

Series TV

Se cuenta con la siguiente lista de series de TV:

```
series = [("los soprano", 6, 1999, "HBO"),
          ("lost", 6, 2004, "ABC"),
          ("4400", 4, 2004, "CBS"),
          ("United States of Tara", 3, 2009, "Dreamworks"),
          ("V", 3, 2009, "Warner Bros"),
          ("dr house", 7, 2004, "Universal")]
```

Cada tupla representa:

- El nombre de la serie
- La cantidad de temporadas
- En qué año se emitió la primera temporada
- Qué cadena lo produjo

Por otra parte se tiene la lista de actores (la muestra es sólo un ejemplo):

```
actores = [("Ken Leung", ["lost", "los soprano"]),
           ("Joel Gretsch", ["4400", "V", "United States of Tara"]),
           ("James Gandolfini", ["los soprano"]),
           ("Elizabeth Mitchell", ["dr house", "V", "lost"])]
```

El formato que sigue la tupla es:

- Nombre del actor
- Lista de series en las que participó

Se cuenta con estas funciones:

```
serie (s, _, _, _) = s                anioComienzo (_, _, a, _) = a
temporadas (_, t, _, _) = t          seriesDeActor = snd
cadenaTV (_, _, _, c) = c            nombreActor = fst
find criterio = head . filter criterio
```

Codificar las funciones indicadas a continuación. En la resolución tenga presente que deben aparecer aplicados, al menos una vez, los siguientes conceptos:

- Aplicación parcial
- Composición
- Funciones de orden superior
- Listas por comprensión

No se puede usar recursividad, a menos que esté indicado en el punto.

Se pide

- 1) Encontrar los datos de una serie en base al nombre

```
>datosDe "4400"
("4400", 4, 2004, "CBS")
```

- 2) Conocer la lista de actores que trabajaron en una serie

```
>listaDeActoresDe "los soprano"
["Ken Leung", "James Gandolfini"]
```

3) Conocer la lista de actores que actuaron en dos series diferentes

```
> quienesActuaronEn "V" "lost"
["Elizabeth Mitchell"]
```

4) a) Poder determinar el año de una serie en base a su nombre

```
> anioDeComienzoDe "lost"
2004
```

b) Saber si una lista de series está ordenada por año de comienzo.

```
> seriesOrdenadas ["dr house", "V", "lost"]
```

False (House arranca en el 2004 y “V Invasión extraterrestre” en el 2009 –hasta acá es correcto- pero luego sigue con “Lost” que comenzó en el 2004)

Este es el único punto en donde se puede usar recursividad.

5) Queremos saber cuáles son las series que cumplen un determinado criterio. Tenemos esta función.

```
queSeriesCumplen unCriterio = map serie . filter unCriterio
```

Utilizar la función para resolver:

- Qué series duraron más de 3 temporadas
- Qué series tuvieron más de 4 actores
- Qué series tienen un título de menos de 5 letras

Nota: no se pueden definir funciones auxiliares ni expresiones lambda/definiciones locales

6) a) Queremos saber en promedio cuántas temporadas duran las series

```
> promedioGeneral
```

```
4 (6 + 6 + 4 + 3 + 3 + 7 = 29 / 6 series, en división entera me da 4)
```

b) Agregar como parámetro una función que devuelva un valor al cual se le calcule el promedio (con división entera) de todas las series

```
> promedio anioComienzo
```

```
2004 (los años de comienzo de cada serie / 6 series)
```

```
> promedio anioFin where anioFin serie = anioComienzo serie + temporadas
serie
```

```
2009 (2005 + 2010 + 2008 + 2012 + 2012 + 2011 = 12058 / 6 = 2009, 666)
```

7) Inferir los tipos de la función funcionHeavy

```
funcionHeavy a b c d e | d > e      = map a c
                       | otherwise = map b c
```