

# Primera práctica de Inteligencia Artificial

## Curso 2019-2020

Departament d'Informàtica i Enginyeria Industrial  
Universitat de Lleida  
carlos@diei.udl.cat  
jojeda@diei.udl.cat  
eduard.torres@udl.cat

### 1. Enunciado

El objetivo de esta práctica es evaluar el conocimiento del alumno sobre los algoritmos de búsqueda informada y no informada descritos hasta el momento en la asignatura. Dichos algoritmos se aplicarán en el contexto de los proyectos docentes de inteligencia artificial sobre el popular juego, Pac-Man <sup>1</sup>.

#### 1.1. Descripción de los proyectos Pac-Man

En la página central de los proyectos Pac-Man podemos leer: “Los proyectos Pac-Man se desarrollaron para un curso introductorio de inteligencia artificial (CS 188) de UC Berkeley. Se aplican una serie de técnicas de IA para jugar al Pac-Man. Sin embargo, estos proyectos no se centran en la construcción de la IA de los videojuegos. En su lugar, se enseñan conceptos fundamentales, tales como búsqueda informada en el espacio de estados de un problema, inferencia probabilística y aprendizaje por refuerzo. Estos conceptos subyacen en áreas de aplicación del mundo real, tales como el procesamiento de lenguaje natural, la visión artificial y robótica.”

Nosotros nos centraremos en esta práctica en el proyecto de búsqueda <sup>2</sup>.

#### 1.2. Algoritmos: 7 puntos

Los algoritmos que se deberán implementar son los siguientes:

- Búsqueda informada (búsqueda en grafo) - **5 puntos**
  - UCS - **2 puntos**
  - Best-H first - **1 punto**
  - A\* - **4 puntos**

Observaciones:

- Considerar que el coste de transición de un estado a otro es 1.
- Utilizar como heurísticas: distancia de Manhattan y distancia Euclídea.
- Siempre que sea posible debes dar la versión optimal del algoritmo.

En este apartado se valora la corrección de los algoritmos implementados.

**Opcional:** Implementar un algoritmo de búsqueda bidireccional.

**Opcional:** Proponer una heurística adicional.

---

<sup>1</sup>[http://ai.berkeley.edu/project\\_overview.html](http://ai.berkeley.edu/project_overview.html)

<sup>2</sup><http://ai.berkeley.edu/search.html>

### 1.3. Implementación: 1 punto

En este apartado se valora la calidad y eficiencia de los algoritmos implementados. Es necesario tener en cuenta los siguientes aspectos de la implementación:

- El lenguaje de programación es Python.
- La utilización de un diseño orientado a objetos.
- La simplicidad y legibilidad del código.
- Se recomienda seguir la guía de estilo <sup>3</sup>.

**Nota 1:** Tal como se ha comentado en clase, la corrección inicial será realizada con una herramienta automática. Por lo tanto para garantizar su correcto funcionamiento tendréis que ceñiros a la siguiente nomenclatura de las funciones:

- Bidireccional  $\rightarrow$  bds
- Best-H  $\rightarrow$  bfsh
- Best-G (UCS)  $\rightarrow$  ucs
- A\*  $\rightarrow$  astar

En el caso de las heurísticas los nombres son los siguientes:

- Manhattan  $\rightarrow$  mandH
- Propuesta por vosotros  $\rightarrow$  custH
- Euclídea  $\rightarrow$  eucdH

A vuestras funciones podéis ponerles el nombre que más os guste y luego al final del archivo *search.py* veréis una sección de *Abbreviations* donde podréis asignar otro nombre a las funciones que tengáis definidas, por ejemplo:

```
# ...
# Abbreviations
ucs = uniformCostSearch
bds = bidirectionalSearch
bfsh = greedyBestFirstSearch
astar = aStarSearch
# ...
```

**Nota 2:** La salida por pantalla del programa que espera la herramienta es la que tiene el proyecto Pac-Man por defecto, si para hacer *debug* necesitáis imprimir algo por pantalla, tomad precauciones para evitar confundir a la herramienta. Dos posibles soluciones:

- Quitad las líneas que imprimen por pantalla antes de entregar la práctica.
- Utilizad la salida estándar de errores `'print >> sys.stderr, "texto"'`.

### 1.4. Evaluación Experimental - 1 punto

A partir de las estadísticas proporcionadas tras la ejecución de los algoritmos sobre los diferentes mapas proporcionados, debéis valorar el rendimiento de los algoritmos.

Mostrad una tabla con los mapas y el rendimiento de los algoritmos. Utilizad como criterio el tiempo de ejecución y el número de nodos generados. La opción '-q' de la herramienta pacman anula el modo gráfico.

**Opcional:** Utiliza el generador de mapas aleatorios realizado en la sesión 1.

### 1.5. Documentación - 1 punto

Documento en pdf (máximo  $\approx$  4 páginas) que incluya:

- una descripción de las decisiones que se han tomado en la implementación de los algoritmos y los resultados de la evaluación experimental

La página inicial ha de contener el nombre de los integrantes del grupo. La práctica puede realizarse en grupos de dos personas o individualmente.

Se valorará la redacción y presentación del documento.

---

<sup>3</sup><https://www.python.org/dev/peps/pep-0008/>

## 1.6. Material a entregar

El material evaluable de esta práctica es:

- Todos los archivos de código fuente, modificados o añadidos. Estos ficheros se descargarán sobre el contenido del directorio *search*.
- Documento de la práctica.

Todo el material requerido se entregará en el paquete de nombre `ia-prac1.[tgz|tar.gz|zip]`

```
ia-prac1.[tgz|tar.gz|zip]
├── pacman
│   └── *.py
└── informe.pdf
```