

## Proyecto 1

### 1. Objetivo

El objetivo del proyecto es la aplicar algoritmos informados de búsquedas y heurísticas en la solución del problema del 15-*Puzzle*.

### 2. Descripción de la actividad

En el trabajo realizado por Hasan et al [2] se presenta una heurística híbrida (Hybridizing Heuristic (HH)) para el 15-*Puzzle*. Para resolver 15-*Puzzle* usan Bidirectional (BA\*) con la heurística HH. Se tiene que HH es una combinación de tres heurísticas: Manhattan Distance (md), Walking Distance (wd)<sup>1</sup> [3] y Linear-Conflict Heuristic (lc) [1]. Dado un estado  $s$  del 15-*Puzzle*, la fórmula de HH es la siguiente:

$$h(s) = \frac{md(s)}{3} + lc(s) + wd(s) \quad (1)$$

Se quiere resolver el 15-*Puzzle* con otro algoritmo, y comparar sus resultados con los resultados obtenidos en [2]. En específico, debe resolver el 15-*Puzzle* usando el algoritmo **Recursive best-first search** (RBFS) con la heurística HH, o con IDA\* con la heurística HH.

La motivación del trabajo es que en [2] se muestra que el enfoque propuesto de BA\* con HH obtiene buenos resultados experimentales. Una de las pruebas consintió en comparar BA\* con HH, con respecto a IDA\* usando como heurística MD+LC, y los resultados obtenidos indican que es BA\* con HH es mejor método. La hipótesis es que RDFS con HH o IDA\* con HH deben obtener soluciones al 15-*Puzzle* de forma competitiva con BA\* con HH.

Se dará una puntuación extra de 10 puntos si usted diseña una heurística que sea competitiva con HH.

### 3. Requerimientos de la Implementación

Debe implementar su solución en un lenguaje de programación compilado. El ejecutable debe poder recibir por la entrada estándar el tablero a recibir. Los requerimientos de la salida son los siguientes. Si la instancia no se puede resolver se debe imprimir por la salida estándar la frase “No hay solución”. En caso de haber una solución, la salida deberá mostrar los siguientes datos: (i) la secuencia de los tableros que resuelven el *puzzle* y que componen el camino a la meta. Cada tablero se separa del otro con un salto de línea, (ii) La longitud del camino desde el estado inicial hasta la meta, (iii) imprime una línea con el número de estados generados.

Junto con el código fuente de su programa, debe entregar un archivo Leeme.txt, explicando como compilar y ejecutar su aplicación.

---

<sup>1</sup>Esta heurística es explicada en [2]

## 4. Condiciones de la entrega

El código del proyecto y la declaración de autenticidad debidamente firmada, deben estar contenidos en un archivo comprimido, con formato *tar.xz*, llamado *Proy1\_X\_Y.tar.xz*, donde *X* y *Y* son los número de carné de los estudiantes. La entrega del archivo *Proy1\_X\_Y.tar.xz*, debe hacerse por medio de la plataforma *Classroom* antes de las 10:00 A.M. del día jueves 20 de febrero de 2025.

## Referencias

- [1] HANSSON, O., MAYER, A., AND YUNG, M. Criticizing solutions to relaxed models yields powerful admissible heuristics. *Information Sciences* 63, 3 (1992), 207–227.
- [2] HASAN, D. O., ALADDIN, A. M., TALABANI, H. S., RASHID, T. A., AND MIRJALILI, S. The fifteen puzzle—a new approach through hybridizing three heuristics methods. *Computers* 12, 1 (2023), 11.
- [3] TAKAKEN. 15 how to create an automatic puzzle solving program. <https://computerpuzzle.net/puzzle/15puzzle/index.html>. En japonés, visitado el 05 de febrero de 2025.