



Universidad  
Rey Juan Carlos

## Test modificaciones práctica 1

Inteligencia Artificial

### Introducción

En este documento se proponen una serie de modificaciones al contenido de la práctica 1. Las modificaciones propuestas sólo tienen como objetivo el mundo del Pacman, no es necesario ajustarlas para otros problemas. Las modificaciones propuestas deben ir dentro de los archivos `search.py` y `searchAgents.py`, no es necesario hacer ningún cambio en otros archivos. Si se hacen cambios en otros ficheros, adjuntar el fichero modificado dentro del .zip de entrega.

En todos los puntos que se modifique la práctica, se deberá marcar claramente con un comentario en la línea inmediatamente superior a qué modificación se refiere el cambio, siguiendo la siguiente estructura:


```
def chebyshovDistance(...):  
    """ MODIFICACION 1 """  
    ...  
  
def ...(...):  
    """ MODIFICACION 2 """  
    ...
```

### 1. Heurística de distancia de Chebyshov para la búsqueda A\*

Se pide añadir una nueva heurística como método Python `chebyshovDistance` que calcule la distancia de Chebyshov para la búsqueda A\*. La distancia de Chebyshov viene definida por:

$$D_{\text{Chev}} = \max(|x_2 - x_1|, |y_2 - y_1|).$$

Un ejemplo de su uso en un tablero de ajedrez sería:

	a	b	c	d	e	f	g	h	
8	5	4	3	2	2	2	2	2	8
7	5	4	3	2	1	1	1	2	7
6	5	4	3	2	1		1	2	6
5	5	4	3	2	1	1	1	2	5
4	5	4	3	2	2	2	2	2	4
3	5	4	3	3	3	3	3	3	3
2	5	4	4	4	4	4	4	4	2
1	5	5	5	5	5	5	5	5	1
	a	b	c	d	e	f	g	h	

Se podrá lanzar el algoritmo de la siguiente forma:

```
python3 pacman.py -l tinyMaze -z .5 -p SearchAgent -a
fn=astar,heuristic=chebyshevDistance
```

```
python3 pacman.py -l mediumMaze -z .5 -p SearchAgent -a
fn=astar,heuristic=chebyshevDistance
```

```
python3 pacman.py -l bigMaze -z .5 -p SearchAgent -a
fn=astar,heuristic=chebyshevDistance
```

## 2. Modificación de función de coste

Se pide modificar el cálculo del coste de las acciones que realiza Pacman. Para la nueva implementación queremos que el coste de los movimientos que no sean hacia el norte tengan un coste de 2x respecto al coste normal esperado para dichos movimientos. Además de esto, el coste de un movimiento de Pacman hacia el norte será de 0. Resumiendo:

- Movimiento Norte: 0
- Movimiento Sur, Este, Oeste: 2x coste.

Explicar en el comentario donde se realicen las modificaciones los cambios en los resultados observados para el algoritmo A\* respecto a los resultados previos.

Se recuerdan los comandos para el lanzamiento del algoritmo:

```
python3 pacman.py -l tinyMaze -z .5 -p SearchAgent -a
fn=astar,heuristic=manhattanHeuristic
```

```
python3 pacman.py -l mediumMaze -z .5 -p SearchAgent -a  
fn=astar,heuristic=manhattanHeuristic
```

```
python3 pacman.py -l bigMaze -z .5 -p SearchAgent -a  
fn=astar,heuristic=manhattanHeuristic
```

## Entregables

- Se crearán 2 carpetas (**sin modificar**, **modificada**) en las que se incluirán los archivos de código Python **search.py** y **searchAgents.py**. En la primera irá el archivo que se ha ido completando para la práctica a lo largo de las sesiones y en la segunda el archivo con las modificaciones propuestas en este documento.
- Ambas carpetas se comprimirán en un único archivo de nombre: practica1\_NombreApellidos.zip/tar.gz que se entregará vía Aula Virtual.
- El código tiene que ir obligatoriamente comentado explicando su funcionalidad, incluyendo las modificaciones. Debe ser legible y estar debidamente tabulado.
- Se utilizarán sistemas anticopia y se podrá requerir explicación individual de la práctica en caso de duda.
- Entregar la práctica **sin modificar** y la práctica **modificada** con las propuestas de este documento.

Para comprimir las carpetas:

```
tar -czf practica1_NombreApellidos.tar.gz modificada/ sin_modificar/  
zip -r practica1_NombreApellidos.zip modificada/ sin_modificar/
```