

RETO 1 - AJEDREZ

Tarea 1

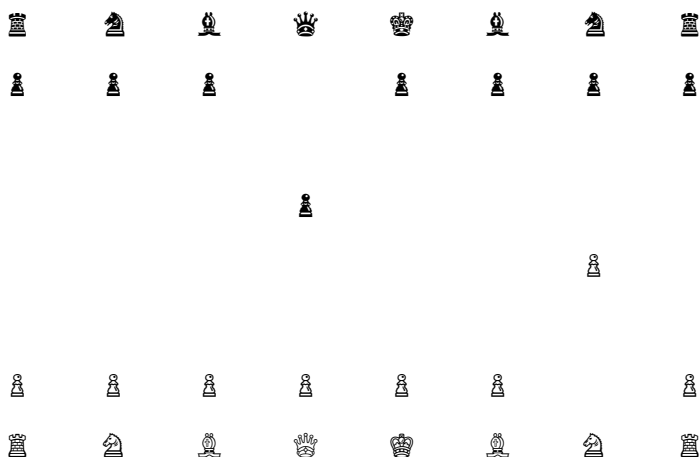
La primera tarea consiste en escribir un programa que guarde en un fichero la secuencia de tableros de una partida de ajedrez. Partiremos del tablero inicial donde las filas del tablero están separadas por cambios de línea y las columnas por tabuladores.

El programa debe guardar el tablero inicial en un fichero con el nombre que elija el usuario. Después debe preguntar al usuario si quiere hacer un movimiento o terminar la partida. Cada vez que el usuario quiera hacer un nuevo movimiento debe preguntar la fila y la columna de la pieza que quiere mover y la fila y la columna a la que la quiere mover. Tras ello añadirá el tablero resultante al final del fichero anterior.

El fichero [partida-ajedrez.txt](#) contiene un ejemplo con el fichero resultante de una partida con 3 movimientos.

Tarea 2

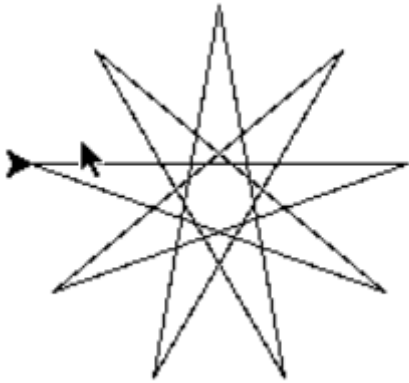
Una vez generado el fichero con los tableros sucesivos de una partida de ajedrez, el programa preguntará por un movimiento y mostrará por pantalla el tablero correspondiente ese movimiento. Por ejemplo, utilizando el fichero [partida-ajedrez.txt](#), si el usuario introduce el movimiento 2, debería mostrar por pantalla el siguiente tablero:



RETO 2 – ESTRELLAS REGULARES

Tarea

El reto consiste en programar una función que dibuje estrellas como la que aparece a continuación con un número de puntas dado.



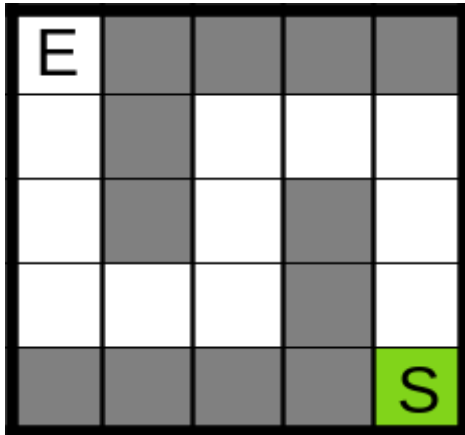
Para ello hay que utilizar el módulo de Python [Turtle](#) que permite realizar trazos sencillos en una ventana gráfica.

En el siguiente video hay una introducción básica al módulo Turtle.

RETO 3 - LABERINTO

Tarea 1

La primera tarea consiste en construir un laberinto como el de la siguiente figura.



El laberinto se representará como una lista de listas, donde cada lista es una fila del laberinto y cada casilla se representará con un espacio ' ' si hay paso o con la letra 'X' si hay un muro, tal y como se muestra a continuación:

```
laberinto = [  
    ['X', 'X', 'X', 'X', 'X'],  
    ['X', 'X', ' ', ' ', ' '],  
    ['X', ' ', ' ', 'X', ' '],  
    [' ', ' ', ' ', 'X', ' '],  
    ['X', 'X', 'X', 'X', 'S']  
]
```

El laberinto se debe crear a partir de una tupla con las coordenadas de las casillas donde hay muro, como la siguiente:

```
muro = ((0,1), (0,2), (0,3), (0,4), (1,1), (2,1), (2,3), (3,3), (4,0), (4,1), (4,2), (4,3))
```

Tarea 2

La segunda tarea a resolver consiste en construir un programa para recorrer el laberinto desde la entrada a la salida. La entrada siempre está en la esquina superior izquierda y la salida en la esquina inferior derecha.

El programa debe devolver una lista con la secuencia de movimientos para ir de la entrada a la salida del laberinto, tal y como se muestra a continuación:

```
['Abajo', 'Abajo', 'Abajo', 'Abajo', 'Derecha', 'Derecha', 'Arriba', 'Arriba', 'Derecha', 'Derecha',  
'Abajo', 'Abajo', 'Abajo']
```