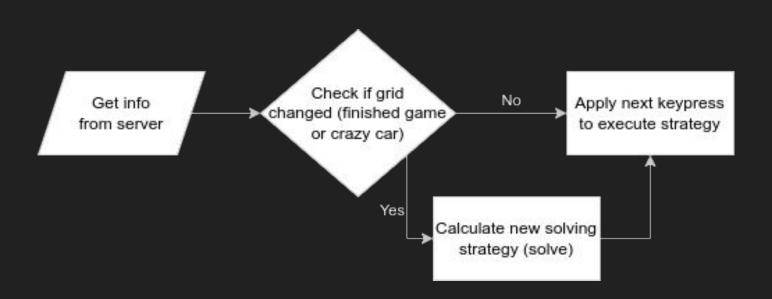
# RUSH HOUR

Artificial Intelligence University of Aveiro

> Pedro Ferreira, nº 98620 Júlia Abrantes, nº 104170 Samuel Teixeira, nº103325

## Flowchart (student.py)



## Flowchart (solve function)

Inirial state: Current grid Pop node from open nodes list Check if node's Return path to node End of loop grid is solved Expand node (get all next possible grids) Add node to open No Skip grids already in path nodes list (sorted by heuristic+cost) Create node for each Was grid already grid (calculate explored? heuristic and cost) Override stored node Yes Is the new node's cost smaller?

Ignore node

### **Heuristics Used**

- H1: Piece Moved is different from the previous one
- H2: Distance from player car to exit
- H3: Number of blocking pieces

### Benchmarks

- Due to A\* and graph search algorithm, a solution can be found quickly and with less steps;
- To implement the A\* algorithm, we used a priority queue for the open nodes (for speed purposes);
- Using the 1st heuristic, we were able to reduce the number of times the cursor selects and unselects pieces (since it favors strategies with sequential moves using the same piece);
- Comparing to the 1st working iteration (using Breadth-first search), this algorithm is about 100x faster.

Level ▼	Avg Moves T	Avg Open Nodes T	Agv Time <b>▼</b>
1	5	0	0
2	6	1	0
3	9	18	0
4	7	28	0
5	11	58	0.01
6	10	22	0
7	13	764	0.26
8	14	3926	1.58
9	15	12296	5.93
10	13	2672	0.83
11	16	20393	11.84
12	21	347	0.12
13	21	846	0.29
14	24	297	0.09
15	24	1654	0.46

Complete benchmarks can be found at results/benchmarks.csv