

# Proyecto #1: búsqueda heurística

Blai Bonet

24 de abril 2015

## Objetivo

El objetivo del proyecto es aprender sobre el modelo de espacio de estados y sobre los diferentes algoritmos de búsqueda heurística. No sólo se evaluará la correctitud de la implementación; es importante que los algoritmos sean eficientes y puedan resolver los problemas propuestos en los tiempos estipulados.

**Fecha de entrega: viernes 8 de mayo 2015**

## Problemas

Consideramos los siguientes problemas:

- n-puzzles: 4x4 y 5x5
- cubo de Rubik: 2x2x2 y 3x3x3
- top spin: 12-4, 14-4, 16-4 y 17-4
- Torre de Hanoi con 4 astas: 12, 14, 16 y 18 discos

En el problema de n-puzzle, consideramos dos modelos de costos: (a) costos unitarios para todos los movimientos y (b) costo de 1 para movimientos horizontales del blanco y 2 para movimientos verticales del blanco.

## Árboles de búsqueda

Estudiar los árboles de búsqueda y su factor de ramificación sin eliminación de duplicados y con eliminación parcial de duplicados.

Se debe crear tablas para cada problema donde se reporte el número de estado a cada profundidad en el árbol de búsqueda a partir del estado goal, hasta la profundidad máxima que se alcance en 15 minutos de ejecución.

## Algoritmos ciegos

Estudiar la búsqueda de soluciones óptimas con algoritmos ciegos. Buscar soluciones para las instancias dadas en cada problema utilizando los algoritmos: bfs con eliminación retardada de duplicados (DDD), ucs con DDD, dfid y depth-first uniform-cost search.

## Algoritmos informados

Estudiar la búsqueda de soluciones óptimas con algoritmos informados. Buscar soluciones para las instancias dadas en cada problema utilizando los algoritmos: a\* con DDD y ida\*. Para las heurísticas en cada problema:

- n-puzzle: Manhattan distance y diferentes additive PDBs
- cubo de Rubik: max de corner PDB y edge PDB
- Top Spin: max de diferentes PDBs
- Torre de Hanoi con 4 astas: max de diferentes PDBs