



A^*



Ayudantía 05





Heurísticas

Función Heurística

Es una función que estima que tan cerca está un estado s de un estado final G .

Es un “atajo” para resolver problemas cuando no hay soluciones exactas o el tiempo de obtención de esta solución es muy largo.

Se utiliza para aproximar, ya que si indicara la distancia real al objetivo, el problema estaría resuelto.



Admisibilidad

$H(n)$ es admisible si no sobreestima el costo real para llegar al objetivo, es decir :

- $h(n) \leq H(n) \forall n$, donde $H(n)$ es el costo real



Admisibilidad

$H(n)$ es admisible si no sobreestima el costo real para llegar al objetivo, es decir :

- $h(n) \leq H(n) \quad \forall n$, donde $H(n)$ es el costo real

I	G=1		
F			

	G=1	H=7	
	H=3		

	G=1	H=8	
	H=9		

Admisibilidad

$H(n)$ es admisible si no sobreestima el costo real para llegar al objetivo, es decir :

- $h(n) \leq H(n) \forall n$, donde $H(n)$ es el costo real

I	G=1		
F			

	G=1	H=7	
	H=3		

	G=1	H=8	
	H=9		



Consistencia

$H(n)$ es consistente si el valor de la heurística para los vecinos de cierto nodo no es mayor al valor de la heurística en el nodo más el costo por llegar al vecino, es decir :

- $h(n) \leq C(n, a, n') + h(n') \quad \forall n' \text{ vecino de } n$, donde $C(n, a, n')$ es el costo para llegar del nodo n al nodo n'
- $h(n) = 0 \quad \forall n \in G$, es decir, n es un objetivo



Consistencia

Reemplazando la condición de consistencia en $F(n')$:

- $F(n') = G(n') + H(n')$
- $F(n') = G(n) + C(n, \alpha, n') + H(n')$
- $F(n') \geq G(n) + H(n) = F(n)$
- $F(n') \geq F(n)$

, de esta forma tenemos que $F(n)$ es una función no decreciente.



Distancia Manhattan y Euclidiana

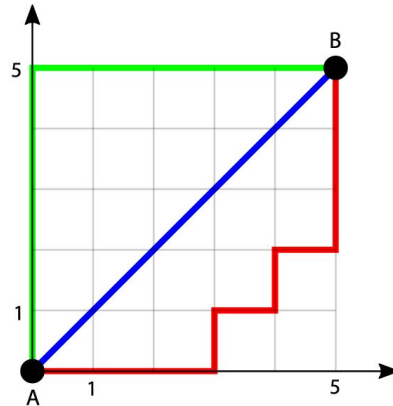
La distancia entre dos puntos es:

- Manhattan: la suma de las diferencias absolutas de sus coordenadas

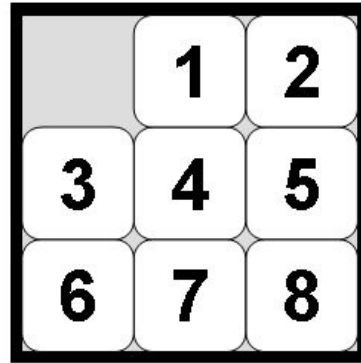
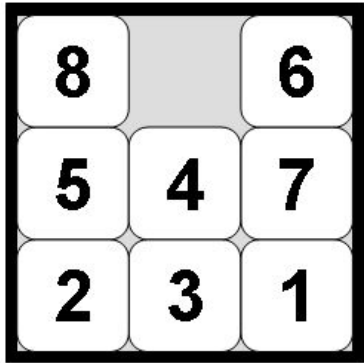
$$D_{\text{Manhattan}} = |x_1 - x_2| + |y_1 - y_2|$$

- Euclidiana: distancia en línea recta entre los puntos

$$D_{\text{Euclidiana}} = \sqrt{[(x_1 - x_2)^2 + (y_1 - y_2)^2]}$$



Relajación de problemas





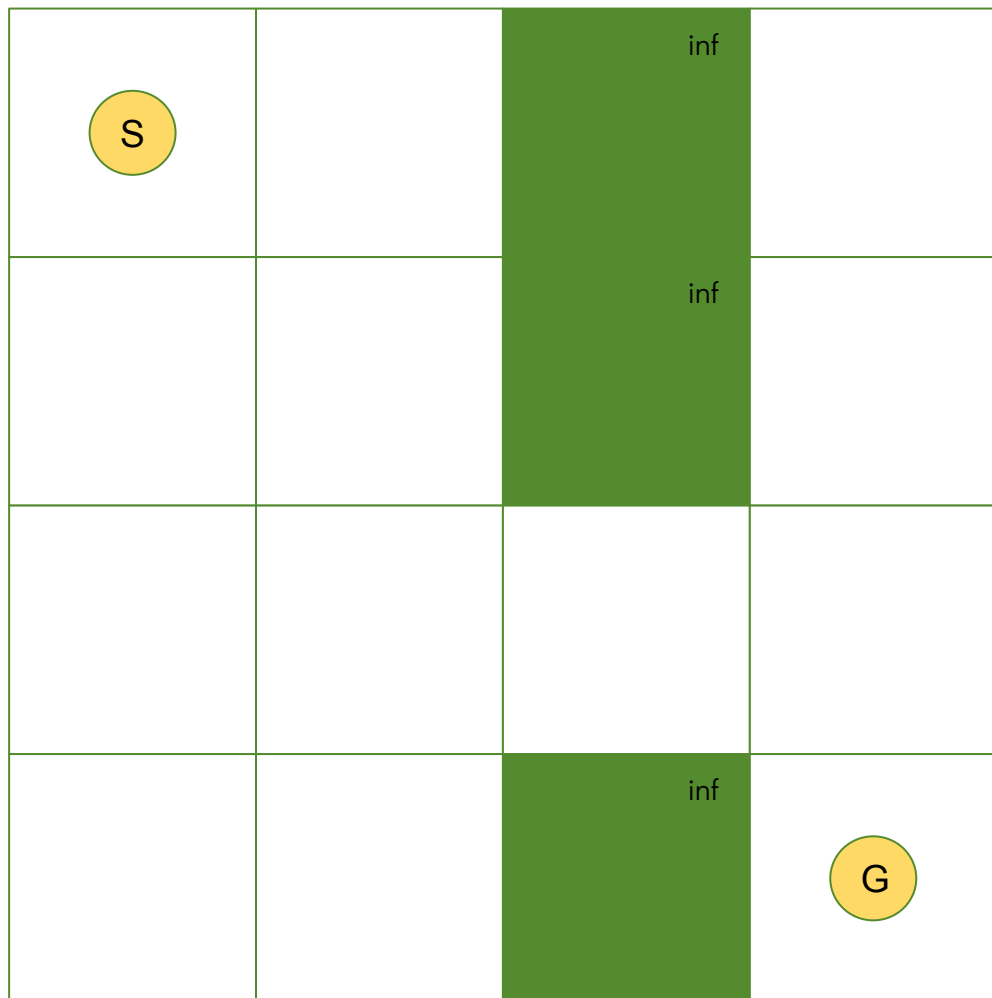
Algoritmo A*

Implementación en PyGame



Pseudocódigo

- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$



1 for each $s \in S$ do $g(s) \leftarrow \infty$

2 $Open \leftarrow \{s_0\}$

3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$

4 while $Open \neq \emptyset$

5 Extrae un u desde $Open$ con menor valor- f

6 if u es objetivo return u

7 for each $v \in Succ(u)$ do

1 $cost_v = g(u) + c(u, v)$

2 if $cost_v \geq g(v)$ return



3 $parent(v) \leftarrow u$

4 $g(v) \leftarrow cost_v$

5 $f(v) \leftarrow g(v) + h(v)$

6 if $v \in Open$ then Reordenar $Open$

7 else Insertar v en $Open$

inf  6	inf 5	inf	inf 3
inf 5	inf 4	inf	inf 2
inf 4	inf 3	inf 2	inf 1
inf 3	inf 2	inf	 inf

1 for each $s \in S$ do $g(s) \leftarrow \infty$

2 $Open \leftarrow \{s_0\}$

3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$

4 while $Open \neq \emptyset$

5 Extrae un u desde $Open$ con menor valor- f

6 if u es objetivo return u

7 for each $v \in Succ(u)$ do

 1 $cost_v = g(u) + c(u, v)$

 2 if $cost_v \geq g(v)$ return



 3 $parent(v) \leftarrow u$

 4 $g(v) \leftarrow cost_v$



 5 $f(v) \leftarrow g(v) + h(v)$

 6 if $v \in Open$ then Reordenar $Open$



 7 else Insertar v en $Open$

inf  6	inf 5	inf	inf 3
inf 5	inf 4	inf	inf 2
inf 4	inf 3	inf 2	inf 1
inf 3	inf 2	inf	 inf



- 1 for each $s \in S$ do $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0; f(s_0) \leftarrow h(s_0)$
- 4 while $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 if u es objetivo return u
 - 7 for each $v \in Succ(u)$ do
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 if $cost_v \geq g(v)$ return
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 if $v \in Open$ then Reordenar $Open$
 - 7 else Insertar v en $Open$

6	0	inf	inf	inf
	6	5		3
	inf	inf	inf	inf
	5	4		2
	inf	inf	inf	inf
	4	3	2	1
	inf	inf	inf	inf
	3	2		



- 1 for each $s \in S$ do $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 while $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 if u es objetivo return u
- 7 for each $v \in Succ(u)$ do
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 if $cost_v \geq g(v)$ return
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 if $v \in Open$ then Reordenar $Open$
 - 7 else Insertar v en $Open$

6	0	inf	inf	inf
	6	5		3
	inf	inf	inf	inf
	5	4		2
	inf	inf	inf	inf
	4	3	2	1
	inf	inf	inf	inf
	3	2		



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 **Extrae un u desde $Open$ con menor valor- f**
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	inf	inf	inf
	6	5		3
	inf	inf	inf	inf
	5	4		2
	inf	inf	inf	inf
	4	3	2	1
	inf	inf	inf	inf
	3	2		



- 1 for each $s \in S$ do $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 while $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if u es objetivo return u**
- 7 for each $v \in Succ(u)$ do
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if $cost_v \geq g(v)$ return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if $v \in Open$ then Reordenar $Open$**
 - 7 **else Insertar v en $Open$**

6	0	inf	inf	inf
	6	5		3
	inf	inf	inf	inf
	5	4		2
	inf	inf	inf	inf
	4	3	2	1
	inf	inf	inf	inf
	3	2		



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	inf	inf	inf
	6	5		3
	inf	inf	inf	inf
	5	4		2
	inf	inf	inf	inf
	4	3	2	1
	inf	inf	inf	inf
	3	2		

- 1 **for each** $s \in \mathcal{S}$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	inf	inf	inf
	6	5		3
	inf	inf	inf	inf
	5	4		2
	inf	inf	inf	inf
	4	3	2	1
	inf	inf	inf	inf
	3	2		

- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	inf	inf	inf
	6	5		3
	inf	inf	inf	inf
	5	4		2
	inf	inf	inf	inf
	4	3	2	1
	inf	inf	inf	inf
	3	2		

- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	inf	inf	inf
S	6	(0,0)	5	3
	inf	inf	inf	inf
	5	4		2
	inf	inf	inf	inf
	4	3	2	1
	inf	inf	inf	inf
	3	2		G



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	1	inf	inf
S	6	(0,0)	5	3
inf	inf	inf	inf	inf
5	4			2
inf	inf	inf	inf	inf
4	3	2	1	
inf	inf	inf	inf	inf
3	2			G



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
S	6	(0,0)	5		3
	inf	inf	inf	inf	inf
	5	4			2
	inf	inf	inf	inf	inf
	4	3	2		1
	inf	inf	inf	inf	inf
	3	2			G



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
- 6 **if** $v \in Open$ **then** Reordenar $Open$
- 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
	inf	inf	inf	inf	inf
	5	4			2
	inf	inf	inf	inf	inf
	4	3	2		1
	inf	inf	inf	inf	inf
	3	2			



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
- 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
	inf	inf	inf	inf	inf
	5	4			2
	inf	inf	inf	inf	inf
	4	3	2		1
	inf	inf	inf	inf	inf
	3	2			



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
	inf		inf	inf	inf
	5		4		2
	inf		inf	inf	inf
	4		3	2	1
	inf		inf	inf	inf
	3		2		



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
	inf		inf	inf	inf
	5		4		2
	inf		inf	inf	inf
	4		3	2	1
	inf		inf	inf	inf
	3		2		



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
	inf		inf	inf	inf
(0,0)	5		4		2
	inf		inf	inf	inf
	4		3	2	1
	inf		inf	inf	inf
	3		2		

- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
	1		inf	inf	inf
(0,0)	5		4		2
	inf		inf	inf	inf
	4		3	2	1
	inf		inf	inf	inf
	3		2		



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1		inf	inf	inf
(0,0)	5		4		2
	inf		inf	inf	inf
	4		3	2	1
	inf		inf	inf	inf
	3		2		



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
 - 2 $Open \leftarrow \{s_0\}$
 - 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
 - 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
- 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
<div>S</div>					
	6	(0,0)	5		3
6	1		inf	inf	inf
(0,0)	5		4		2
	inf		inf	inf	inf
	4		3	2	1
	inf		inf	inf	inf
	3		2		<div>G</div>



- 1 **for each** $s \in \mathcal{S}$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1		inf	inf	inf
(0,0)	5		4		2
	inf		inf	inf	inf
	4		3	2	1
	inf		inf	inf	inf
	3		2		



- 1 for each $s \in S$ do $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 while $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 if u es objetivo return u
 - 7 for each $v \in Succ(u)$ do
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 if $cost_v \geq g(v)$ return
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 if $v \in Open$ then Reordenar $Open$
 - 7 else Insertar v en $Open$

6	0	6	1	inf	inf
	6	(0,0)	5		3
6	1		inf	inf	inf
(0,0)	5		4		2
	inf		inf	inf	inf
	4		3	2	1
	inf		inf	inf	inf
	3		2		



- 1 for each $s \in S$ do $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 while $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if u es objetivo return u**
- 7 for each $v \in Succ(u)$ do
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if $cost_v \geq g(v)$ return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if $v \in Open$ then Reordenar $Open$**
 - 7 **else Insertar v en $Open$**

6	0	6	1	inf	inf
	6	(0,0)	5		3
6	1		inf	inf	inf
(0,0)	5		4		2
	inf		inf	inf	inf
	4		3	2	1
	inf		inf	inf	inf
	3		2		



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
	6	(0,0)	5		3
6	1		inf	inf	inf
(0,0)	5		4		2
	inf		inf	inf	inf
	4		3	2	1
	inf		inf	inf	inf
	3		2		



- 1 for each $s \in S$ do $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 while $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 if u es objetivo return u
 - 7 for each $v \in Succ(u)$ do
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 if $cost_v \geq g(v)$ return
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 if $v \in Open$ then Reordenar $Open$
 - 7 else Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	inf	inf	inf	inf
(0,0)	5	4			2
	inf	inf	inf	inf	inf
	4	3	2	1	
	inf	inf	inf	inf	inf
	3	2			



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
	6	(0,0)	5		3
6	1	inf	inf	inf	inf
(0,0)	5	4			2
	inf	inf	inf	inf	inf
	4	3	2	1	
	inf	inf	inf	inf	inf
	3	2			



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	inf		inf	inf
(0,0)	5	(1,0)	4		2
	inf	inf	inf	inf	inf
	4	3	2	1	
	inf	inf	inf	inf	inf
	3	2			



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1		2	inf	inf
(0,0)	5	(1,0)	4		2
	inf		inf	inf	inf
	4		3	2	1
	inf		inf	inf	inf
	3		2		



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
	inf		inf	inf	inf
	4		3	2	1
	inf		inf	inf	inf
	3		2		



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
	inf		inf	inf	inf
	4		3	2	1
	inf		inf	inf	inf
	3		2		



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
	inf		inf	inf	inf
	4		3	2	1
	inf		inf	inf	inf
	3		2		



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 **Extrae un u desde $Open$ con menor valor- f**
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
	inf		inf	inf	inf
	4		3	2	1
	inf		inf	inf	inf
	3		2		



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
	inf		inf	inf	inf
	4		3	2	1
	inf		inf	inf	inf
	3		2		



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
	inf	inf	inf	inf	inf
	4	3	2		1
	inf	inf	inf	inf	inf
	3	2			



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
	inf	inf	inf	inf	inf
	4	3	2	1	
	inf	inf	inf	inf	inf
	3	2			



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
	inf	inf	inf	inf	inf
	4	3	2	1	
	inf	inf	inf	inf	inf
	3	2			



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
	inf	inf	inf	inf	inf
	4	(1,1)	3	2	1
	inf	inf	inf	inf	inf
	3		2		

- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
	inf	3		inf	inf
	4	(1,1)	3	2	1
	inf	inf	inf	inf	inf
	3	2			



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
	inf	6	3	inf	inf
	4	(1,1)	3	2	1
	inf		inf	inf	inf
	3		2		



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
- 6 **if** $v \in Open$ **then** Reordenar $Open$
- 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
S					
	6	(0,0)	5	inf	3
6	1	6	2		inf
(0,0)	5	(1,0)	4	inf	2
	inf	6	3		inf
	4	(1,1)	3	2	1
	inf		inf	inf	inf
	3		2		G



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
	inf	6	3	inf	inf
	4	(1,1)	3	2	1
	inf		inf	inf	inf
					
	3		2		



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
	inf	6	3	inf	inf
	4	(1,1)	3	2	1
	inf		inf	inf	inf
	3		2		



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
	inf	6	3	inf	inf
	4	(1,1)	3	2	1
	inf		inf	inf	inf
	3		2		



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
	inf	6	3	inf	inf
	4	(1,1)	3	2	1
	inf		inf	inf	inf
	3		2		



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
	inf	6	3	inf	inf
	4	(1,1)	3	2	1
	inf		inf	inf	inf
	3		2		



- 1 for each $s \in S$ do $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 while $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 if u es objetivo return u
 - 7 for each $v \in Succ(u)$ do
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 if $cost_v \geq g(v)$ return
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 if $v \in Open$ then Reordenar $Open$
 - 7 else Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
	inf	6	3	inf	inf
	4	(1,1)	3	2	1
	inf		inf	inf	inf
	3		2		



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
	inf	6	3	inf	inf
	4	(1,1)	3	2	1
	inf		inf	inf	inf
	3		2		



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
	inf	6	3	inf	inf
	4	(1,1)	3	2	1
	inf	inf	inf	inf	inf
	3	2			



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
	inf	6	3	inf	inf
	4	(1,1)	3	2	1
	inf	inf	inf	inf	inf
	3	2			



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
	inf	6	3	inf	inf
	4	(1,1)	3	2	1
	inf	inf	inf	inf	inf
	3	2			



- 1 for each $s \in S$ do $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 while $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 if u es objetivo return u
 - 7 for each $v \in Succ(u)$ do
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 if $cost_v \geq g(v)$ return
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 if $v \in Open$ then Reordenar $Open$
 - 7 else Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
	inf	6	3	inf	inf
(0,1)	4	(1,1)	3	2	1
	inf	inf	inf	inf	inf
					
	3		2		



- 1 **for each** $s \in \mathcal{S}$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
	2	6	3	inf	inf
(0,1)	4	(1,1)	3	2	1
	inf		inf	inf	inf
					
	3		2		



- 1 **for each** $s \in \mathcal{S}$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
6	2	6	3	inf	inf
(0,1)	4	(1,1)	3	2	1
	inf		inf	inf	inf
					
	3		2		



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
6	2	6	3	inf	inf
(0,1)	4	(1,1)	3	2	1
	inf		inf	inf	inf
					
	3		2		



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
6	2	6	3	inf	inf
(0,1)	4	(1,1)	3	2	1
	inf		inf	inf	inf
					
	3		2		



- 1 **for each** $s \in \mathcal{S}$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 **Extrae un u desde $Open$ con menor valor- f**
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
6	2	6	3	inf	inf
(0,1)	4	(1,1)	3	2	1
	inf		inf	inf	inf
					
	3		2		


- 1 **for each** $s \in \mathcal{S}$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 **Extrae un u desde $Open$ con menor valor- f**
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
6	2	6	3	inf	inf
(0,1)	4	(1,1)	3	2	1
	inf		inf	inf	inf
					
	3		2		



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
6	2	6	3	inf	inf
(0,1)	4	(1,1)	3	2	1
	inf	inf	inf	inf	inf
					
	3	2			


- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
6	2	6	3	inf	inf
(0,1)	4	(1,1)	3	2	1
	inf		inf	inf	inf
					
	3		2		

- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
6	2	6	3	inf	inf
(0,1)	4	(1,1)	3	2	1
	inf		inf	inf	inf
					
	3		2		

- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
6	2	6	3	inf	inf
(0,1)	4	(1,1)	3	2	1
	inf		inf	inf	inf
	3		2		

- 1 **for each** $s \in \mathcal{S}$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
6	2	6	3	inf	inf
(0,1)	4	(1,1)	3	(1,2)	2
	inf		inf	inf	inf
					
	3		2		

- 1 **for each** $s \in \mathcal{S}$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
6	2	6	3	4	inf
(0,1)	4	(1,1)	3	(1,2)	1
	inf		inf	inf	inf
					
	3		2		

- 1 **for each** $s \in \mathcal{S}$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf	
<div>S</div>	6	(0,0)	5		3	
6	1	6	2	inf	inf	
(0,0)	5	(1,0)	4		2	
6	2	6	3	6	4	inf
(0,1)	4	(1,1)	3	(1,2)	2	1
	inf		inf	inf	inf	
	3		2		<div>G</div>	

- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
- 6 **if** $v \in Open$ **then** Reordenar $Open$
- 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5	inf	3
6	1	6	2		inf
(0,0)	5	(1,0)	4	inf	2
6	2	6	3		inf
(0,1)	4	(1,1)	3	(1,2)	1
	inf	inf	inf	inf	inf
					
	3		2		

- 1 **for each** $s \in \mathcal{S}$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5	inf	3
6	1	6	2		inf
(0,0)	5	(1,0)	4	inf	2
6	2	6	3		inf
(0,1)	4	(1,1)	3	inf	1
	inf	inf	inf		inf
				inf	
	3	2			

- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5	inf	3
6	1	6	2		inf
(0,0)	5	(1,0)	4		2
6	2	6	3	6	4
(0,1)	4	(1,1)	3	(1,2)	2
	inf	inf	inf	inf	inf
					
	3		2		

- 1 for each $s \in S$ do $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 while $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 if u es objetivo return u
 - 7 for each $v \in Succ(u)$ do
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 if $cost_v \geq g(v)$ return
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 if $v \in Open$ then Reordenar $Open$
 - 7 else Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
6	2	6	3	6	4
(0,1)	4	(1,1)	3	(1,2)	2
	inf	inf	inf	inf	inf
	3		2		

- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
6	2	6	3	6	4
(0,1)	4	(1,1)	3	(1,2)	2
	inf	inf	inf	inf	inf
	3		2		

- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
6	2	6	3	6	4
(0,1)	4	(1,1)	3	(1,2)	2
	inf	inf	inf	inf	inf
	3	(1,2)	2		


- 1 **for each** $s \in \mathcal{S}$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
6	2	6	3	6	4
(0,1)	4	(1,1)	3	(1,2)	2
	inf		4	inf	inf
					
	3	(1,2)	2		

- 1 **for each** $s \in \mathcal{S}$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
6	2	6	3	6	4
(0,1)	4	(1,1)	3	(1,2)	2
	inf	6	4	inf	inf
					
	3	(1,2)	2		

- 1 **for each** $s \in \mathcal{S}$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
- 6 **if** $v \in Open$ **then** Reordenar $Open$
- 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5	inf	3
6	1	6	2		inf
(0,0)	5	(1,0)	4	inf	2
6	2	6	3		inf
(0,1)	4	(1,1)	3	inf	1
	inf	6	4		inf
	3	(1,2)	2	inf	

- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
- 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf	
<div>S</div>	6	(0,0)	5		3	
6	1	6	2	inf	inf	
(0,0)	5	(1,0)	4		2	
6	2	6	3	6	4	inf
(0,1)	4	(1,1)	3	(1,2)	2	1
	inf	6	4	inf	inf	
3	(1,2)	2	<div>G</div>			

- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf	
<div>S</div>	6	(0,0)	5		3	
6	1	6	2	inf	inf	
(0,0)	5	(1,0)	4		2	
6	2	6	3	6	4	inf
(0,1)	4	(1,1)	3	(1,2)	2	1
inf	6	4	inf	inf	<div>G</div>	
3	(1,2)	2				

- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
6	2	6	3	6	4
(0,1)	4	(1,1)	3	(1,2)	2
	inf	6	4	inf	inf
	3	(1,2)	2		



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
6	2	6	3	6	4
(0,1)	4	(1,1)	3	(1,2)	2
	inf	6	4	inf	inf
	3	(1,2)	2		

- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 **Extrae un u desde $Open$ con menor valor- f**
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
6	2	6	3	6	4
					inf
(0,1)	4	(1,1)	3	(1,2)	2
					1
	inf	6	4	inf	inf
					
	3	(1,2)	2		



- 1 for each $s \in S$ do $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 while $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 if u es objetivo return u
- 7 for each $v \in Succ(u)$ do
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 if $cost_v \geq g(v)$ return
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 if $v \in Open$ then Reordenar $Open$
 - 7 else Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
6	2	6	3	6	4
(0,1)	4	(1,1)	3	(1,2)	2
	inf	6	4	inf	inf
					
	3	(1,2)	2		


- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf	
<div>S</div>	6	(0,0)	5		3	
6	1	6	2	inf	inf	
(0,0)	5	(1,0)	4		2	
6	2	6	3	6	4	inf
(0,1)	4	(1,1)	3	(1,2)	2	1
	inf	6	4	inf	inf	<div>G</div>
	3	(1,2)	2			



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
6	2	6	3	6	4
					inf
(0,1)	4	(1,1)	3	(1,2)	2
					1
	inf	6	4	inf	inf
					
	3	(1,2)	2		

- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
6	2	6	3	6	4
(0,1)	4	(1,1)	3	(1,2)	2
	inf	6	4	inf	inf
	3	(1,2)	2		

- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
6	2	6	3	6	4
(0,1)	4	(1,1)	3	(1,2)	2
	inf	6	4	inf	inf
	3	(1,2)	2		

- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf		
<div>S</div>	6	(0,0)	5		3		
	6	1	2		inf		
(0,0)	5	(1,0)	4	2			
6	2	6	3	6	4	inf	
(0,1)	4	(1,1)	3	(1,2)	2	(2,2)	1
	inf	6	4	inf			
	3	(1,2)	2			<div>G</div>	

- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf		
<div>S</div>	6	(0,0)	5		3		
6	1	6	2	inf	inf		
(0,0)	5	(1,0)	4		2		
6	2	6	3	6	4	5	
(0,1)	4	(1,1)	3	(1,2)	2	(2,2)	1
	inf	6	4	inf	inf		<div>G</div>
	3	(1,2)	2				


- 1 **for each** $s \in \mathcal{S}$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
6	2	6	3	6	4
(0,1)	4	(1,1)	3	(1,2)	2
	inf	6	4	inf	inf
					
	3	(1,2)	2		


- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
- 6 **if** $v \in Open$ **then** Reordenar $Open$
- 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
6	2	6	3	6	4
(0,1)	4	(1,1)	3	(1,2)	2
	inf	6	4	inf	inf
					
	3	(1,2)	2		

- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
- 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
6	2	6	3	6	4
(0,1)	4	(1,1)	3	(1,2)	2
	inf	6	4	inf	inf
					
	3	(1,2)	2		

- 1 for each $s \in S$ do $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 while $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 if u es objetivo return u
 - 7 for each $v \in Succ(u)$ do
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 if $cost_v \geq g(v)$ return
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 if $v \in Open$ then Reordenar $Open$
 - 7 else Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5	inf	3
6	1	6	2		inf
(0,0)	5	(1,0)	4	inf	2
6	2	6	3		6
(0,1)	4	(1,1)	3	(1,2)	2
	inf	6	4	inf	inf
	3	(1,2)	2		



- 1 for each $s \in S$ do $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 while $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 if u es objetivo return u
 - 7 for each $v \in Succ(u)$ do
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 if $cost_v \geq g(v)$ return
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 if $v \in Open$ then Reordenar $Open$
 - 7 else Insertar v en $Open$

6	0	6	1	inf	inf		
<div>S</div>	6	(0,0)	5		3		
6	1	6	2	inf	inf		
(0,0)	5	(1,0)	4		2		
6	2	6	3	6	4	6	5
(0,1)	4	(1,1)	3	(1,2)	2	(2,2)	1
	inf	6	4	inf	inf		
3	(1,2)	2	<div>G</div>				



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf		
<div>S</div>	6	(0,0)	5		3		
6	1	6	2	inf	inf		
(0,0)	5	(1,0)	4		2		
6	2	6	3	6	4	6	5
(0,1)	4	(1,1)	3	(1,2)	2	(2,2)	1
	inf	6	4	inf	inf		
3	(1,2)	2	<div>G</div>				



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
6	2	6	3	6	4
(0,1)	4	(1,1)	3	(1,2)	2
	inf	6	4	inf	inf
	3	(1,2)	2		



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
6	2	6	3	6	4
(0,1)	4	(1,1)	3	(1,2)	2
	inf	6	4	inf	inf
	3	(1,2)	2		



- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		2
6	2	6	3	6	4
(0,1)	4	(1,1)	3	(1,2)	2
	inf	6	4	inf	inf
					
	3	(1,2)	2		

- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	inf
(0,0)	5	(1,0)	4		(3,2) 2
6	2	6	3	6	4
(0,1)	4	(1,1)	3	(1,2)	2
	inf	6	4	inf	inf
	3	(1,2)	2		

- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf
					
	6	(0,0)	5		3
6	1	6	2	inf	6
(0,0)	5	(1,0)	4		(3,2) 2
6	2	6	3	6	4
(0,1)	4	(1,1)	3	(1,2)	2
	inf	6	4	inf	inf
					
	3	(1,2)	2		

- 1 **for each** $s \in \mathcal{S}$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf		inf	
<div>S</div>							
	6	(0,0)	5			3	
6	1	6	2	inf		8	6
(0,0)	5	(1,0)	4			(3,2)	2
6	2	6	3	6	4	6	5
(0,1)	4	(1,1)	3	(1,2)	2	(2,2)	1
inf		6	4	inf		inf	
						<div>G</div>	
3		(1,2)	2				

- 1 **for each** $s \in \mathcal{S}$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
- 6 **if** $v \in Open$ **then** Reordenar $Open$
- 7 **else** Insertar v en $Open$

6	0	6	1	inf		inf	
S						3	
	6	(0,0)	5	inf			
6	1	6	2			8	6
(0,0)	5	(1,0)	4			(3,2)	2
6	2	6	3	6	4	6	5
(0,1)	4	(1,1)	3	(1,2)	2	(2,2)	1
	inf	6	4	inf		inf	
	3	(1,2)	2			G	

- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
- 7 **else** Insertar v en $Open$

6061	inf	inf
<div>S</div>		
6(0,0)65	inf	3
6162	inf	86
(0,0)5(1,0)4		(3,2)2
6263	64	65
(0,1)4(1,1)3	(1,2)2	(2,2)1
inf	64inf	inf
3	(1,2)2	<div>G</div>

- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6061	inf	inf
<div>S</div>		
6(0,0)6(0,0)	5(0,0)4(1,0)	3(3,2)2(3,2)
6162	inf	86
(0,0)5(0,1)	(1,0)4(1,1)	(2,2)1(2,2)
626364	4(1,2)2(1,2)	5(2,2)1(2,2)
(0,1)4(0,1)	(1,1)3(1,1)	(2,2)1(2,2)
inf3	64(1,2)2	inf
		<div>G</div>

- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf		inf	
<div>S</div>							
	6	(0,0)	5	inf		8	6
6	1	6	2				
(0,0)	5	(1,0)	4			(3,2)	2
6	2	6	3	6	4	6	5
(0,1)	4	(1,1)	3	(1,2)	2	(2,2)	1
inf		6	4	inf		inf	
						<div>G</div>	
3		(1,2)	2			(3,2)	

- 1 **for each** $s \in \mathcal{S}$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf		inf	
<div>S</div>							
	6	(0,0)	5			3	
6	1	6	2	inf		8	6
(0,0)	5	(1,0)	4			(3,2)	2
6	2	6	3	6	4	6	5
(0,1)	4	(1,1)	3	(1,2)	2	(2,2)	1
	inf	6	4	inf		6	
						<div>G</div>	
	3	(1,2)	2			(3,2)	

- 1 **for each** $s \in \mathcal{S}$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf		inf	
<div>S</div>							
	6	(0,0)	5			3	
6	1	6	2	inf		8	6
(0,0)	5	(1,0)	4			(3,2)	2
6	2	6	3	6	4	6	5
(0,1)	4	(1,1)	3	(1,2)	2	(2,2)	1
	inf	6	4	inf		6	6
						<div>G</div>	
	3	(1,2)	2			(3,2)	

- 1 **for each** $s \in \mathcal{S}$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
- 6 **if** $v \in Open$ **then** Reordenar $Open$
- 7 **else** Insertar v en $Open$

6	0	6	1	inf	inf		
<div>S</div>	6	(0,0)	5		3		
6	1	6	2	inf	8	6	
(0,0)	5	(1,0)	4		(3,2)	2	
6	2	6	3	6	4	6	5
(0,1)	4	(1,1)	3	(1,2)	2	(2,2)	1
	inf	6	4	inf	6	6	
	3	(1,2)	2		<div>G</div>	(3,2)	

- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf		inf	
<div>S</div>							
	6	(0,0)	5			3	
6	1	6	2	inf		8	6
(0,0)	5	(1,0)	4			(3,2)	2
6	2	6	3	6	4	6	5
(0,1)	4	(1,1)	3	(1,2)	2	(2,2)	1
inf		6	4	inf		6	6
						<div>G</div>	
3		(1,2)	2			(3,2)	

- 1 for each $s \in S$ do $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 while $Open \neq \emptyset$
 - 5 Extrae un u desde $Open$ con menor valor- f
 - 6 if u es objetivo return u
 - 7 for each $v \in Succ(u)$ do
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 if $cost_v \geq g(v)$ return
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 if $v \in Open$ then Reordenar $Open$
 - 7 else Insertar v en $Open$

6	0	6	1	inf		inf	
<div>S</div>							
	6	(0,0)	5			3	
6	1	6	2	inf		8	6
(0,0)	5	(1,0)	4			(3,2)	2
6	2	6	3	6	4	6	5
(0,1)	4	(1,1)	3	(1,2)	2	(2,2)	1
inf		6	4	inf		6	6
						<div>G</div>	
3		(1,2)	2			(3,2)	



- 1 **for each** $s \in \mathcal{S}$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
 - 5 **Extrae un u desde $Open$ con menor valor- f**
 - 6 **if** u es objetivo **return** u
 - 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf		inf	
<div>S</div>							
	6	(0,0)	5			3	
6	1	6	2	inf		8	6
(0,0)	5	(1,0)	4			(3,2)	2
6	2	6	3	6	4	6	5
(0,1)	4	(1,1)	3	(1,2)	2	(2,2)	1
inf		6	4	inf		6	6
						<div>G</div>	
3		(1,2)	2			(3,2)	

- 1 for each $s \in S$ do $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 while $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if u es objetivo return u**
- 7 for each $v \in Succ(u)$ do
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if $cost_v \geq g(v)$ return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if $v \in Open$ then Reordenar $Open$**
 - 7 **else Insertar v en $Open$**

6	0	6	1	inf		inf	
S							
	6	(0,0)	5			3	
6	1	6	2	inf		8	6
(0,0)	5	(1,0)	4			(3,2)	2
6	2	6	3	6	4	6	5
(0,1)	4	(1,1)	3	(1,2)	2	(2,2)	1
inf		6	4	inf		6	6
						G	
	3	(1,2)	2			(3,2)	

- 1 **for each** $s \in \mathcal{S}$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

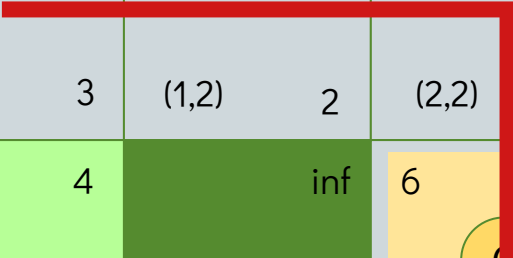
6	0	6	1	inf	inf
					
	6	(0,0)	5	inf	3
6	1	6	2		8
				inf	6
(0,0)	5	(1,0)	4		(3,2)
				inf	2
6	2	6	3		6
				inf	5
(0,1)	4	(1,1)	3		(2,2)
				inf	1
	inf	6	4		6
				inf	6
					
	3	(1,2)	2	inf	(3,2)

- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf		inf	
S						3	
	6	(0,0)	5	inf			
6	1	6	2			8	6
(0,0)	5	(1,0)	4			(3,2)	2
6	2	6	3			6	5
(0,1)	4	(1,1)	3	(1,2)	2	(2,2)	1
	inf	6	4	inf		6	6
	3	(1,2)	2			G	
						(3,2)	

- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf		inf	
<div>S</div>							
	6	(0,0)	5			3	
6	1	6	2	inf		8	6
(0,0)	5	(1,0)	4			(3,2)	2
6	2	6	3	6	4	6	5
(0,1)	4	(1,1)	3	(1,2)	2	(2,2)	1
	inf	6	4	inf		6	<div>G</div>
	3	(1,2)	2			(3,2)	



- 1 **for each** $s \in \mathcal{S}$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf		inf	
<div>S</div>							
	6	(0,0)	5			3	
6	1	6	2	inf		8	6
(0,0)	5	(1,0)	4			(3,2)	2
6	2	6	3	6	4	6	5
(0,1)	4	(1,1)	3	(1,2)	2	(2,2)	1
	inf	6	4	inf		6	<div>G</div>
	3	(1,2)	2			(3,2)	

- 1 **for each** $s \in S$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf		inf	
<div>S</div>		6	(0,0)	5			3
6	1	6	2	inf		8	6
(0,0)	5	(1,0)	4			(3,2)	2
6	2	6	3	6	4	6	5
(0,1)	4	(1,1)	3	(1,2)	2	(2,2)	1
inf		6	4	inf		6	<div>G</div>
3		(1,2)	2			(3,2)	

- 1 **for each** $s \in \mathcal{S}$ **do** $g(s) \leftarrow \infty$
- 2 $Open \leftarrow \{s_0\}$
- 3 $g(s_0) \leftarrow 0$; $f(s_0) \leftarrow h(s_0)$
- 4 **while** $Open \neq \emptyset$
- 5 Extrae un u desde $Open$ con menor valor- f
- 6 **if** u es objetivo **return** u
- 7 **for each** $v \in Succ(u)$ **do**
 - 1 $cost_v = g(u) + c(u, v)$
 - 2 **if** $cost_v \geq g(v)$ **return**
 - 3 $parent(v) \leftarrow u$
 - 4 $g(v) \leftarrow cost_v$
 - 5 $f(v) \leftarrow g(v) + h(v)$
 - 6 **if** $v \in Open$ **then** Reordenar $Open$
 - 7 **else** Insertar v en $Open$

6	0	6	1	inf		inf	
S		6	(0,0)	5			3
6	1	6	2	inf		8	6
(0,0)	5	(1,0)	4			(3,2)	2
6	2	6	3	6	4	6	5
(0,1)	4	(1,1)	3	(1,2)	2	(2,2)	1
inf		6	4	inf		6	6
3		(1,2)	2			G	
						(3,2)	



Optimalidad

¿Por qué A* es óptimo?

$$f(n) = g(n) + h(n)$$

$g(n)$: costo de un camino desde s_{start} hasta el nodo n

$h(n)$: *estima* el costo de un camino desde n hasta s_{goal}



¿Por qué A^* es óptimo?

$\delta(s, t)$: costo de un camino óptimo entre s y t

Admisibilidad : h es *admissible* si y solo si

$$h(s) \leq \delta(s, s_{goal}),$$

para todo estado s .



¿Por qué A* es óptimo?

$\delta(s, t)$: costo de un camino óptimo entre s y t

Admisibilidad : h es *admissible* si y solo si

$$h(s) \leq \delta(s, s_{goal}),$$

para todo estado s .



Algoritmo A*

$$h(s) \leq \delta(s, s_{goal})$$

$$f(s^*) = g(s^*) + h(s^*)$$



Algoritmo Weighted A*

$$h(s) \leq \delta(s, s_{goal})$$

$$f(s^*) = g(s^*) + wh(s^*)$$



Weighted A*

$$h(s) \leq \delta(s, s_{goal})$$

Al ser ejecutado con una heurística h admisible,
Weighted A* retorna una solución cuyo costo está
acotado superiormente por $w \cdot \underbrace{\delta(s_{start}, s_{goal})}_{\text{costo óptimo}}$

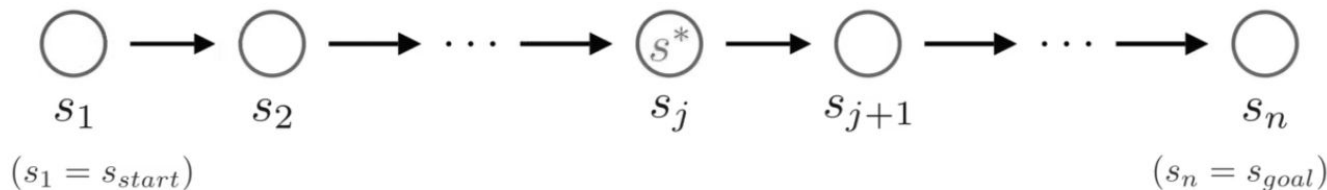
$$f(s^*) = g(s^*) + wh(s^*)$$

Weighted A*

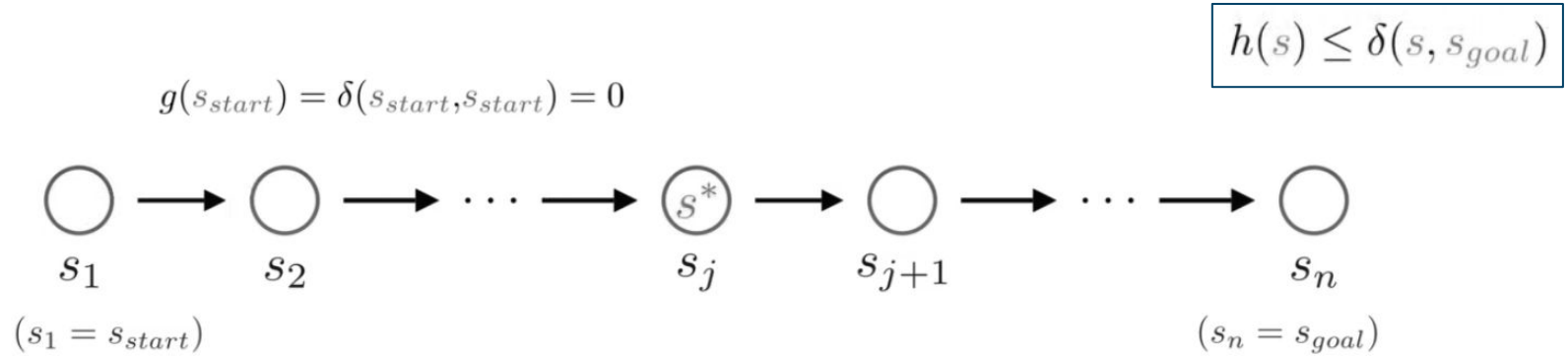
1. s^* está en un camino óptimo hacia s_{goal}
2. se cumple que $g(s^*) = \delta(s_{start}, s^*)$

$$h(s) \leq \delta(s, s_{goal})$$

$$g(s_{start}) = \delta(s_{start}, s_{start}) = 0$$



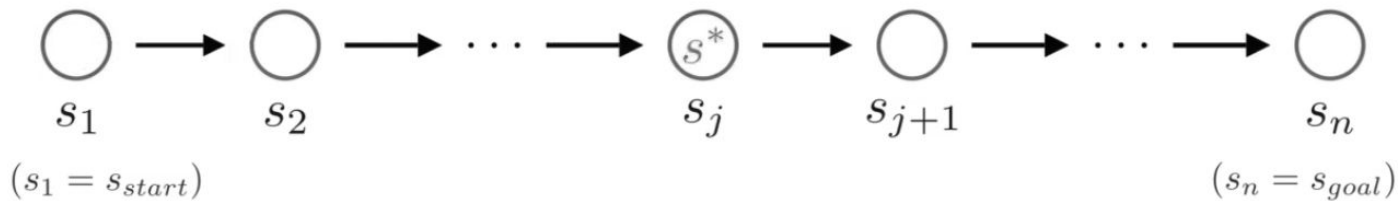
Weighted A*



Weighted A*

$$h(s) \leq \delta(s, s_{goal})$$

$$g(s_{start}) = \delta(s_{start}, s_{start}) = 0$$

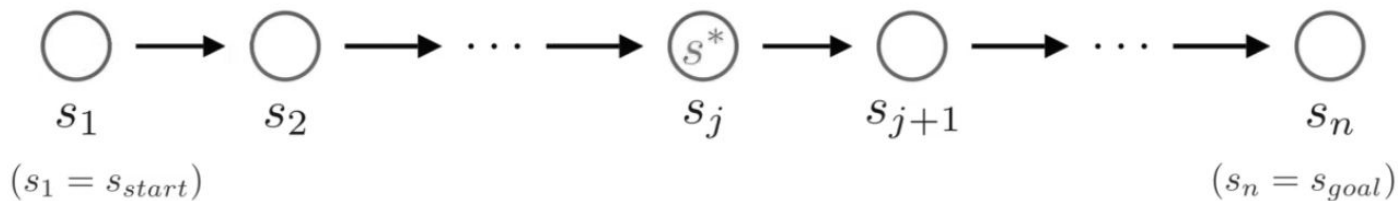


$$\delta(s_{start}, s^*) + \delta(s^*, s_{goal}) = \delta(s_{start}, s_{goal})$$

Weighted A*

$$h(s) \leq \delta(s, s_{goal})$$

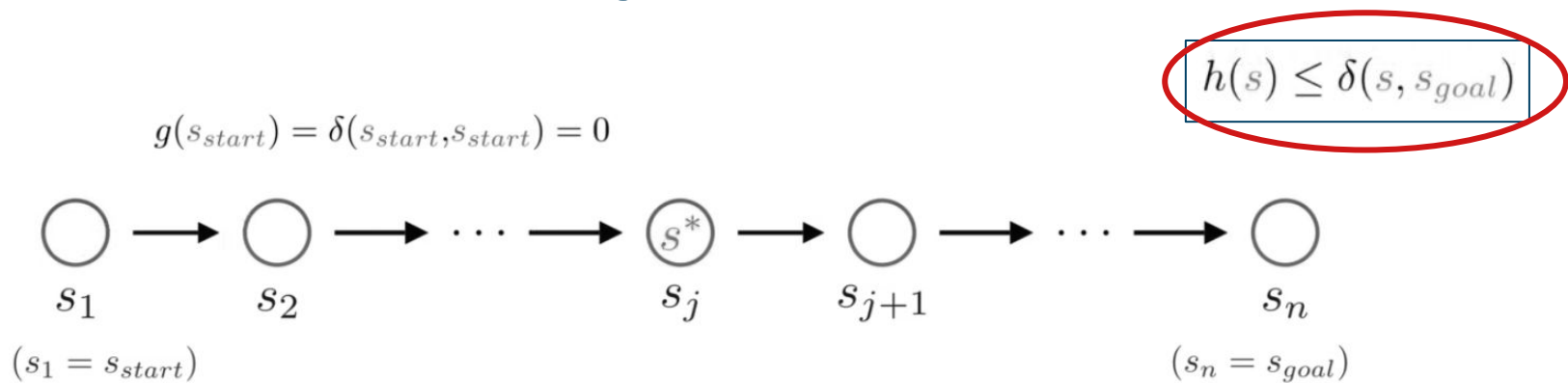
$$g(s_{start}) = \delta(s_{start}, s_{start}) = 0$$



$$\delta(s_{start}, s^*) + \delta(s^*, s_{goal}) = \delta(s_{start}, s_{goal})$$

$$g(s^*)$$

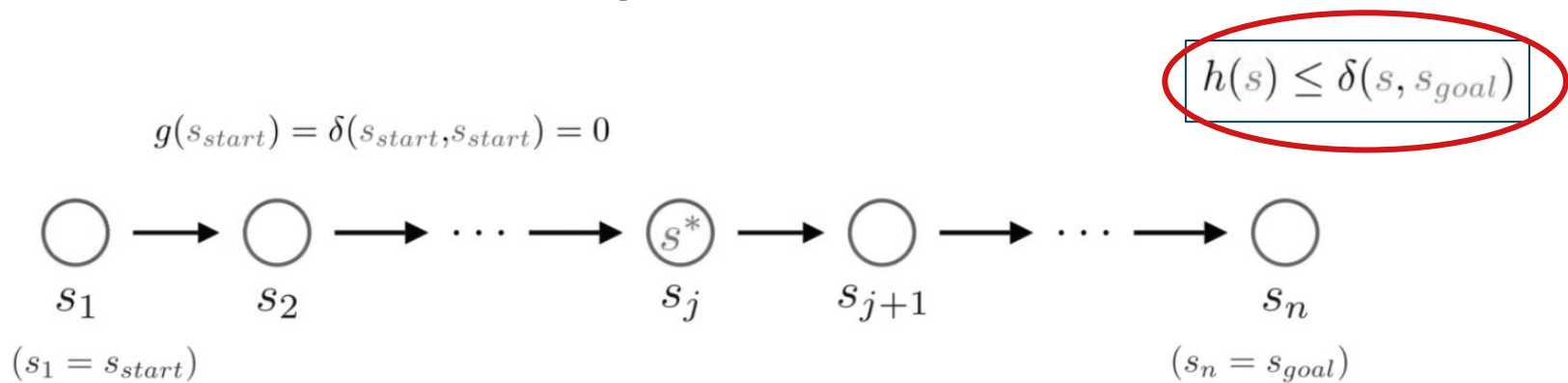
Weighted A*



$$\delta(s_{start}, s^*) + \delta(s^*, s_{goal}) = \delta(s_{start}, s_{goal})$$

$$g(s^*)$$

Weighted A*

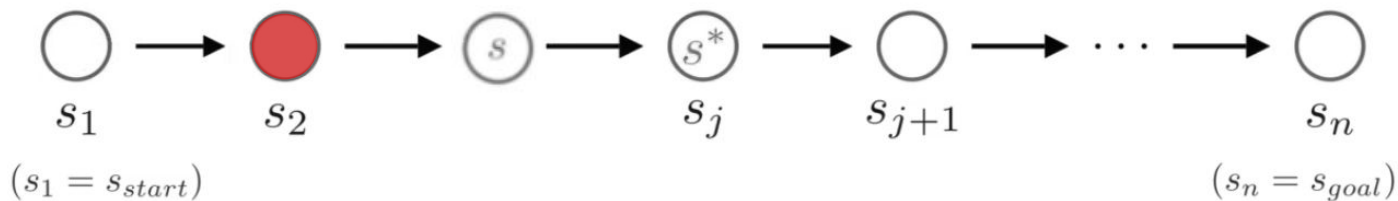


$$\delta(s_{start}, s^*) + \delta(s^*, s_{goal}) = \delta(s_{start}, s_{goal})$$

$$g(s^*) + h(s^*) \leq \delta(s_{start}, s_{goal})$$

Weighted A*

$$g(s_{start}) = \delta(s_{start}, s_{start}) = 0$$



$$f(s) = \min_{t \in Open} \{f(t)\}$$

$$f(s) \leq f(s^*)$$

Weighted A*

$$f(s) = \min_{t \in Open} \{f(t)\}$$
$$f(s) \leq f(s^*)$$



Weighted A*

$$f(s) = \min_{t \in Open} \{f(t)\}$$

$$f(s) \leq f(s^*)$$



Weighted A*

$$f(s) = \min_{t \in Open} \{f(t)\}$$

$$f(s) \leq f(s^*)$$

$$f(s) \leq g(s^*) + wh(s^*)$$



Weighted A*

$$f(s) = \min_{t \in Open} \{f(t)\}$$

$$f(s) \leq f(s^*)$$

$$f(s) \leq g(s^*) + wh(s^*)$$

$$f(s) \leq w(g(s^*) + h(s^*)) \quad (con\ w > 1)$$



Weighted A*

$$f(s) = \min_{t \in Open} \{f(t)\}$$

$$f(s) \leq f(s^*)$$

$$f(s) \leq g(s^*) + wh(s^*)$$

$$f(s) \leq w(g(s^*) + h(s^*)) \quad (con\ w > 1)$$



Weighted A*

$$f(s) = \min_{t \in Open} \{f(t)\}$$

$$f(s) \leq f(s^*)$$

$$f(s) \leq g(s^*) + wh(s^*)$$

$$f(s) \leq w(g(s^*) + h(s^*)) \quad (con\ w > 1)$$

$$f(s) \leq w\delta(s_{start}, s_{goal})$$



Weighted A*

Todo estado s que es extraído de Open satisface:

$$f(s) \leq w\delta(s_{start}, s_{goal})$$



Weighted A*

Todo estado s que es extraído de Open satisface:

$$f(s) \leq w\delta(s_{start}, s_{goal})$$

En particular, cuando s_{goal} es extraído:

$$g(s_{goal}) + wh(s_{goal}) \leq w\delta(s_{start}, s_{goal})$$



Weighted A*

Todo estado s que es extraído de Open satisface:

$$f(s) \leq w\delta(s_{start}, s_{goal})$$

En particular, cuando s_{goal} es extraído:

$$g(s_{goal}) + \cancel{wh(s_{goal})} \leq w\delta(s_{start}, s_{goal})$$

$$g(s_{goal}) \leq w\delta(s_{start}, s_{goal})$$



¿Por qué A* es óptimo?

$$g(s_{goal}) \leq w\delta(s_{start}, s_{goal})$$



¿Por qué A* es óptimo?

$$g(s_{goal}) \leq \delta(s_{start}, s_{goal})$$

con $w = 1$



¿Por qué A^* es óptimo?

https://www.youtube.com/watch?v=_41v4I5GTNc

