son 444 goles y 194 asistencias en 483 partidos oficiales de liga hasta 2019.

Nos gustaría responder a la pregunta: ¿Cuántos ligas más tendría que haber jugado Maradona a rendimiento máximo para superar a Messi? Para ello, usaremos los datos reales de Maradona y Messi (para éste hasta 2019, que corresponde a la liga 2019/2020).

1. Defina una función de lectura de fichero para poder leer el tipo de fichero proporcionado: **leerdatos (infile)**, donde infile es la variable que devuelve la función open de python. La función lee del fichero el año de inicio y los datos necesitados para calcular el rendimiento del jugador para cada año, dado por la expresión:
rendimiento=(goles+0,7*asistencias)/ partidos

La función devuelve el año inicial y una lista de rendimientos.

2. Desde el programa principal, abra el fichero messi.dat y llame a la función de leerdatos para obtener el año inicial de Messi en la liga y la lista de sus rendimientos, y cierre el fichero. Repita la operación para el fichero maradona.dat. El programa debe controlar los errores de lectura del fichero.
3. Defina una función dibuja con los parámetro necesarios para dibujar la lista de rendimientos en función del año. Un ejemplo de lo que debe mostrar aparece en la figura. El título debe mostrar el nombre del jugador pasado como parámetro de entrada, así como la media de sus rendimientos anuales. Ponga ticks en los años pares.



4. Llame a la función dibuja dos veces desde el programa principal: una para Messi y otra para Maradona.

5. Calcule e imprima el número de años adicionales que tendría que jugar Maradona a rendimiento máximo para poder alcanzar el rendimiento medio de Messi hasta 2019.

SOLUCIÓN:

```
from numpy import int_,arange,mean
from matplotlib.pyplot import plot,show,xlim,title,xlabel,ylabel,xticks
```

```
def leerdatos(file):
```

```
    file.readline()
    file.readline()
    año=int(file.readline())
    file.readline()
    rendimiento=[]
```

```
    for line in file:
        v=int_(line.split())
```

```
        if v[0]==0:
            rendimiento.append(0)
        else:
            rendimiento.append((v[1]+v[2]*0.7)/v[0])
```

```
    return año,rendimiento
```

```
#####
```

```
def dibuja (añoinicial,y,nombrejg):
    x=int_(arange(añoinicial,añoinicial+len(y)))
    xticks(x[x%2==0])
    plot(x,y,'*r')
    xlabel("Año")
    ylabel("Rendimiento")
    prom=sum(y)/len(y)
    title("Promedio de %s: %.2f"%(nombrejg,prom))
    show()
```

```
#####
```

```
#PROGRAMA PRINCIPAL
```

```
#####
```

```
try:
```

```
    filename=open("messi.dat")
    añomessi,rendmessi= leerdatos(filename)
    filename.close()
    dibuja(añomessi,rendmessi,"Messi")
```

```
filename=open("maradona.dat")
añosmara,rendmara= leerdatos(filename)
filename.close()
dibuja(añosmara,rendmara,"Maradona")

prommessi=mean(rendmessi)
prommara=mean(rendmara)

añosañadidos=0
while(prommara<prommessi):
    rendmara.append(max(rendmara))
    prommara=mean(rendmara)
    añosañadidos=añosañadidos+1

print("Maradona tendría que jugar %d años más para alcanzar a Messi
en 2019"%añosañadidos)

except IOError:
    print ('Error al abrir fichero')
```