

FUNDAMENTOS DE INTELIGÊNCIA ARTIFICIAL

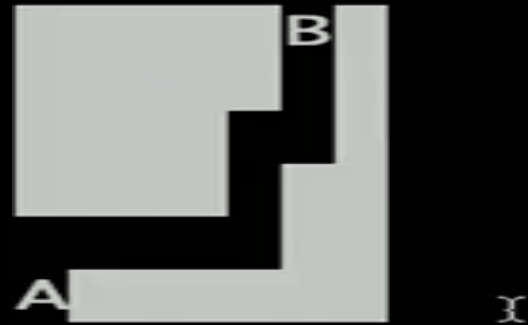
Aula 5: SEARCHING

Prof. Dr. Rodrigo Xavier de Almeida Leão
Cientista de Dados e Big Data

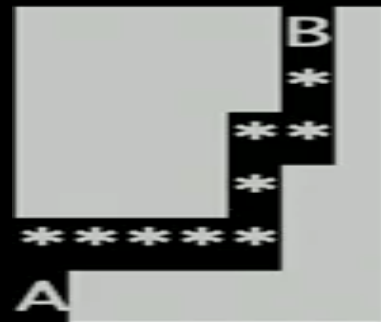


MAZE

```
python maze.py maze1.txt
```



```
Solving...  
States Explored: 11  
Solution:
```



```
workspace@Brian-MBP maze % open maze.png
```

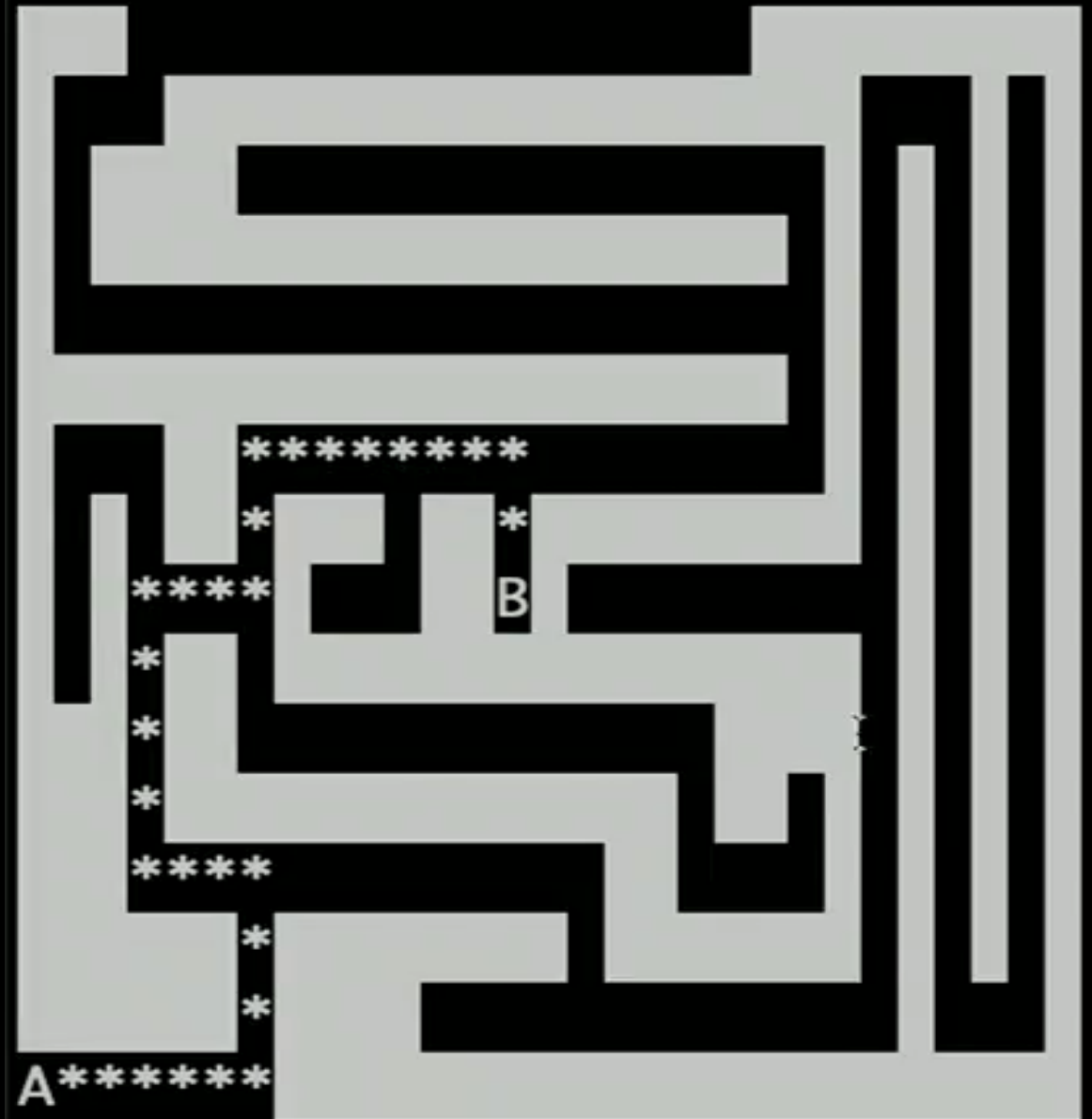


maze2.txt

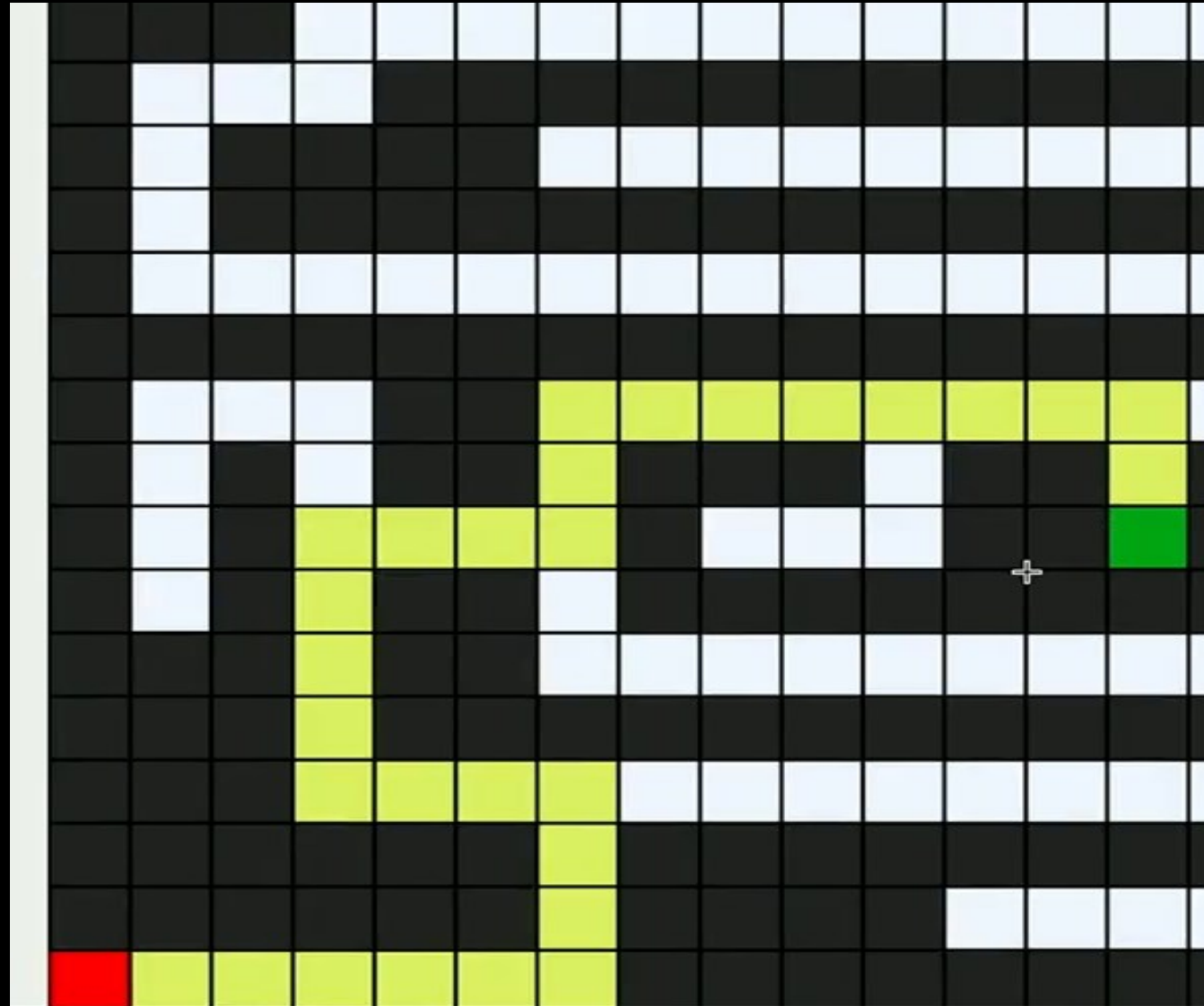
```
5      #                               # # # #
6      #####
7      #    ##                          # # # #
8      # # ## ### ## #####          # # #
9      # #    #   ##B#                  # # #
10     # # ## #####
11     ### ##                          ##### # # #
12     ### ##### ## # # # #
13     ###                      ##    # # # #
14     ##### ##### ##### # # #
15     ##### #####                #   #
16     A      #####
17
```

```
% python maze.py maze2.txt
```

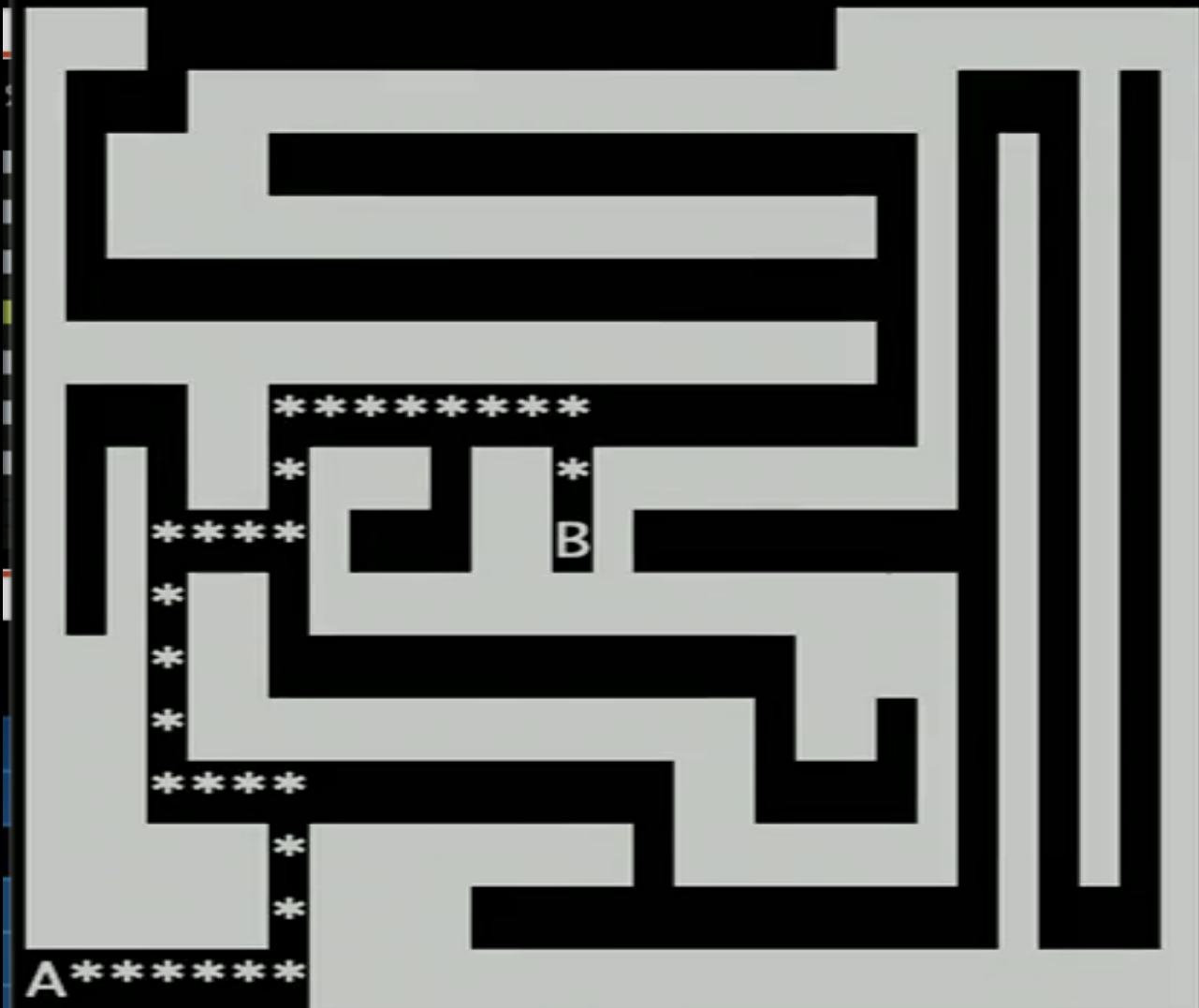
Solution:



|workspace@Brian-MBP maze % open maze.png



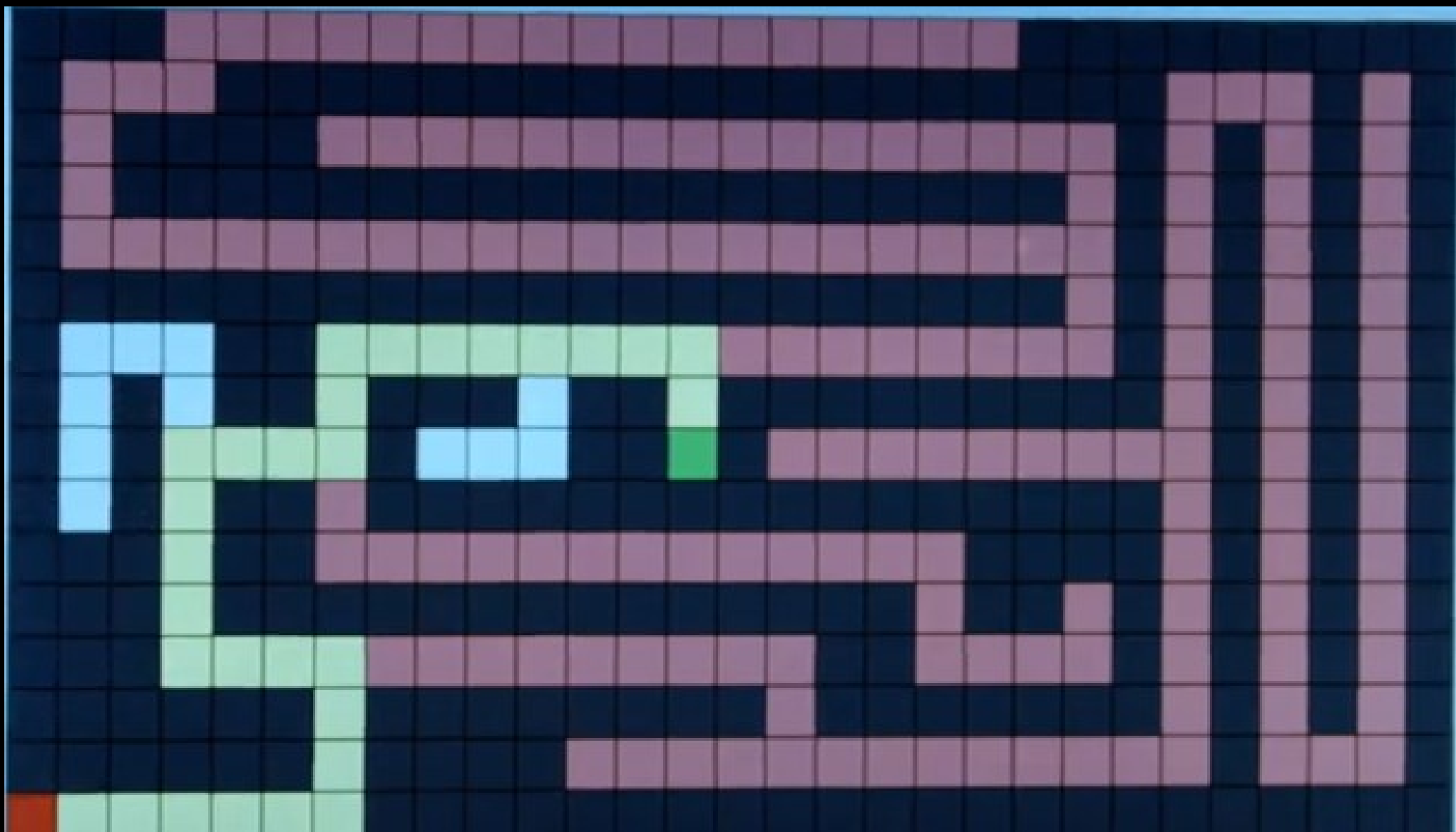
```
States Explored: 399
Solution:
```



```
m = Maze(sys.argv[1])  
print("Maze:")  
m.print()  
print("Solving...")  
m.solve()  
print("States Explored:", m.num_explored)  
print("Solution:")  
m.print()  
m.output_image("maze.png", show_explored=True)
```



```
python maze.py maze2.txt
```

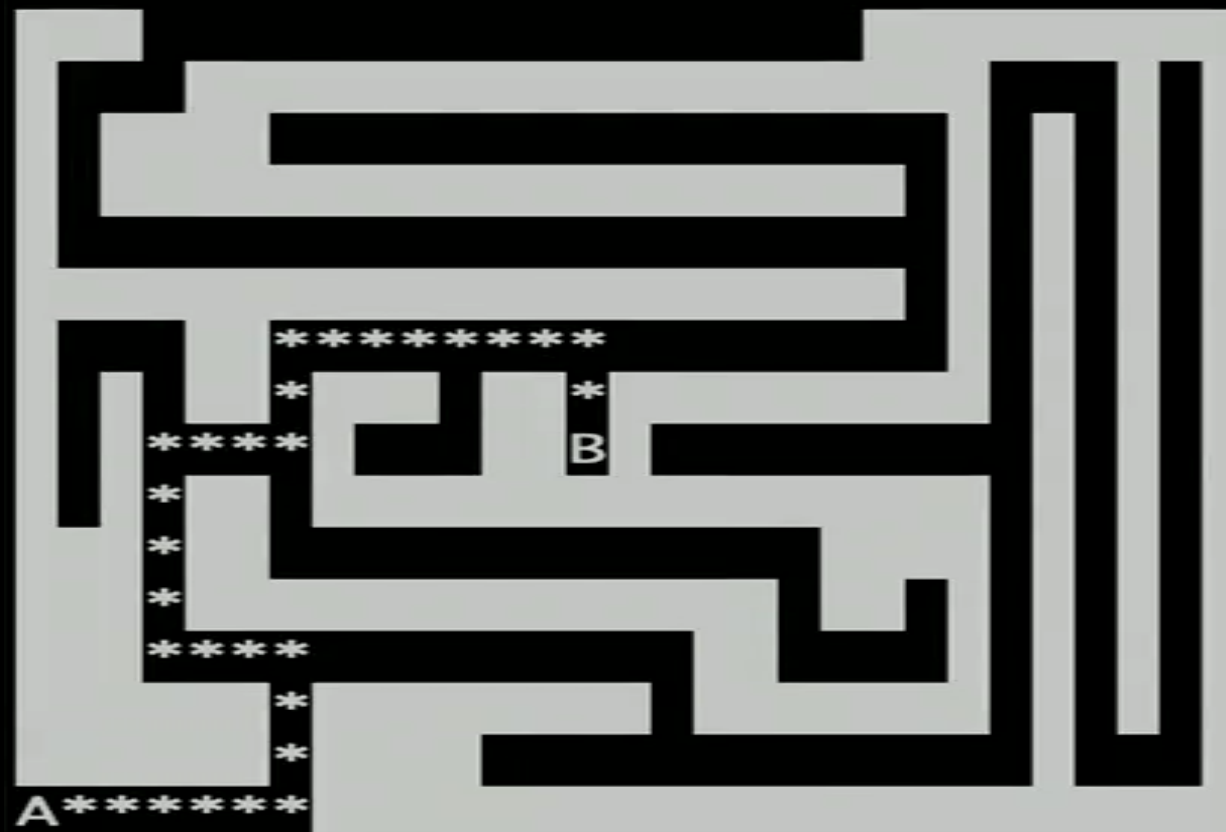


```
# Initialize frontier to just the starting position
start = Node(state=self.start, parent=None, action=None)
frontier = QueueFrontier()
frontier.add(start)
```

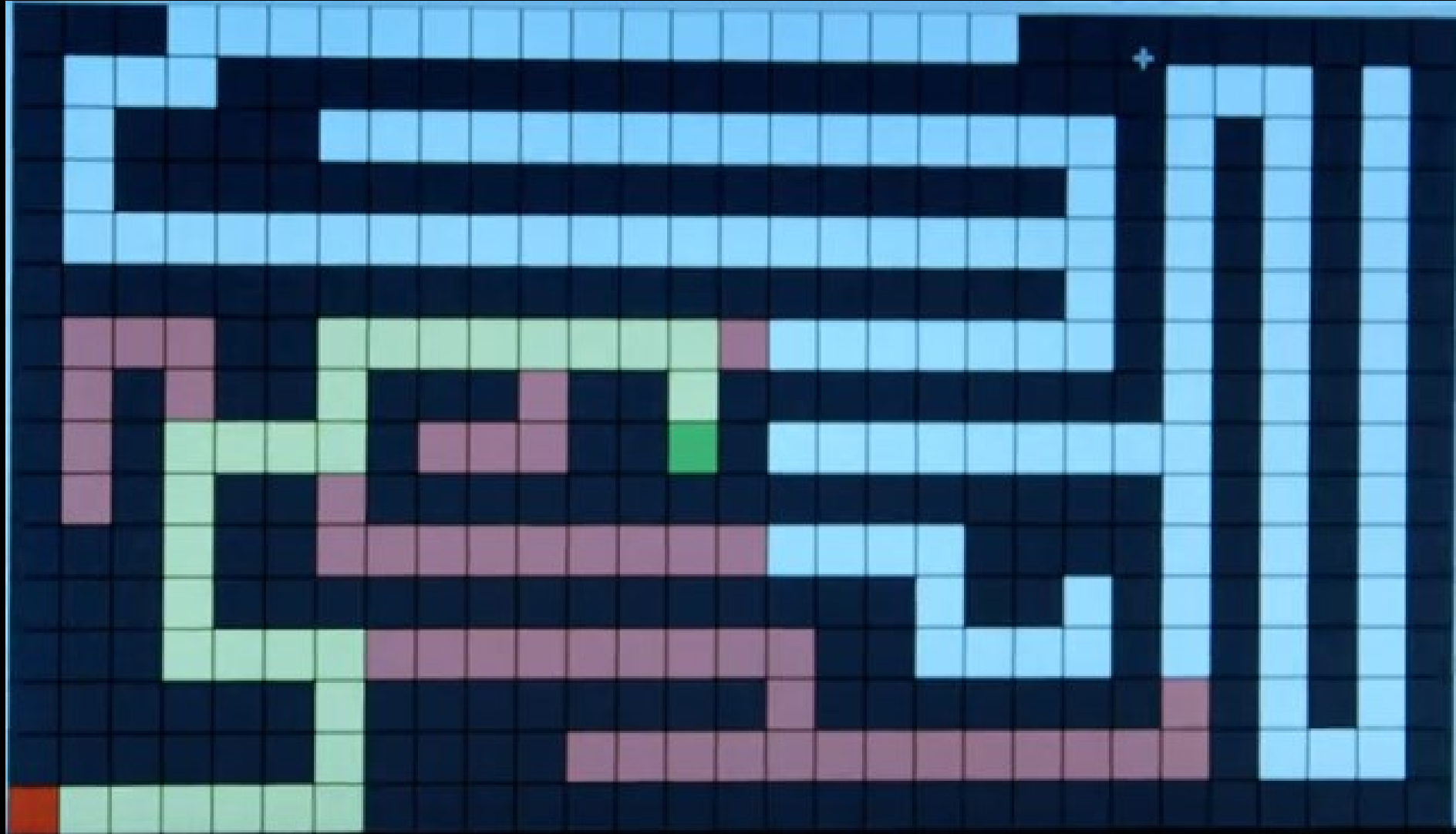
Solving...

States Explored: 77

Solution:



```
workspace@Brian-MBP maze % open maze.png
```

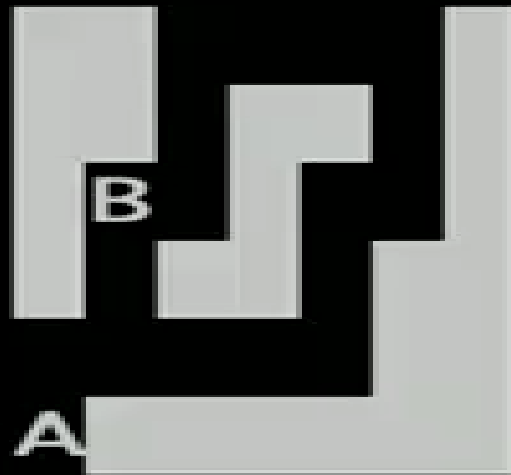


MAZE 3

maze3.txt

```
1    ##    #  
2    ##  ##  #  
3    #B  #   #  
4    #  ##  ##  
5    |      |  ##  
6    A#####  
7
```

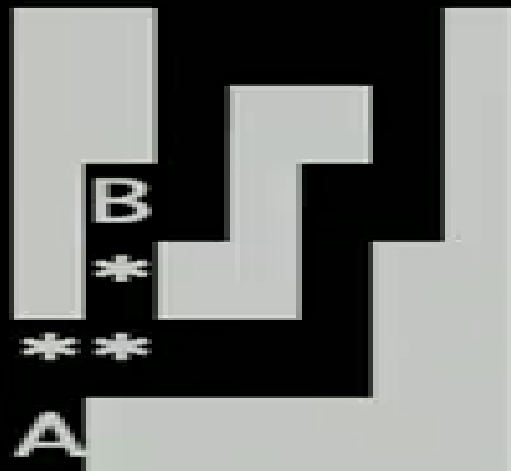
BFS



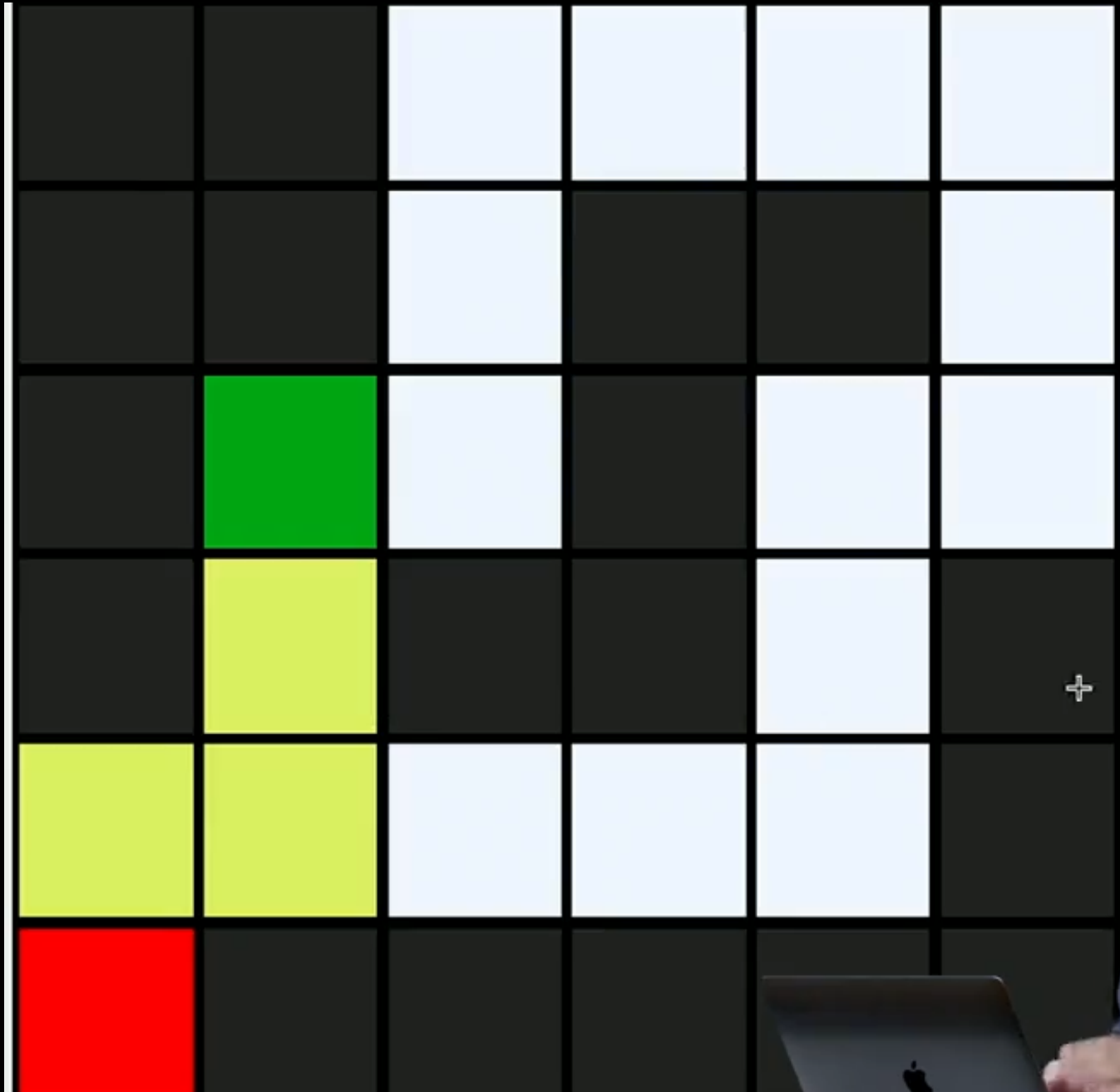
Solving...

States Explored: 6

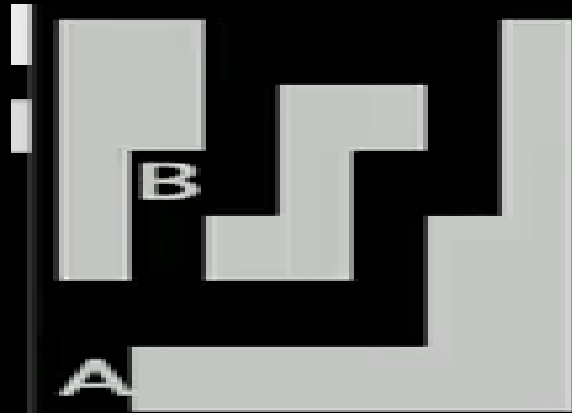
Solution:



BFS



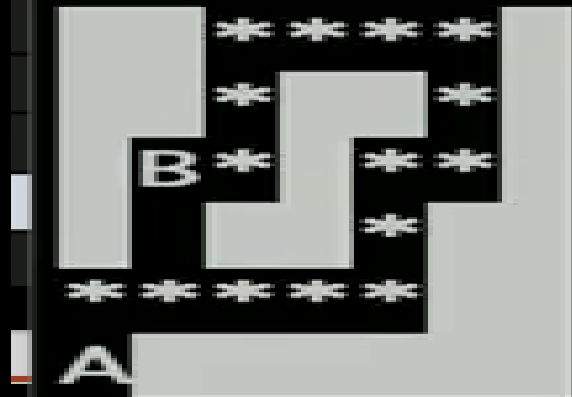
DFS



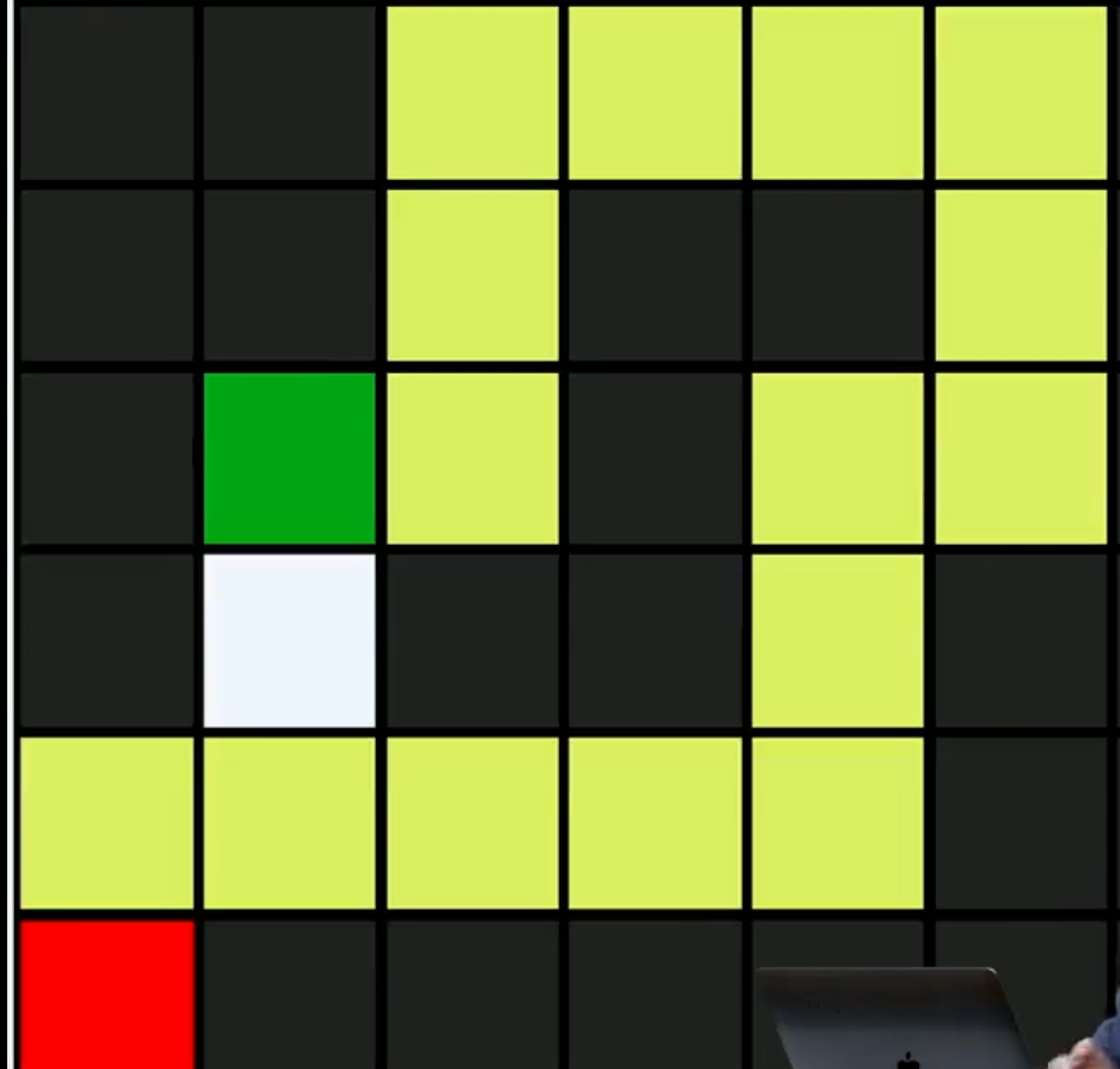
Solving...

States Explored: 17

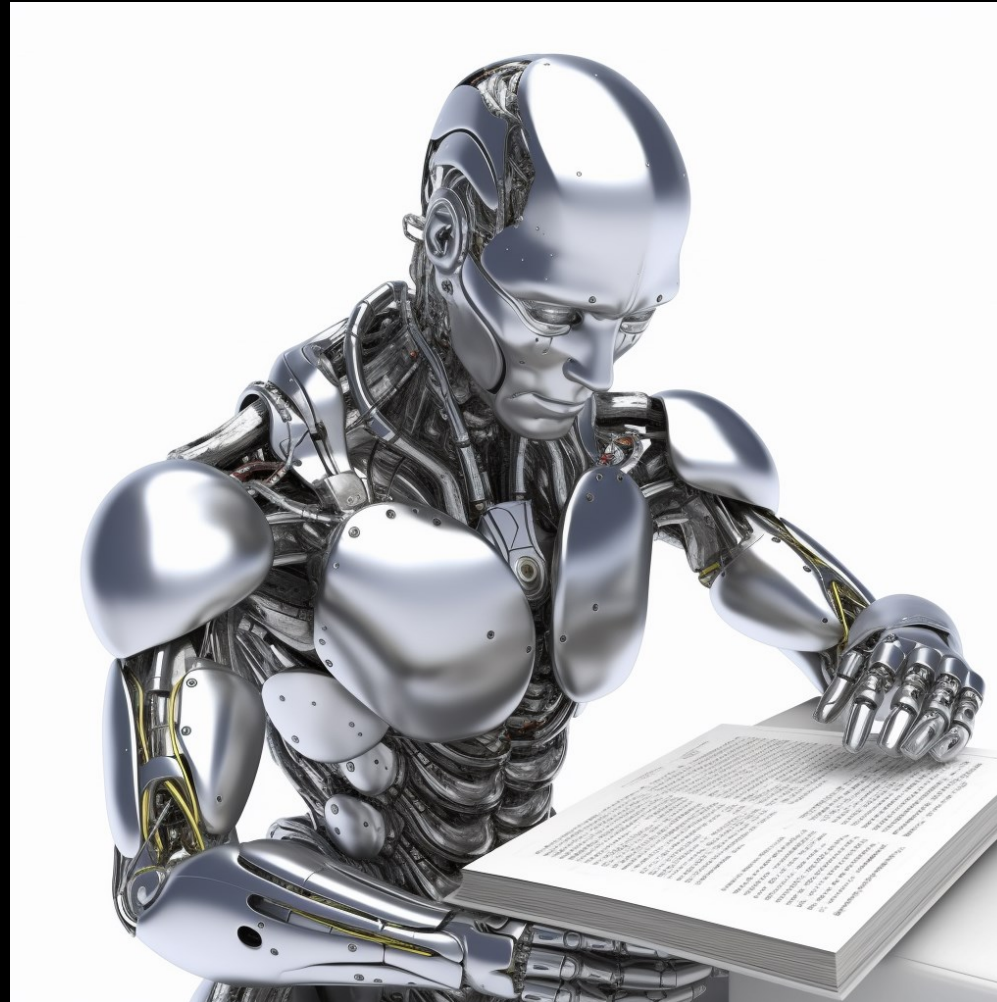
Solution:



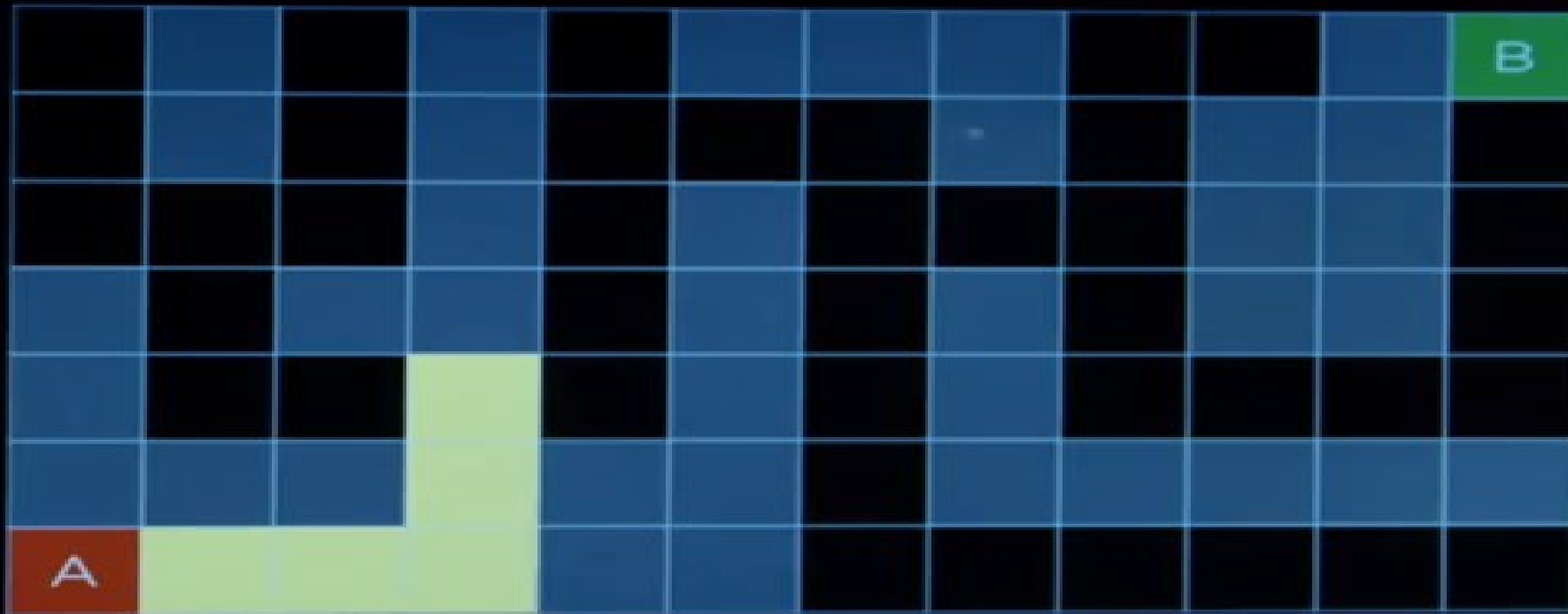
DFS



COMO DEIXAR O ALGORITMO MAIS INTELIGENTE?



CONSIDERAÇÕES NO PONTO DE DECISÃO



UNINFORMED SEARCH: Algoritmo não possui conhecimento prévio sobre o problema.

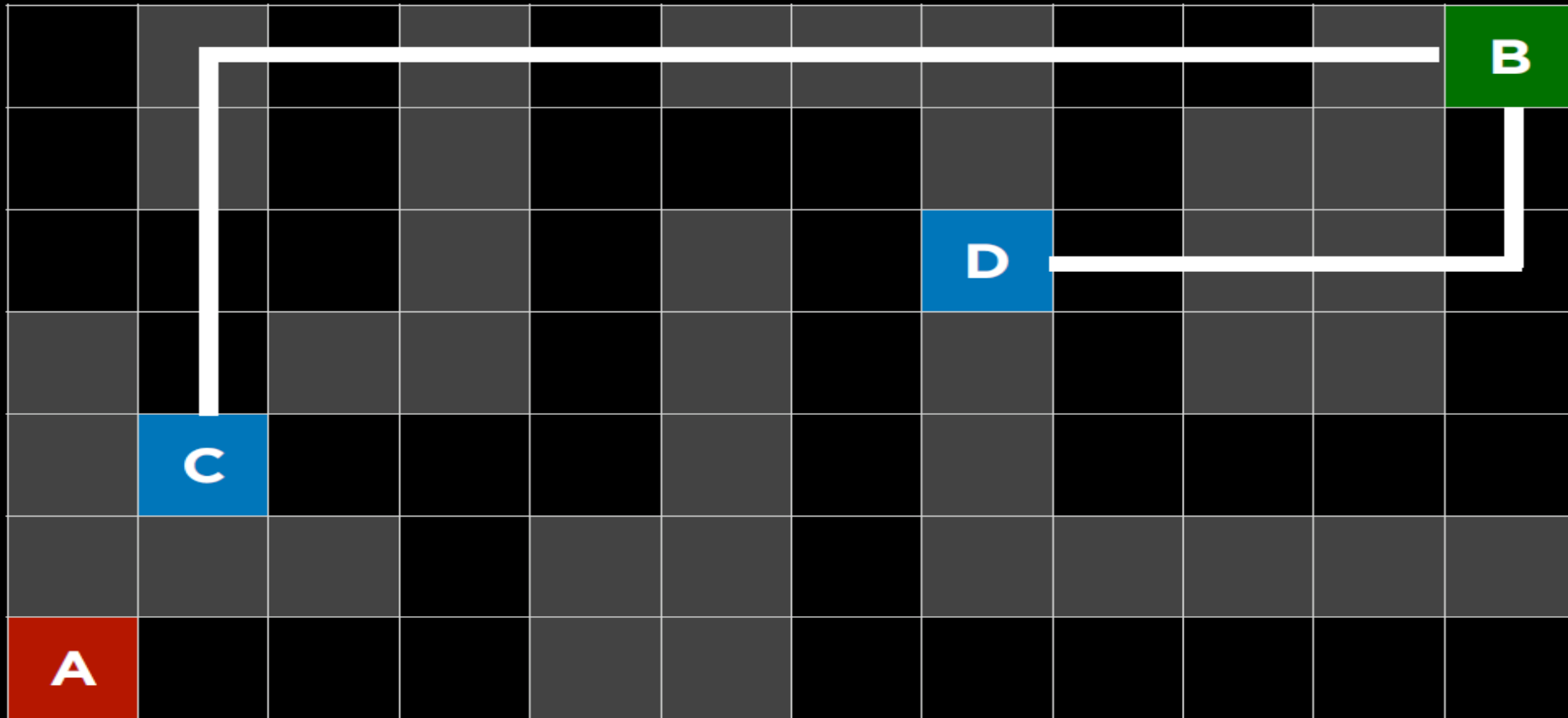
UNINFORMED SEARCH: Algoritmo utiliza algum conhecimento sobre o problema para encontrar a solução.

GREEDY BEST-FIRST SEARCH

- Algoritmo expande para o nó mais perto da solução.
- A expansão é feita através de uma função heurística $h(n)$.

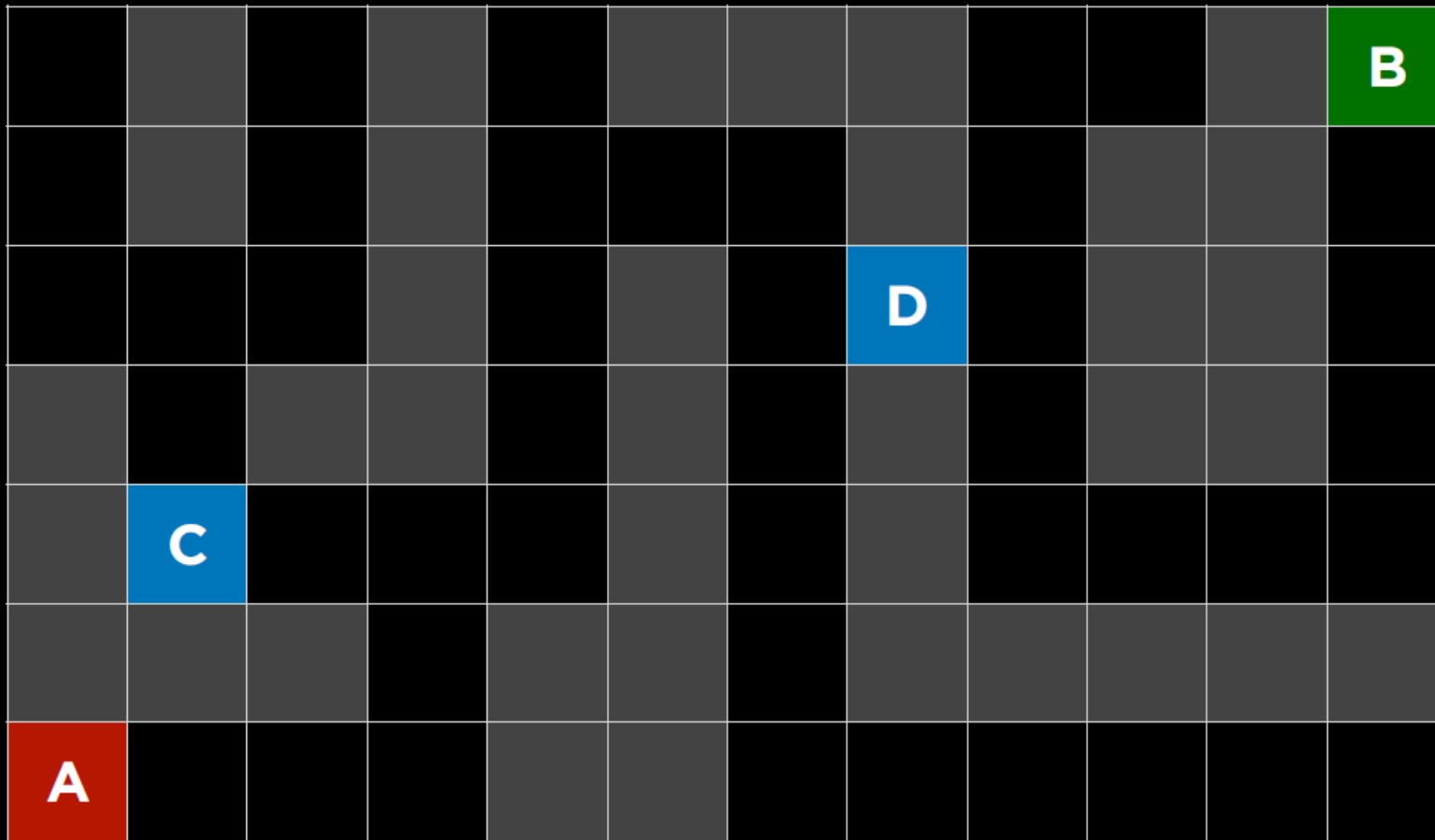
Manhattan Distance

- O nó removido do Frontier e explorado é aquele com a menor distância.



Manhattan Distance

- O nó removido do Frontier e explorado é aquele com a menor distância.



Manhattan Distance

- O nó removido do Frontier e explorado é aquele com a menor distância.

11		9		7				3	2		B
12		10		8	7	6		4			1
13	12	11		9		7	6	5			2
	13			10		8		6			3
	14	13	12	11		9		7	6	5	4
			13			10					
A	16	15	14			11	10	9	8	7	6

Manhattan Distance

- O nó removido do Frontier e explorado é aquele com a menor distância.

	10	9	8	7	6	5	4	3	2	1	B
	11										1
	12		10	9	8	7	6	5	4		2
	13		11						5		3
	14	13	12		10	9	8	7	6		4
			13		11						5
A	16	15	14		12	11	10	9	8	7	6

A* Search

- O nó removido do Frontier e explorado é aquele com a menor valor de $g(n) + h(n)$
- $g(n)$ -> custo
- $h(n)$ -> distância

A* Search

	10	9	8	7	6	5	4	3	2	1	B
	11										1
	12		7+10	8+9	9+8	10+7	11+6	12+5	13+4		2
	13		6+11						14+5		3
	7+14	6+13	5+12		10	9	8	7	15+6		4
			4+13		11						5
A	1+16	2+15	3+14		12	11	10	9	8	7	6