



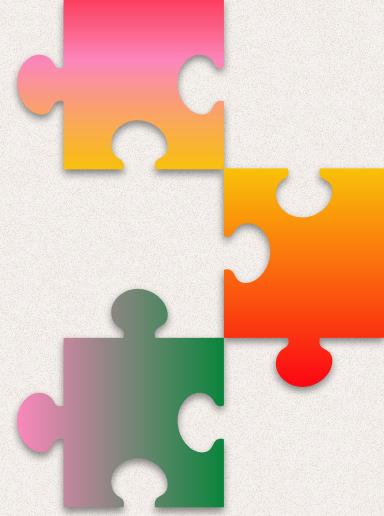
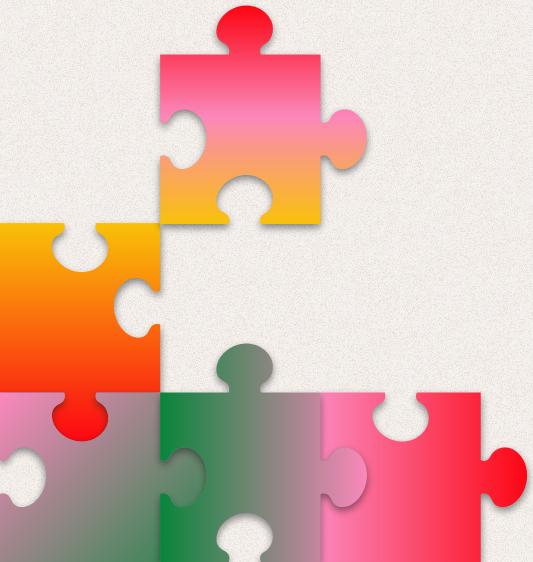
SIA - TP1

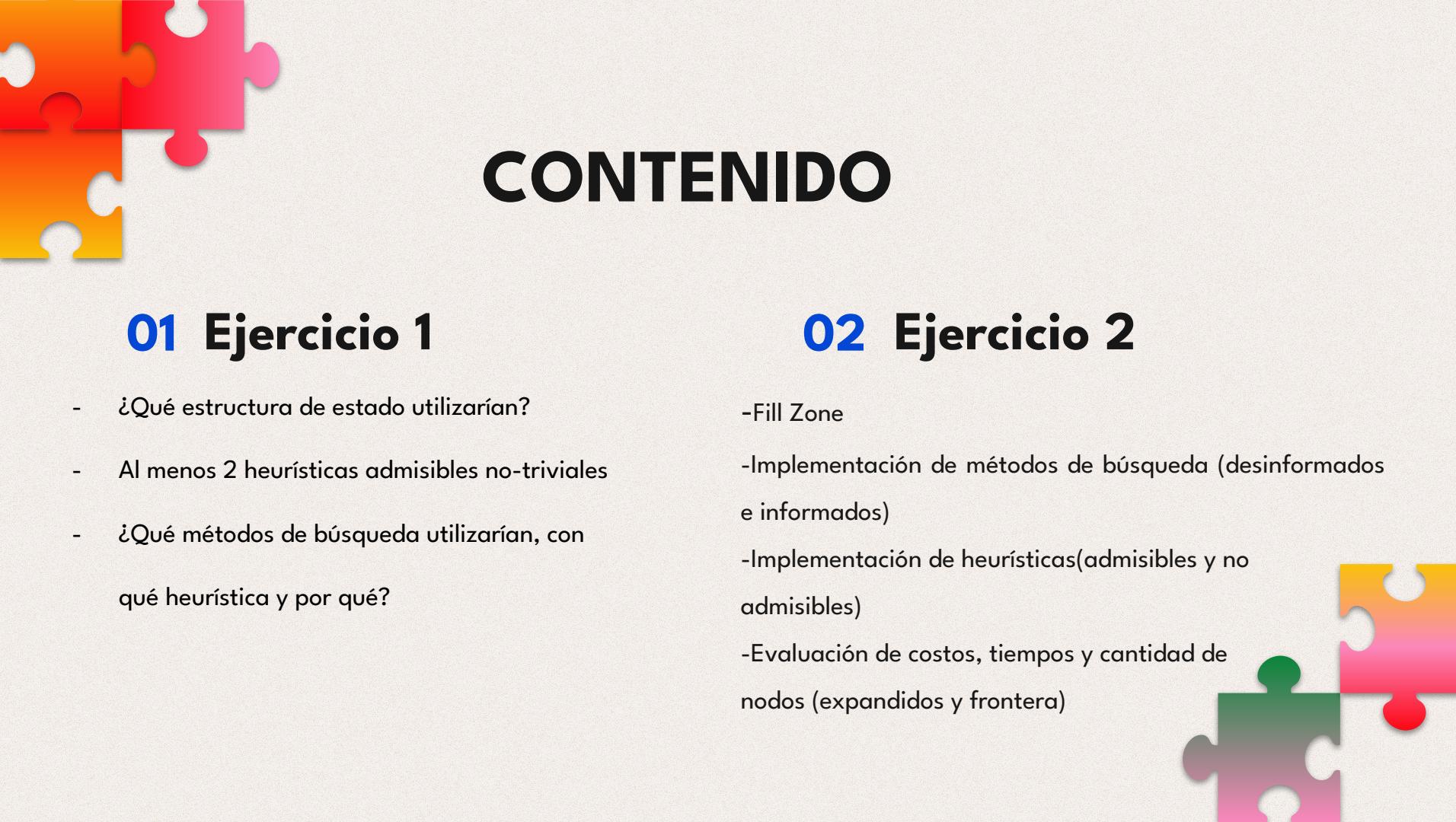
METODOS DE BUSQUEDA

Grupo 1

INTEGRANTES

Agustina, Lucciano, Mercedes, Victoria.
*Estudiantes de Ingeniería Informática y
Bioingeniería.*





CONTENIDO

01 Ejercicio 1

- ¿Qué estructura de estado utilizarían?
- Al menos 2 heurísticas admisibles no-triviales
- ¿Qué métodos de búsqueda utilizarían, con qué heurística y por qué?

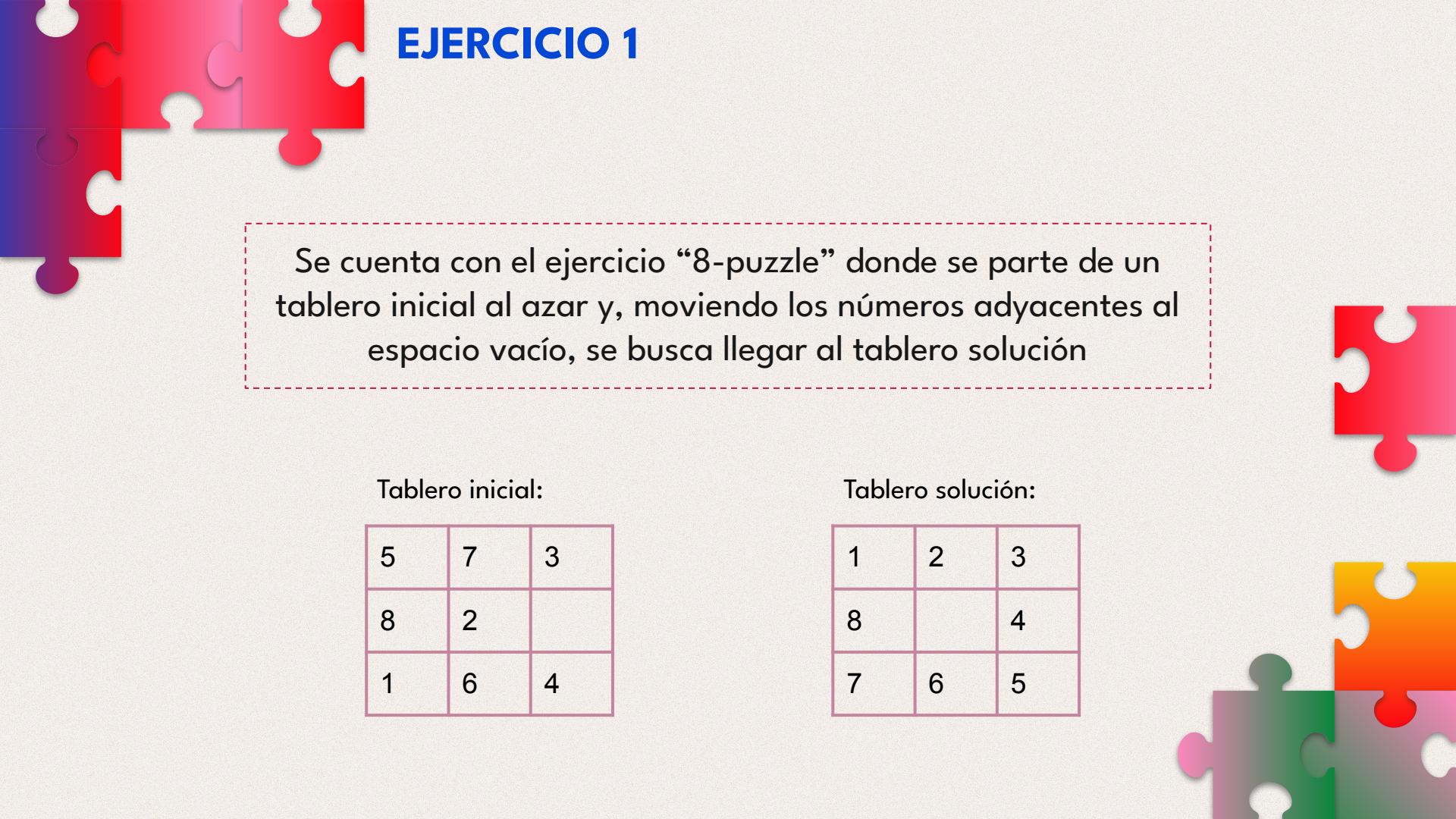
02 Ejercicio 2

- Fill Zone
- Implementación de métodos de búsqueda (desinformados e informados)
- Implementación de heurísticas(admisibles y no admisibles)
- Evaluación de costos, tiempos y cantidad de nodos (expandidos y frontera)



EJERCICIO

01



EJERCICIO 1

Se cuenta con el ejercicio “8-puzzle” donde se parte de un tablero inicial al azar y, moviendo los números adyacentes al espacio vacío, se busca llegar al tablero solución

Tablero inicial:

5	7	3
8	2	
1	6	4

Tablero solución:

1	2	3
8		4
7	6	5

EJERCICIO 1

¿QUÉ ESTRUCTURA DE DATOS VAMOS A UTILIZAR?

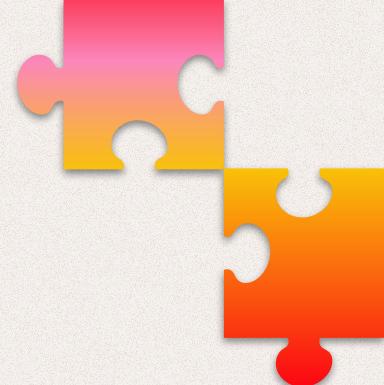
MATRIZ

- Almacenamiento
- Manipulación
- Implementación
- Control
- Visualización

1	2	3
4	5	6
7	8	

HEURISTICAS

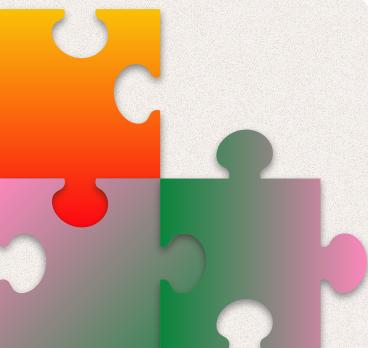
Las heurísticas deben ser **admisibles** y **no-triviales**



MANHATTAN

CHEBYSHEV

EUCLIDES

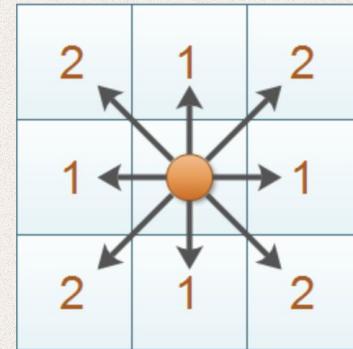


EJERCICIO 1

MANHATTAN

- Es la distancia de cada ficha al lugar correcto de la misma.
- Se calcula con la siguiente ecuación:

$$|x_1 - x_2| + |y_1 - y_2|$$



Tablero inicial:

5	7	3
8	2	
1	6	4

Tablero solución:

1	2	3
8		4
7	6	5

Manhattan:

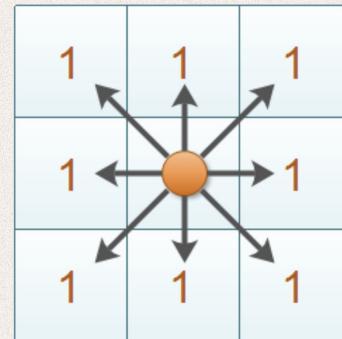
4	3	0
0	1	
2	0	1

$$h(t) = 11$$

EJERCICIO 1

CHEBYSHEV

- Mide la distancia de Chebyshev entre la posición actual de cada ficha y su posición correspondiente en la solución correcta.
- Es admisible
- La distancia se calcula : $\max(|x_i - x_2|, |y_i - y_2|)$



Tablero inicial:

5	7	3
8	2	
1	6	4

Tablero solución:

1	2	3
8		4
7	6	5

Chebyshev:

2	2	0
0	1	1
2	0	1

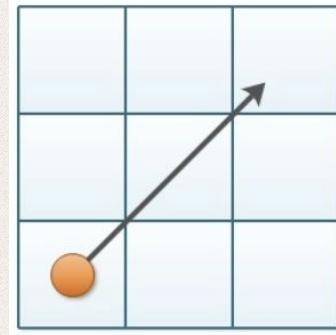
$$h(t) = 9$$

EJERCICIO 1

EUCLIDES

- Mide la distancia euclíadiana entre la posición actual de cada ficha y su posición correcta.
- Es admisible
- Se calcula aplicando el teorema de pitágoras:

$$\sqrt{(x_1 - x_2)^2 + (y_1 - y_2)^2}$$



Tablero inicial:

5	7	3
8	2	
1	6	4

Tablero solución:

1	2	3
8		4
7	6	5

Euclides:

2,83	2,24	0
0	1	1
2	0	1

$$h(t) = 10,07$$

Considera movimientos no permitidos en el juego por lo tanto no será seleccionado para el algoritmo de búsqueda

¿QUÉ MÉTODOS DE BÚSQUEDA UTILIZAREMOS?

A* Search



- Admisible
- Completa
- Solución óptima

¿POR QUÉ UN MÉTODO INFORMADO POR SOBRE UN DESINFORMADO?

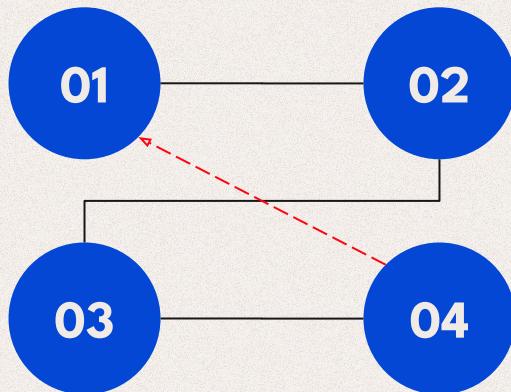
- Se hace uso del conocimiento de la situación final
- Consume menos tiempo y costo (*eficiencia*)
- Hace una búsqueda en profundidad heurística en primer lugar

EJERCICIO 1

A* Search

Elige el nodo a expandir que promete una mejor solución

Obtiene $f(n) = g(n) + h(n)$ como estimación total



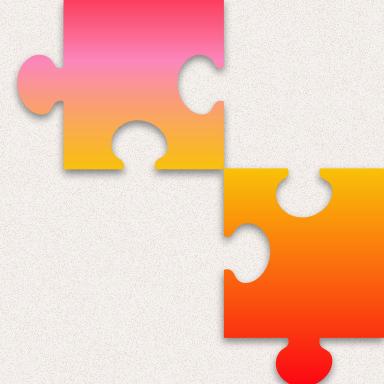
Para cada nodo frontera n conoce $g(n)$ y $h(n)$

Selecciona el nodo con menor valor $f(n)$

¿CON QUÉ HEURÍSTICAS?

Las mismas seleccionadas anteriormente.

Son heurísticas admisibles y consistentes

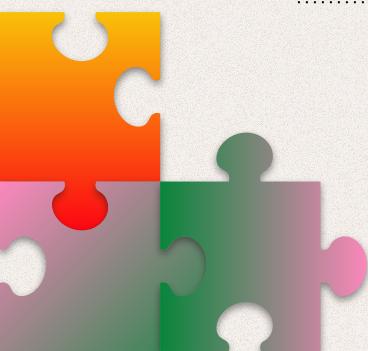


Manhattan

- Calculo sencillo
- Bajo costo computacional

Chebyshev

- Estimacion precisa
- Se basa en posiciones





EJERCICIO

02

EJERCICIO 2

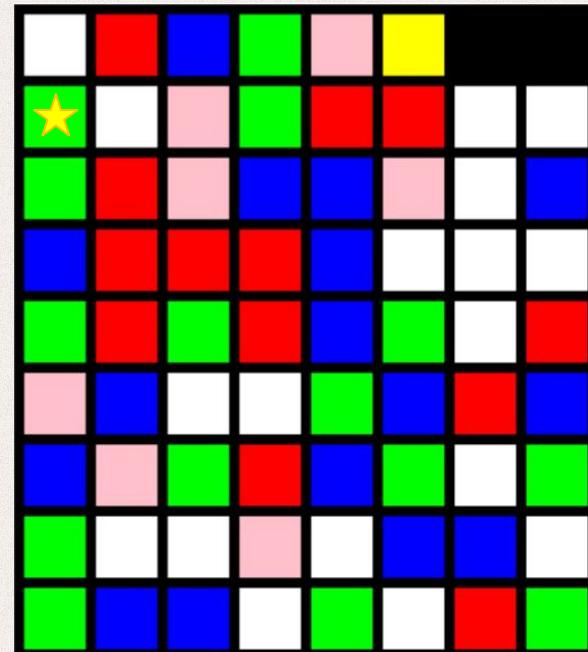
Estructura de Estado

Nos vamos a referir a:

- La ficha con estrella como **ficha principal**

Al conjunto de fichas adyacentes de un mismo color se los denomina:

- **Bloque principal** si incluyen a la ficha principal
- **Bloque** si no la incluyen
- **Barra de colores** a las fichas que refieren a los M colores posibles



EJERCICIO 2

Estructura de Estado

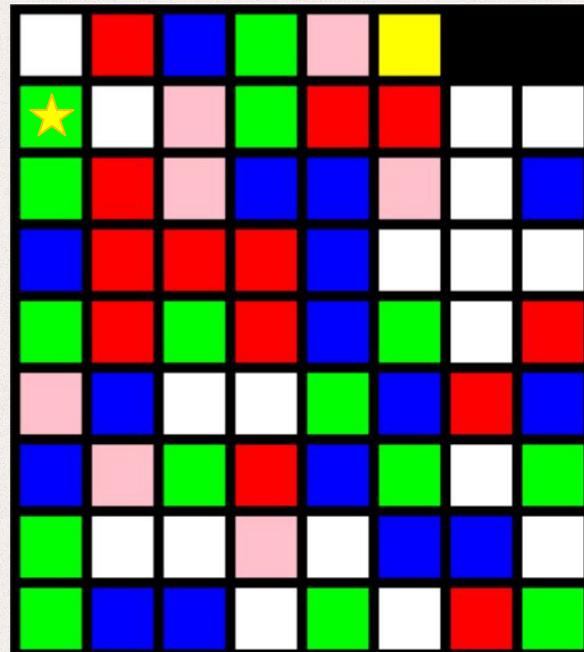
Se cuenta con una matriz de NxN con M posibles colores

Ejemplo: N=8 y M=6

Estado inicial: Matriz con un orden aleatorio de colores

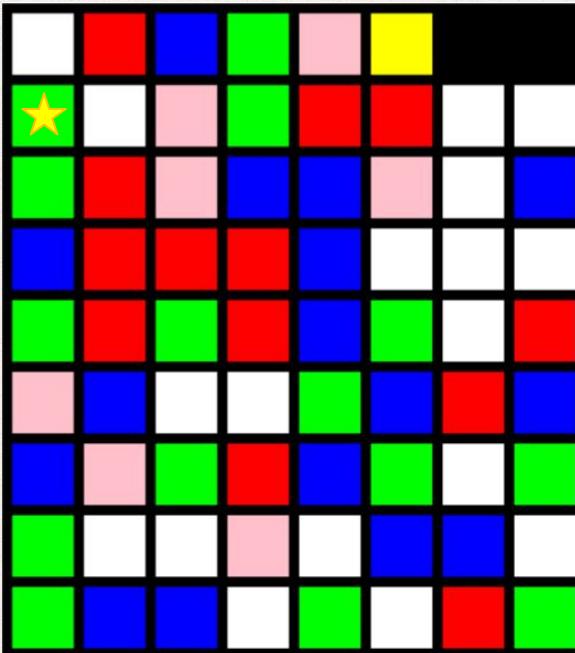
Punto de inicio: Se toma como partida la ficha principal

Costo: Se define un costo unitario ($C = 1$) para todos los movimientos



EJERCICIO 2

FILL ZONE



OBJETIVO:

- Que todo el tablero quede de un solo color

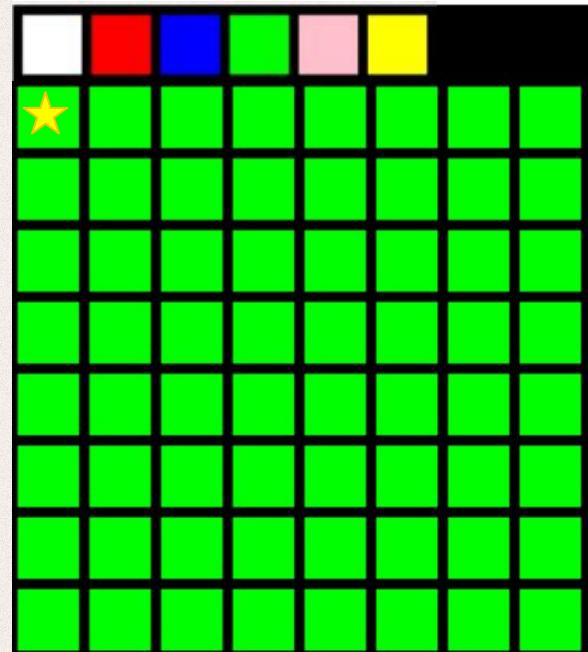
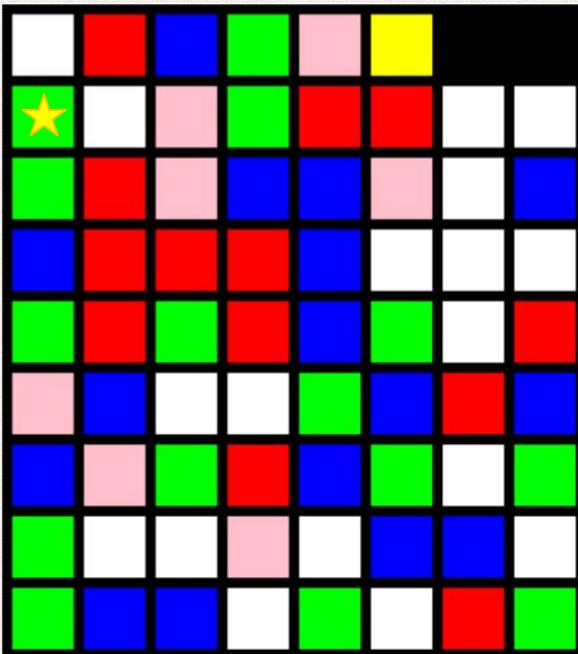
CÓMO SE JUEGA?

- Se selecciona 1 color de la barra de colores y se colorean todas las fichas del **bloque principal** con ese color. Así sucesivamente hasta colorear la matriz completa

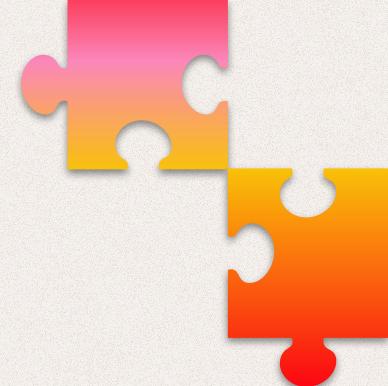
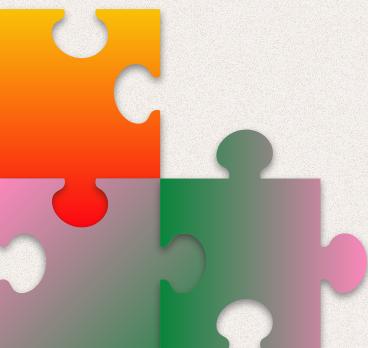
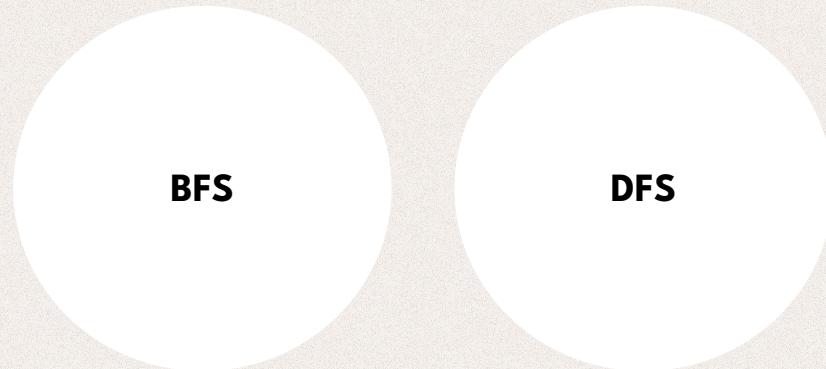


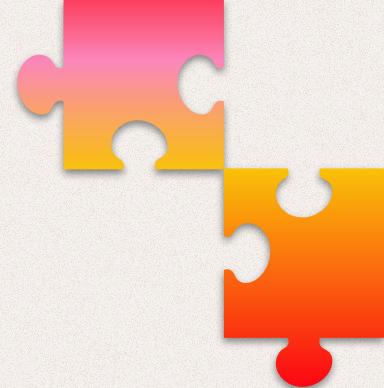
EJERCICIO 2

FILL ZONE



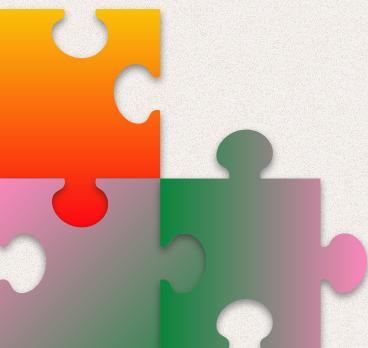
METODOS DE BUSQUEDA DESINFORMADOS

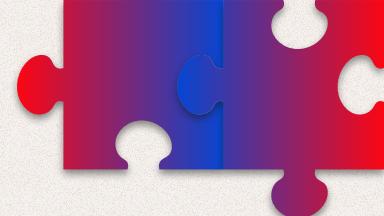




BFS - Breadth First Search

- Expande nodos de menor profundidad primero
- Encuentra la solución (es completa) si su factor de ramificación es finito
- Encuentra la solución con menor profundidad
- Es óptima cuando el problema es de costo uniforme





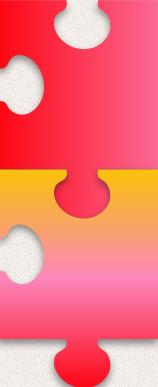
BFS - Breadth First Search

Trabajando con 6 colores, 50 iteraciones

NxN	Costo de Solucion	Nodos expandidos	Nodos Frontera	Tiempo de procesamiento (s)
4x4	$7 \pm 1,01$	$3553 \pm \emptyset$	$2071 \pm \emptyset$	$0,3 \pm \emptyset$
6x6	$9 \pm 0,96$	$426030 \pm \emptyset$	$303998 \pm \emptyset$	$61 \pm \emptyset$
8x8				



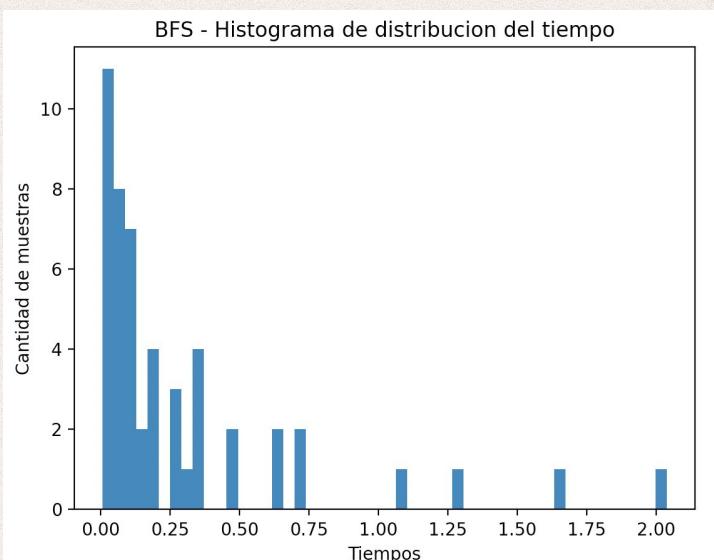
Comentarios acerca del algoritmo

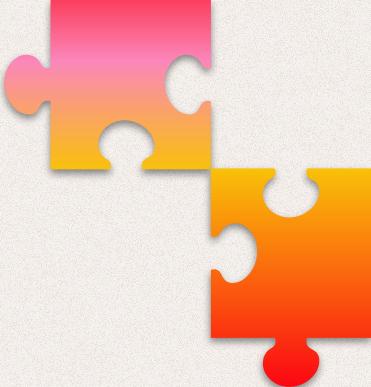


- No se tomó en cuenta el desvío estándar en BFS para calcular el error ya que no se encontró una distribución gaussiana en la cantidad de iteraciones definidas.

Ejemplo 4x4 con 6 colores:

Esto se extiende a las pruebas con variación de colores

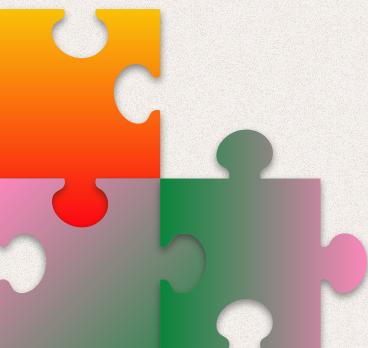
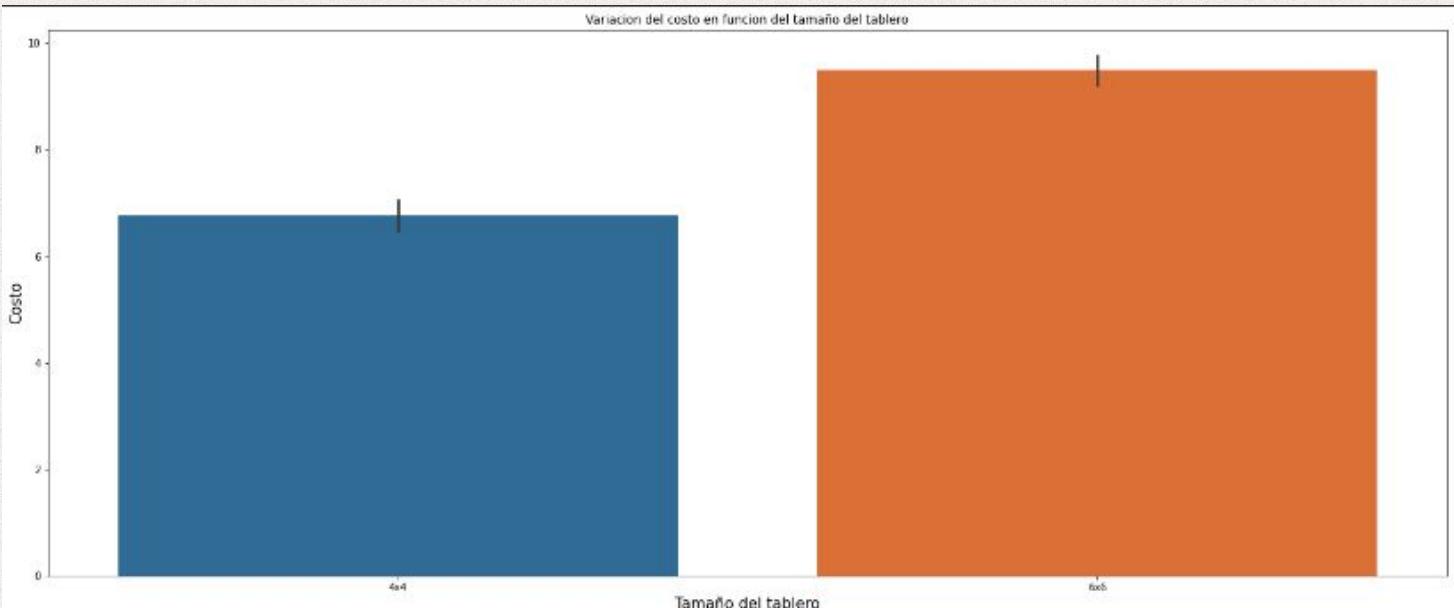


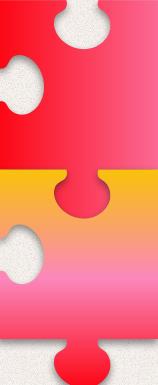


BFS - Breadth First Search

Boxplot:

- Mediante 50 pruebas de cada caso planteado en el cuadro anterior obtenemos:





BFS - Breadth First Search

Trabajando con un tablero de 4x4, 50 iteraciones

M	Costo de Solucion	Nodos expandidos	Nodos Frontera	Tiempo de procesamiento
3	$4 \pm 0,69$	$21 \pm \emptyset$	$7 \pm \emptyset$	$0,00108 \pm \emptyset$
6	$7 \pm 1,01$	$3553 \pm \emptyset$	$2071 \pm \emptyset$	$0,3 \pm \emptyset$
8	$8 \pm 0,97$	$26294 \pm \emptyset$	$16248 \pm \emptyset$	$2,91621 \pm \emptyset$



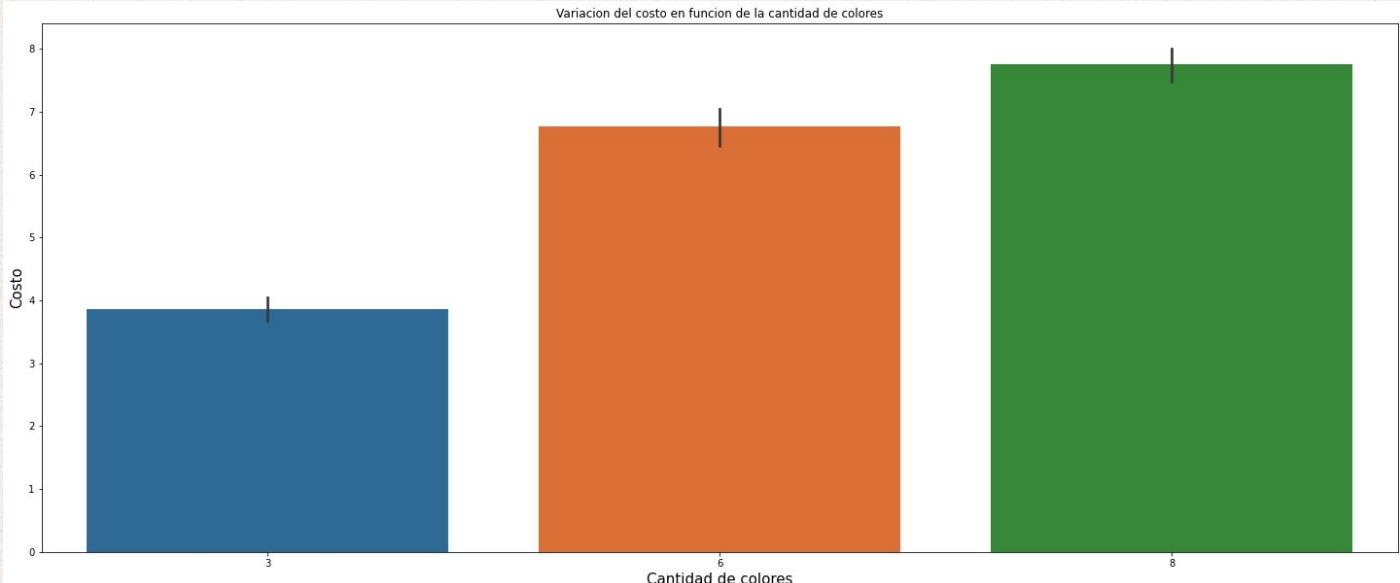
EJERCICIO 2

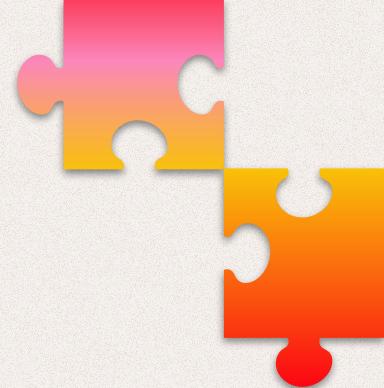
BFS - Breadth First Search

Boxplot:

- Mediante 50 pruebas de cada caso planteado en el cuadro anterior obtenemos:

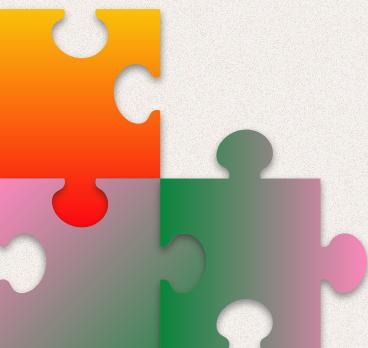
Tablero de 4x4

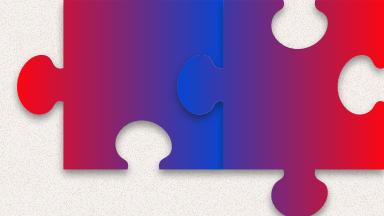




DFS - Depth First Search

- Comienza en el nodo raíz y expande el nodo de mayor profundidad.
- Explora en la medida de lo posible a lo largo de cada rama antes del retroceso.
- Es completa cuando se usa búsqueda basada en grafos finitos.
- No es óptima.





DFS - Depth First Search

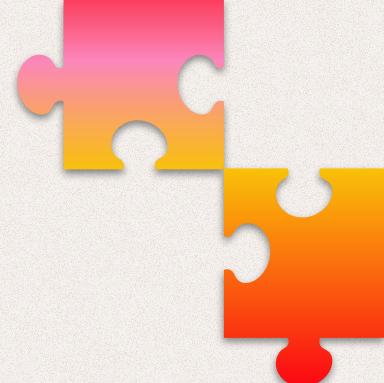
Trabajando con 6 colores, 50 iteraciones

NxN	Costo de Solucion	Nodos expandidos	Nodos Frontera	Tiempo de procesamiento (s)
4x4	$8 \pm 1,39$	$9 \pm 1,39$	$4 \pm 0,848$	$0,00157 \pm 0,001$
6x6	$15 \pm 2,53$	$16 \pm 2,53$	$7 \pm 1,27$	$0,00545 \pm 0,003$
8x8	$22 \pm 2,96$	$23 \pm 2,96$	$10 \pm 1,46$	$0,00778 \pm 0,002$



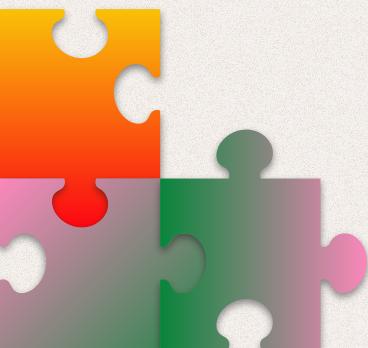
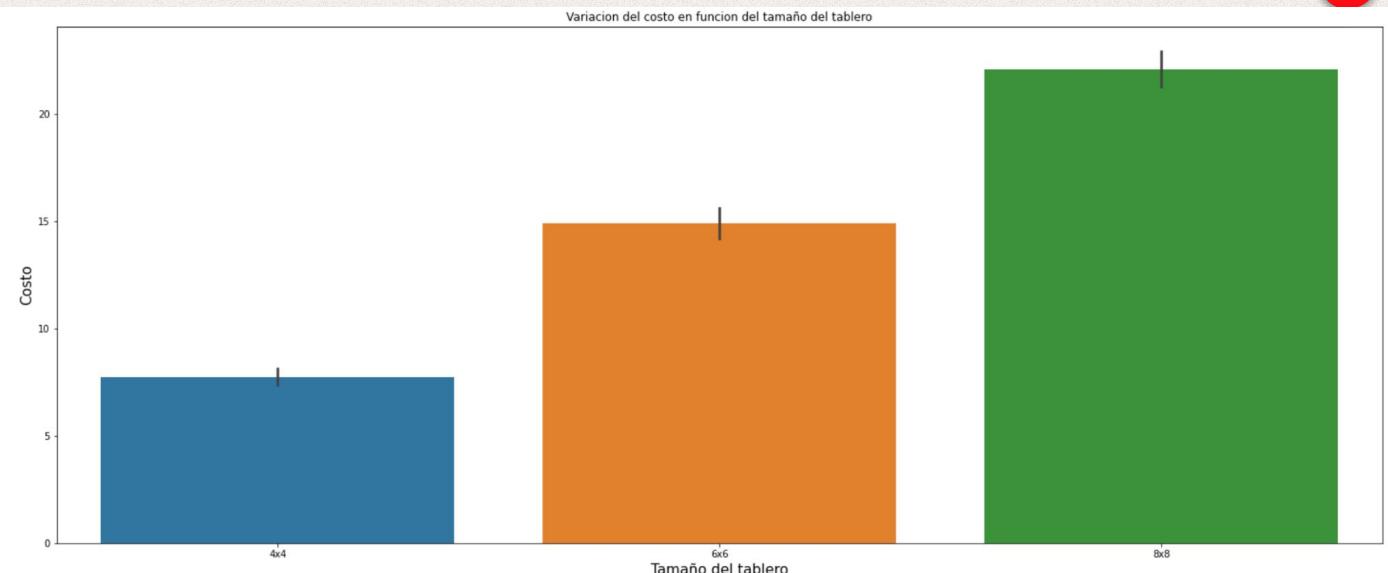
EJERCICIO 2

DFS - Depth First Search



Boxplot:

- Mediante 50 pruebas de cada caso planteado en el cuadro anterior obtenemos:



EJERCICIO 2

DFS - Depth First Search

Trabajando con un tablero de 4x4, 50 iteraciones

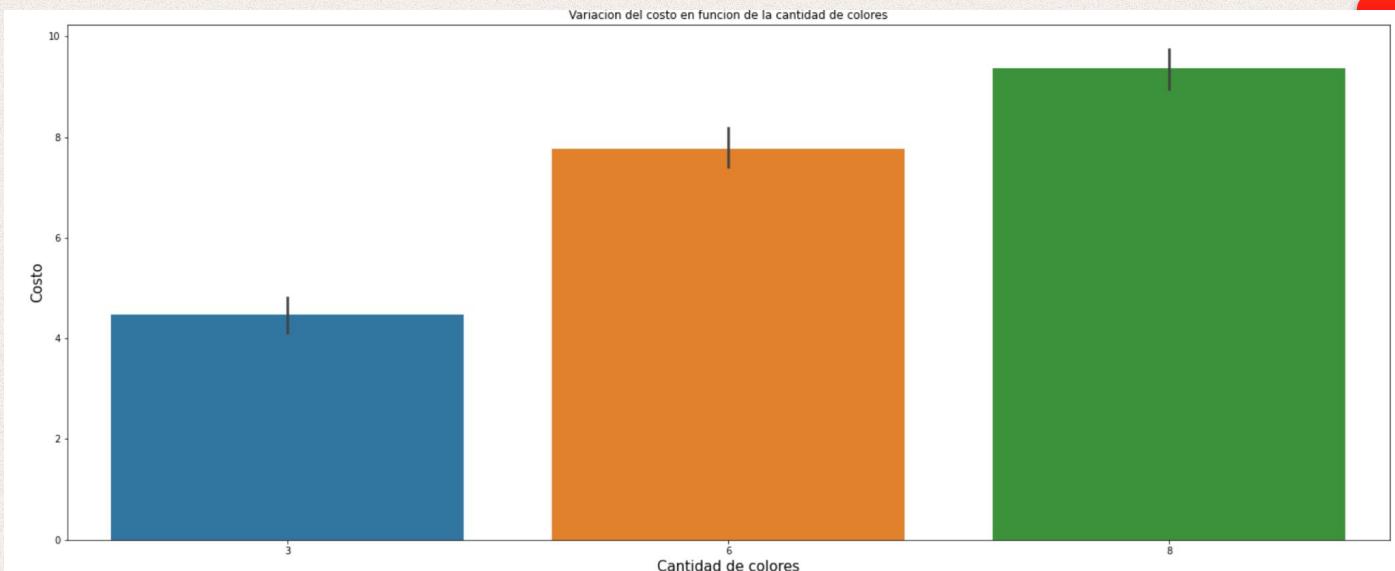
M	Costo de Solucion	Nodos expandidos	Nodos Frontera	Tiempo de procesamiento
3	$5 \pm 1,31$	$6 \pm 1,31$	$3 \pm 0,72$	$0,00055 \pm 0,006$
6	$8 \pm 1,39$	$9 \pm 1,39$	$7 \pm 1,27$	$0,00157 \pm 0,001$
8	$9 \pm 1,46$	$10 \pm 1,46$	$5 \pm 1,5$	$0,00223 \pm 0,001$

EJERCICIO 2

DFS - Depth First Search

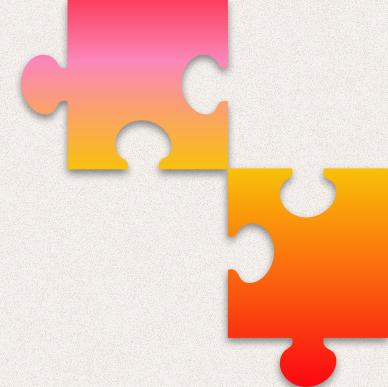
Boxplot:

- Mediante 50 pruebas de cada caso planteado en el cuadro anterior obtenemos:



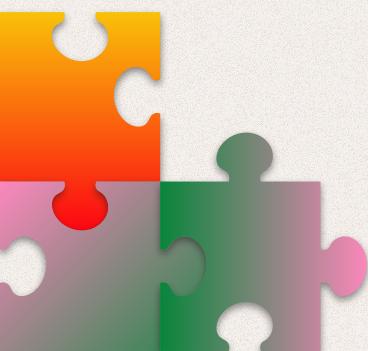
HEURISTICAS

Admisibles



Heurística 1

Cantidad de colores restantes

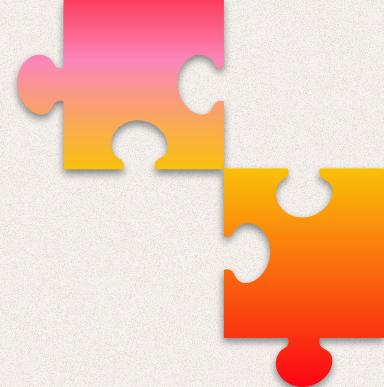


Heurística 2

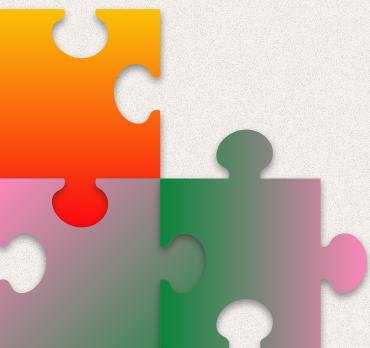
Cantidad de bloques adyacentes
de igual color distinto a sí mismo

HEURISTICAS

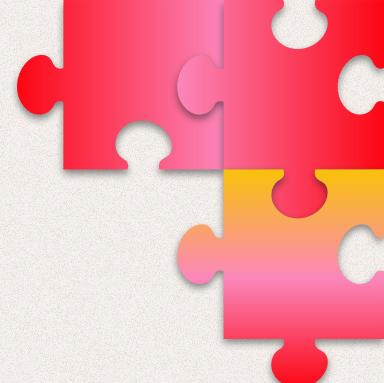
No admisibles



Heurística 3
Cantidad de bloques restantes



METODOS DE BUSQUEDA INFORMADOS



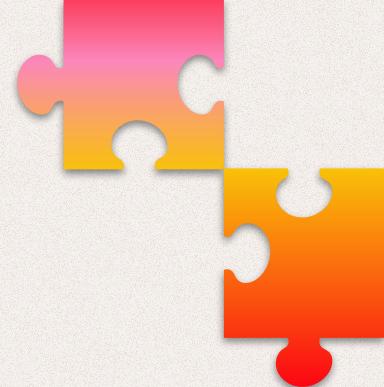
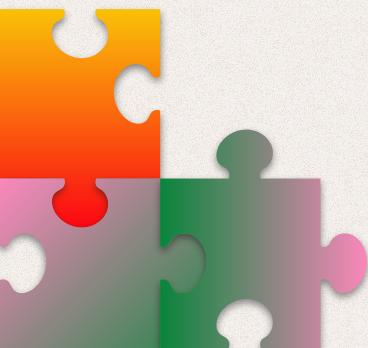
GREEDY

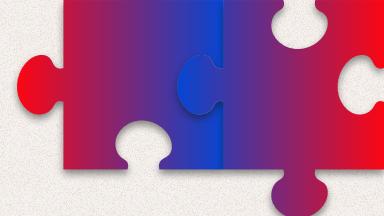
A* SEARCH



LOCAL GREEDY SEARCH

- Comienza desde el nodo raíz.
- Baja a los hijos y calcula su heurística. Se queda con el hijo que tenga menor valor heurístico y expande.
- No es óptima.
- No es completa si no evalúa repetición de estados.
- Veloz.



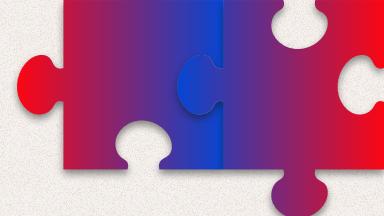


LOCAL GREEDY SEARCH

Trabajando con 6 colores, 50 iteraciones

HEURISTICA 1				
NxN	Costo de Solucion	Nodos expandidos	Nodos Frontera	Tiempo de procesamiento (s)
4x4	$7 \pm 1,28$	$18 \pm 4,14$	$10 \pm 3,08$	$0,00309 \pm 0,001$
6x6	$13 \pm 2,38$	$38 \pm 7,66$	$24 \pm 5,64$	$0,01193 \pm 0,004$
8x8	$20 \pm 3,23$	$66 \pm 11,28$	$45 \pm 8,36$	$0,02166 \pm 0,006$



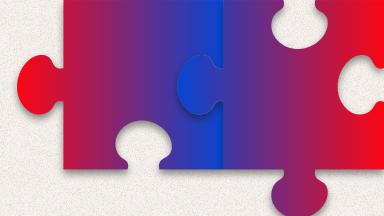


LOCAL GREEDY SEARCH

Trabajando con 6 colores, 50 iteraciones

HEURISTICA 2				
NxN	Costo de Solucion	Nodos expandidos	Nodos Frontera	Tiempo de procesamiento (s)
4x4	$8 \pm 1,39$	$21 \pm 5,32$	$12 \pm 4,16$	$0,00314 \pm 0,001$
6x6	$15 \pm 2,56$	$46 \pm 8,23$	$30 \pm 6,15$	$0,01358 \pm 0,007$
8x8	$22 \pm 3,2$	$75 \pm 10,57$	$52 \pm 7,78$	$0,02512 \pm 0,009$





LOCAL GREEDY SEARCH

Trabajando con 6 colores, 50 iteraciones

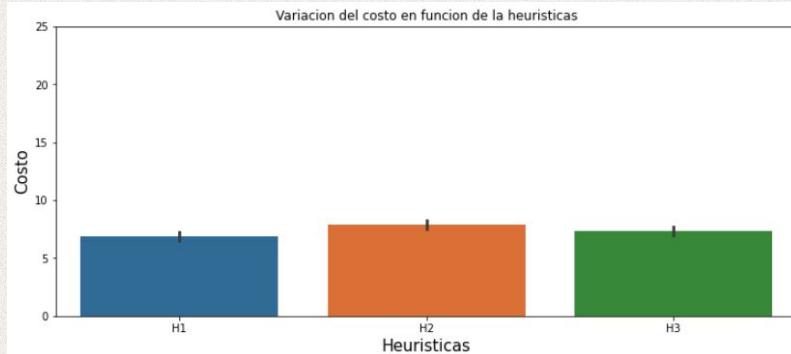
HEURISTICA 3				
NxN	Costo de Solucion	Nodos expandidos	Nodos Frontera	Tiempo de procesamiento (s)
4x4	$8 \pm 1,3$	$21 \pm 5,43$	$13 \pm 4,24$	$0,00307 \pm 0,001$
6x6	$12 \pm 1,86$	$37 \pm 7,77$	$25 \pm 6,04$	$0,00959 \pm 0,004$
8x8	$16 \pm 1,87$	$58 \pm 8,4$	$41 \pm 6,68$	$0,01565 \pm 0,005$



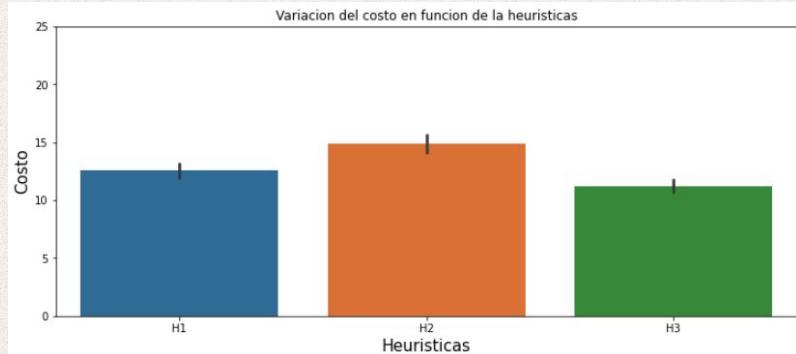
EJERCICIO 2

Trabajando con 6 colores, observamos la variación del costo en función de las heurísticas

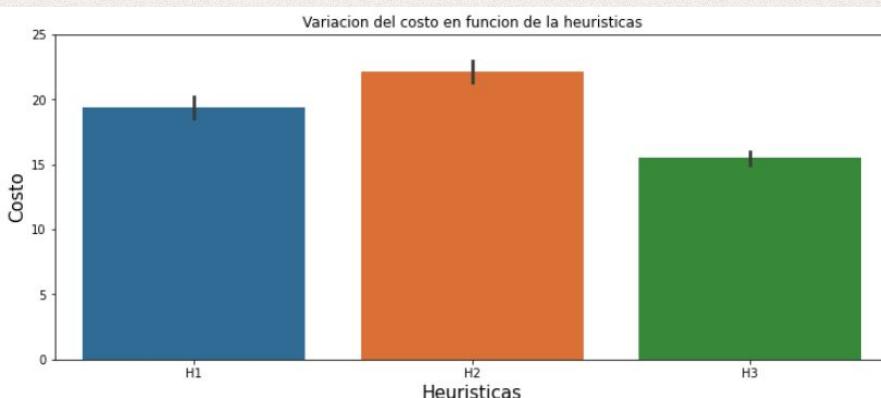
Tablero de 4x4

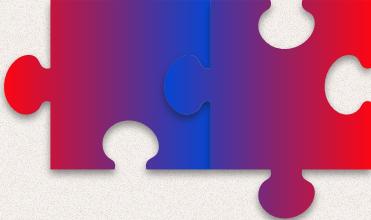


Tablero de 6x6



Tablero de 8x8



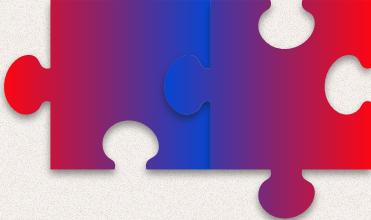


LOCAL GREEDY

Trabajando con un tablero de 4x4, 50 iteraciones

HEURISTICA 1				
M	Costo de Solucion	Nodos expandidos	Nodos Frontera	Tiempo de procesamiento
3	4 ± 1,1	8 ± 1,99	3 ± 1,004	0,00135 ± 0,0009
6	7 ± 1,28	18 ± 4,14	10 ± 3,08	0,00309 ± 0,001
8	8 ± 1,1	22 ± 4,2	13 ± 3,26	0,00547 ± 0,002





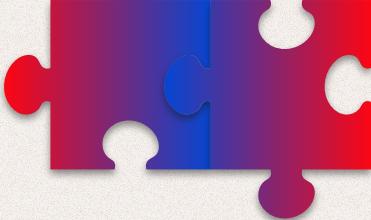
LOCAL GREEDY

Trabajando con un tablero de 4x4, 50 iteraciones

HEURISTICA 2				
M	Costo de Solucion	Nodos expandidos	Nodos Frontera	Tiempo de procesamiento
3	$5 \pm 1,24$	$8 \pm 1,13$	$3 \pm 2,28$	$0,00106 \pm 0,0006$
6	$8 \pm 1,39$	$21 \pm 5,32$	$12 \pm 4,16$	$0,00314 \pm 0,001$
8	$10 \pm 1,52$	$38 \pm 5,16$	$17 \pm 6,4$	$0,00520 \pm 0,002$



EJERCICIO 2



LOCAL GREEDY

Trabajando con un tablero de 4x4, 50 iteraciones

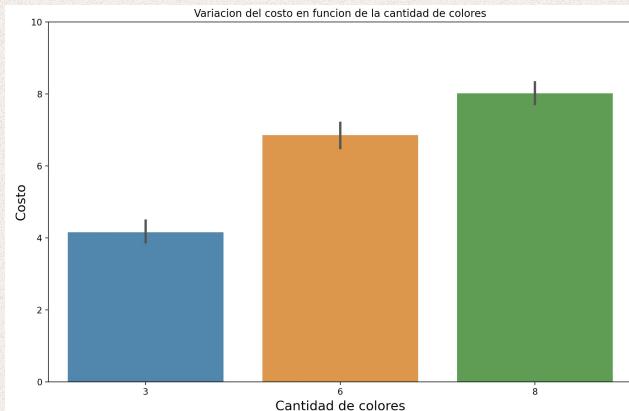
HEURISTICA 3				
M	Costo de Solucion	Nodos expandidos	Nodos Frontera	Tiempo de procesamiento
3	$4 \pm 0,91$	$8 \pm 0,94$	$3 \pm 1,74$	$0,00091 \pm 0,0006$
6	$8 \pm 1,3$	$21 \pm 5,43$	$13 \pm 4,24$	$0,00307 \pm 0,001$
8	$9 \pm 1,45$	$27 \pm 5,14$	$17 \pm 6,48$	$0,00473 \pm 0,001$



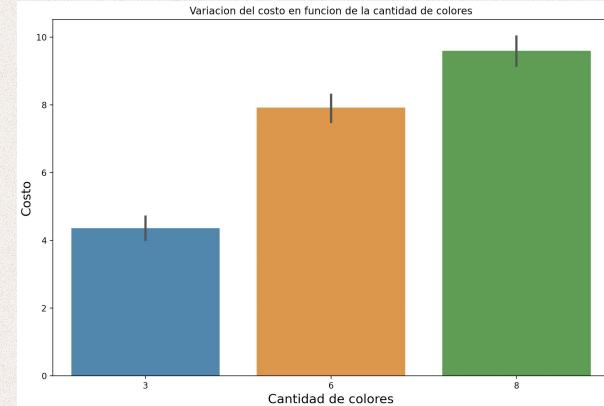
EJERCICIO 2

Trabajando con un tablero de 4x4, observamos la variación del costo en función de la cantidad de colores

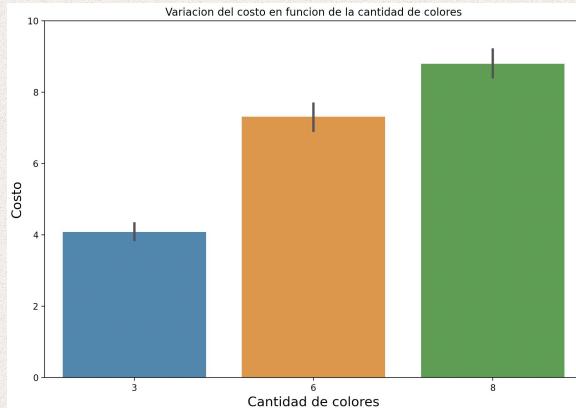
Heuristica 1



Heuristica 2

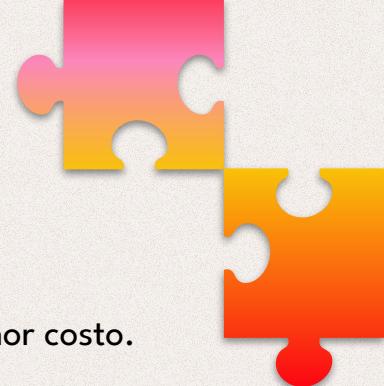
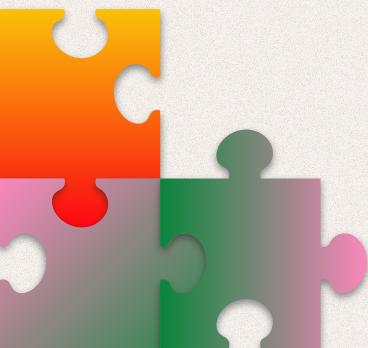


Heuristica 3



A * SEARCH

- Comienza en un nodo específico y busca encontrar el nodo final teniendo el menor costo.
- Mantiene un árbol de rutas que se extiende
- Si la heurística que se usa es admisible, garantiza devolver una ruta de menor costo de principio a fin.
- Puede ser óptima.
- Puede ser completo.



EJERCICIO 2

A * SEARCH

Trabajando con 6 colores, 50 iteraciones

HEURISTICA 1				
NxN	Costo de Solucion	Nodos expandidos	Nodos Frontera	Tiempo de procesamiento (s)
4x4	7 ± 0,85	197 ± ∅	115 ± ∅	0,05731 ± ∅
6x6	10 ± 0,93	1075 ± ∅	738 ± ∅	0,38402 ± ∅
8x8	13 ± 1,06	27403 ± ∅	20691 ± ∅	116,03654 ± ∅

EJERCICIO 2

A * SEARCH

Trabajando con 6 colores, 50 iteraciones

HEURISTICA 2				
NxN	Costo de Solucion	Nodos expandidos	Nodos Frontera	Tiempo de procesamiento (s)
4x4	3 ± 1,54	66 ± ∅	43 ± ∅	0,01119 ± ∅
6x6	4 ± 2,25	1792 ± ∅	1329 ± ∅	2,37729 ± ∅
8x8	5 ± 2,59	16470 ± ∅	12536 ± ∅	253,15731 ± ∅

EJERCICIO 2

A * SEARCH

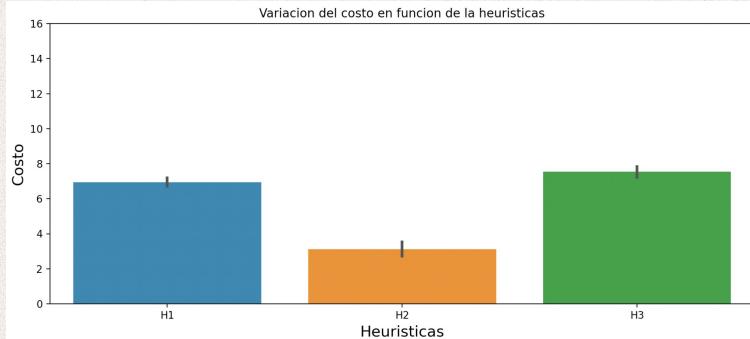
Trabajando con 6 colores, 50 iteraciones

HEURISTICA 3				
NxN	Costo de Solucion	Nodos expandidos	Nodos Frontera	Tiempo de procesamiento (s)
4x4	8 ± 1,13	35 ± ∅	22 ± ∅	0,00663 ± ∅
6x6	12 ± 1,43	42 ± ∅	30 ± ∅	0,00801 ± ∅
8x8	15 ± 1,88	58 ± ∅	42 ± ∅	0,01154 ± ∅

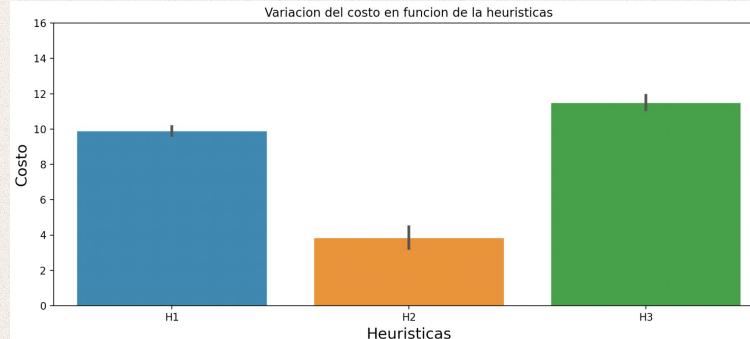
EJERCICIO 2

Trabajando con 6 colores, observamos la variación del costo en función de las heurísticas

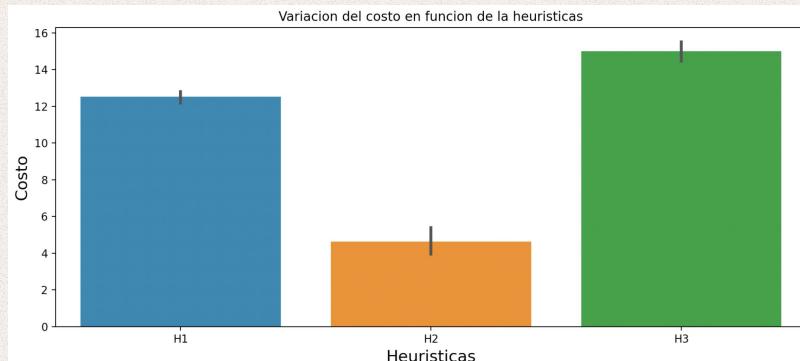
Tablero de 4x4



Tablero de 6x6



Tablero de 8x8



EJERCICIO 2

A * SEARCH

Trabajando con un tablero de 4x4, 50 iteraciones

HEURISTICA 1				
M	Costo de Solucion	Nodos expandidos	Nodos Frontera	Tiempo de procesamiento
3	$4 \pm 1,019$	$11 \pm \emptyset$	$5 \pm \emptyset$	$0,00178 \pm \emptyset$
6	$7 \pm 0,85$	$197 \pm \emptyset$	$115 \pm \emptyset$	$0,05731 \pm \emptyset$
8	$8 \pm 0,87$	$717 \pm \emptyset$	$412 \pm \emptyset$	$0,35447 \pm \emptyset$

EJERCICIO 2

A * SEARCH

Trabajando con un tablero de 4x4, 50 iteraciones

HEURISTICA 2				
M	Costo de Solucion	Nodos expandidos	Nodos Frontera	Tiempo de procesamiento
3	$3 \pm 1,36$	$8 \pm \emptyset$	$4 \pm \emptyset$	$0,00125 \pm \emptyset$
6	$3 \pm 1,54$	$66 \pm \emptyset$	$43 \pm \emptyset$	$0,01119 \pm \emptyset$
8	$3 \pm 0,217$	$135 \pm \emptyset$	$88 \pm \emptyset$	$0,03696 \pm \emptyset$

EJERCICIO 2

A * SEARCH

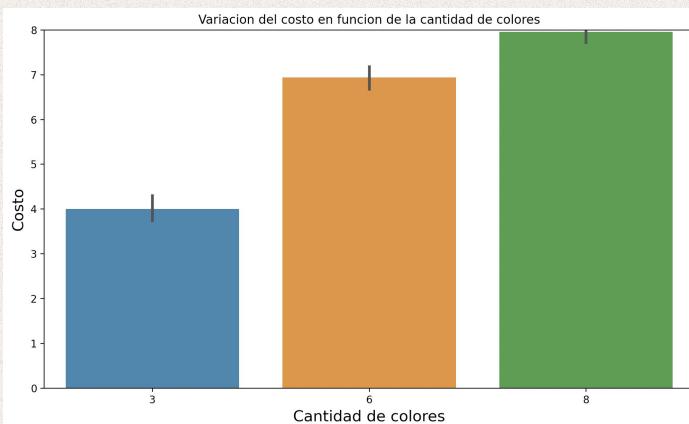
Trabajando con un tablero de 4x4, 50 iteraciones

HEURISTICA 3				
M	Costo de Solucion	Nodos expandidos	Nodos Frontera	Tiempo de procesamiento
3	$4 \pm 1,05$	$7 \pm \emptyset$	$3 \pm \emptyset$	$0,00097 \pm \emptyset$
6	$8 \pm 1,13$	$35 \pm \emptyset$	$22 \pm \emptyset$	$0,00663 \pm \emptyset$
8	$9 \pm 1,25$	$146 \pm \emptyset$	$87 \pm \emptyset$	$0,02819 \pm \emptyset$

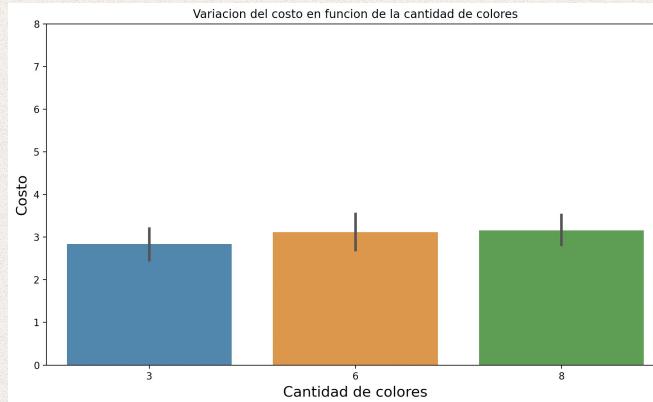
EJERCICIO 2

Trabajando con un tablero de 4x4, observamos la variación del costo en función de la cantidad de colores

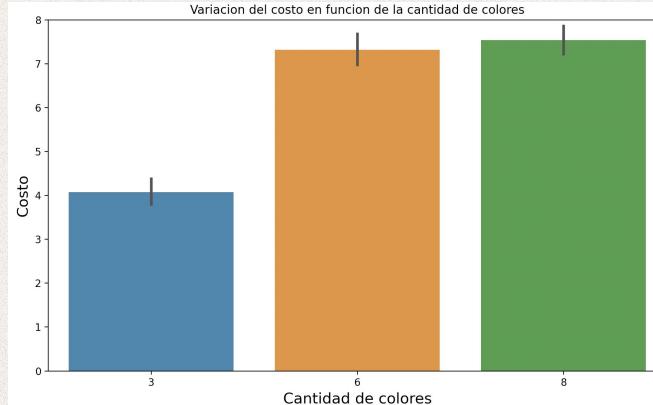
Heuristica 1



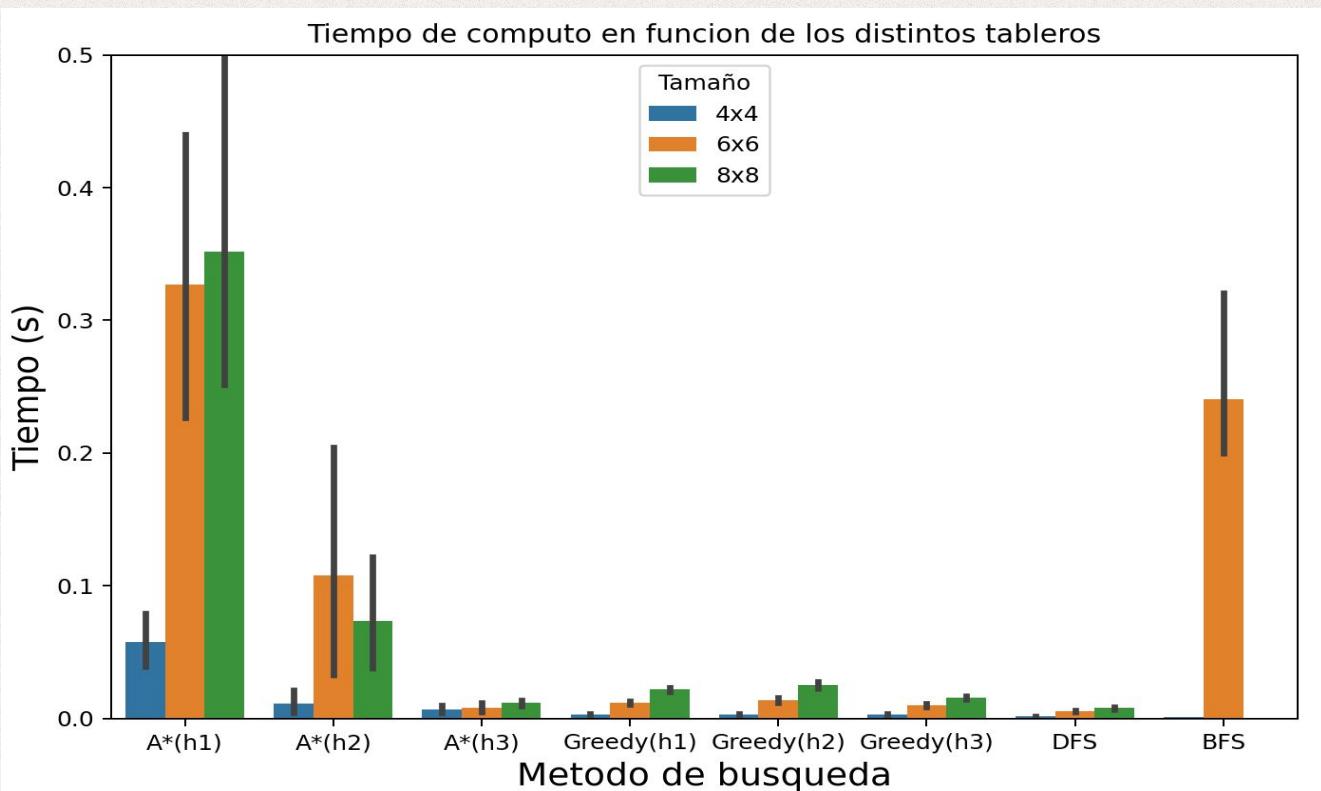
Heuristica 2

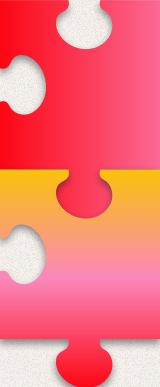


Heuristica 3



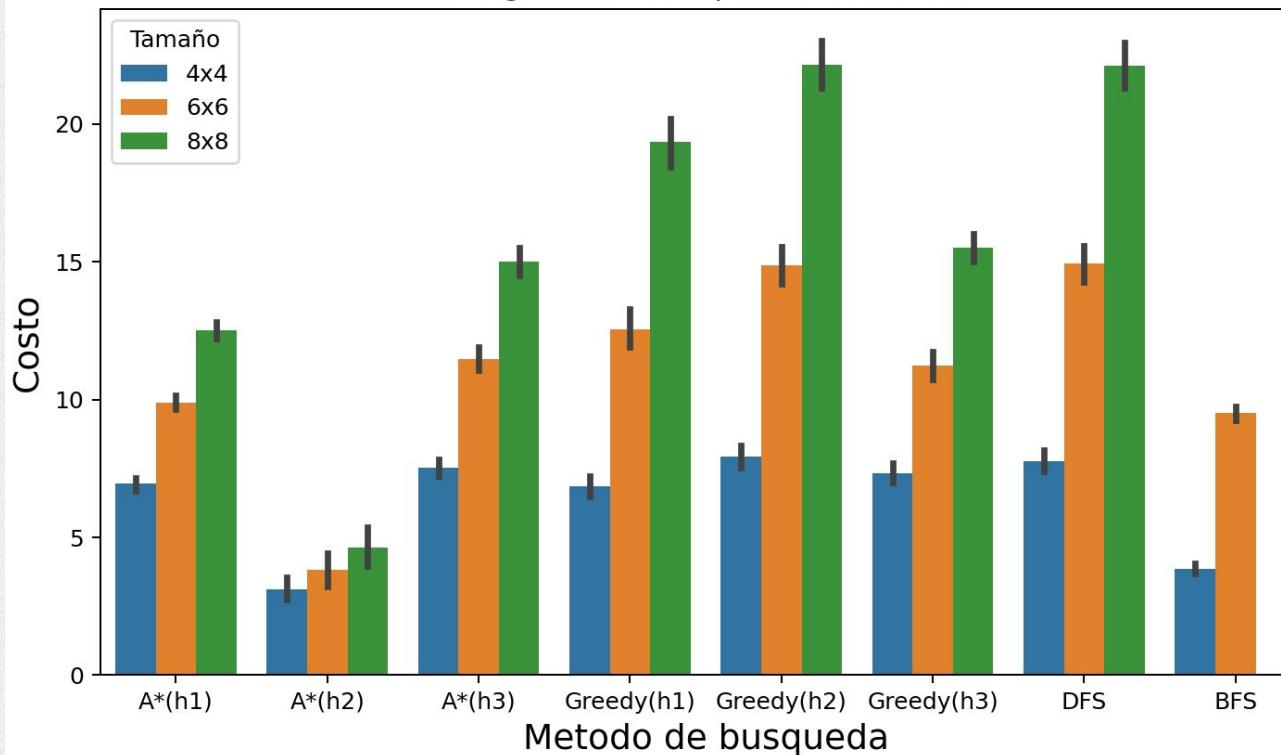
CONCLUSIONES





CONCLUSIONES

Costo segun el metodo para distintos tableros



GRACIAS!