# Estructura de la memoria

**Título** Practica 1: Busqueda

**Autores** Mario Iribias y Miguel Ibáñez

## Algoritmos de búsqueda (search.py)

* 1. **Diseño general (0.5 puntos):**
     + 1. **Describid la estructura de datos utilizada para representar el nodo de búsqueda y si ha sido necesario modificarla a medida que se realizaba la práctica para facilitar la implementación de alguno de los algoritmos de búsqueda.**
       2. **Explicad las alternativas que se han barajado y las razones por las cuales se ha elegido dicha representación.**

## Sección 1 (2.0 puntos)

* 1. **Diseño:**
     + 1. **Explicad el enfoque adoptado para diseñar el algoritmo de búsqueda.**
       2. **Listad y explicad las funciones que habéis utilizado, de entre las proporcionadas para realizar la práctica.**
       3. **Incluid y explicad el código que habéis implementado.**
       4. **¿cuántos nodos expande?**
       5. **¿llega a la solución óptima?**
       6. **¿Es óptimo?**
  2. **Responded a la pregunta 1.1 del enunciado.**
  3. **Responded a la pregunta 1.2 del enunciado.**
  4. **Tests: Incluid las pruebas realizadas para ilustrar los resultados de la búsqueda.**

**Utilizad capturas de pantalla para analizar las pruebas realizadas, los resultados de su ejecución e ilustrar vuestras respuestas.**

## Sección 2 (0.75 puntos)

* 1. **Diseño:**
     + 1. **Explicad el enfoque adoptado para diseñar el algoritmo de búsqueda.**
       2. **Listad y explicad las funciones que habéis utilizado, de entre las proporcionadas para realizar la práctica.**
       3. **Incluid y explicad el código que habéis implementado.**
  2. **Eficiencia del algoritmo de búsqueda:**
     + 1. **¿cuántos nodos expande?**
       2. **¿llega a la solución óptima?**
       3. **¿Es óptimo?**
  3. **Responded a la pregunta 2.1 del enunciado.**
  4. **Tests: Incluid las pruebas realizadas para ilustrar los resultados de la búsqueda.**

**Utilizad capturas de pantalla para analizar las pruebas realizadas, los resultados de su ejecución e ilustrar vuestras respuestas.**

## Sección 3 (0.75 puntos)

* 1. **Diseño:**
     + 1. **Explicad el enfoque adoptado para diseñar el algoritmo de búsqueda.**
       2. **Listad y explicad las funciones que habéis utilizado, de entre las proporcionadas para realizar la práctica.**
       3. **Incluid y explicad el código que habéis implementado.**
  2. **Eficiencia del algoritmo de búsqueda:**
     + 1. **¿cuántos nodos expande?**
       2. **¿llega a la solución óptima?**
       3. **¿Es óptimo?**
  3. **Tests: Incluid las pruebas realizadas para ilustrar los resultados de la búsqueda.**

**Utilizad capturas de pantalla para analizar las pruebas realizadas, los resultados de su ejecución e ilustrar vuestras respuestas.**

## Sección 4 (2 puntos)

* 1. **Diseño**
     + 1. **Explicad el enfoque adoptado para diseñar el algoritmo de búsqueda.**
       2. **Listad y explicad las funciones que habéis utilizado, de entre las proporcionadas para realizar la práctica.**
       3. **Incluid y explicad el código que habéis implementado.**
  2. **Eficiencia del algoritmo de búsqueda:**
     + 1. **¿cuántos nodos expande?**
       2. **¿llega a la solución óptima?**
       3. **¿Es óptimo?**
  3. **Responded a la pregunta 4.1 del enunciado.**
  4. **Tests: Incluid las pruebas realizadas para ilustrar los resultados de la búsqueda.**

**Utilizad capturas de pantalla para analizar las pruebas realizadas, los resultados de su ejecución e ilustrar vuestras respuestas.**

## Agentes basados en búsqueda (searchAgents.py)

## Sección 5 (2 puntos)

* 1. **Estado del sistema:**
     1. **Describid cómo se caracteriza el estado del sistema (no confundir con “nodo de búsqueda”) para resolver este problema de búsqueda.**
     2. **Explicad las alternativas que se han barajado y las razones por las cuales se ha elegido dicha representación.**
     3. **Incluid ejemplos de estados concretos.**
  2. **Implementación:**
     + 1. **Listad y explicad las funciones que habéis utilizado, de entre las proporcionadas para realizar la práctica.**
       2. **Incluid y explicad el código que habéis implementado.**
       3. **Explicad qué devuelve la función ‘getSuccessors’ para este problema.**
       4. **Describid la implementación ‘isGoalState’ para este problema.**
  3. **Tests: Incluid las pruebas realizadas para ilustrar la implementación.**

**Utilizad capturas de pantalla para analizar las pruebas realizadas, los resultados de su ejecución e ilustrar vuestras respuestas.**

## Sección 6 (2 puntos)

* 1. **Heurística:**
     + 1. **Describid el proceso que se ha seguido para diseñar la heurística.**
       2. **Explicad la lógica de la heurística.**
  2. **Implementación:**
     + 1. **Listad y explicad las funciones que habéis utilizado, de entre las proporcionadas para realizar la práctica.**
       2. **Incluid y explicad el código que habéis implementado.**
  3. **Tests: Incluid los resultados de las pruebas realizadas para analizar la heurística.**

**Utilizad capturas de pantalla para analizar las pruebas realizadas, los resultados de su ejecución e ilustrar vuestras respuestas.**

## Sección 7

**Comentarios personales.**