



# Trabajo TP6-1

## Resolución de problemas y búsqueda.

### Búsqueda Local

#### 1. Objetivo de la práctica

Familiarizarse con el código en java para resolver problemas de búsqueda local y diferenciar sus características. Volveremos a utilizar el código en <http://code.google.com/p/aima-java/>.

Tu trabajo consistirá básicamente en realizar familiarizarse con el algoritmo Hill-Climbing. Utilizaremos el problema de las 8-reinas. El algoritmo Hill-Climbing se “atasca” en un llano aproximadamente el 86% de las veces no pudiendo encontrar una solución, y consigue el éxito en el 14% de las veces. Cuando se atasca o consigue encontrar un objetivo lo hace relativamente rápido: 4 pasos de media cuando tiene éxito y 3 de media cuando se atasca, lo cual no está nada mal para un espacio de estados de  $8^8 \approx 17$  millones de estados.

#### 2. Tareas

1. Realiza 1000 experimentos a partir de estados iniciales de las 8-reinas con la clase **HillClimbingSearch**. Comprueba que da un porcentaje de éxitos con una media de pasos similar a la presentada.
2. Una buena solución para mejorar el porcentaje de éxito es permitir 100 pasos como máximo cuando se está en un llano (vecinos con valores iguales o peores). Crea la clase **HillClimbingSearchFlat** que permite dar un número máximo de pasos en llano. Realiza 1000 experimentos y comprueba cual es la mejora obtenida en porcentaje de éxitos y la media de pasos en caso de éxito y fallo. Realiza las mejoras que estimes oportunas.
3. Implementa el método **RandomRestartHillClimbing** que reinicia el estado inicial hasta que se obtiene el éxito. Muestra número de reintentos, solución y estadísticas.
4. Entrega una clase **NQueensTP61** en la que se muestre la realización de las tareas anteriores, y entrega una **memoria** comentando los experimentos y resultados obtenidos.

Nota: En la clase NQueensDemo se muestra un ejemplo de uso de la clase HillClimbingSearch. Pero ten cuidado porque las funciones que de forma incremental añaden una reina en cada columna a partir de un tablero vacío (NQueensFunctionFactory.getIActionsFunction()). Para hacer una búsqueda con HillClimbingSearch debemos partir de un estado completo y cambiar la posición de las reinas. Deberás utilizar las funciones en NQueensFunctionFactory.getCActionsFunction().