



# Rushhour

Kyra Kieskamp & Oscar Keur (Boodschappenlijstje)

# Het probleem

**Doel:** De rode auto naar de opening (rechts) op het bord krijgen.

**Probleem:** Andere auto's staan tussen de rode auto en de 'uitgang'.

## Voorwaarden

Rode auto bij de uitgang in zo min mogelijk stappen

De auto's mogen maar één vakje verschuiven per stap

Auto's mogen alleen in hun lengte richting verschuiven



**Oplossing:** De andere auto's (en de rode auto) in meerdere stappen zo verschuiven dat de rode auto uiteindelijk bij de uitgang is.

### Voorwaarden

Rode auto bij de uitgang in zo min mogelijk stappen

De auto's mogen maar één vakje verschuiven per stap

Auto's mogen alleen in hun lengte richting verschuiven



## Data input:



id	dir	length	x	y
	6	6	6	6
t	h	2	3	4
a	v	3	1	3
b	h	2	1	4
c	v	3	1	6
d	v	3	4	4
e	h	2	5	4
f	v	2	1	5
g	h	2	2	5
h	h	2	5	5

## Data in programma:

List of cars

Board matrix

```
List of cars:
0 : ['t', 'h', 2, 2, 3]
1 : ['a', 'v', 3, 0, 2]
2 : ['b', 'h', 2, 0, 3]
3 : ['c', 'v', 3, 0, 5]
4 : ['d', 'v', 3, 3, 3]
5 : ['e', 'h', 2, 3, 4]
6 : ['f', 'v', 2, 4, 0]
7 : ['g', 'h', 2, 4, 1]
8 : ['h', 'h', 2, 5, 4]
Boardmatrix:
['0', '0', 'a', 'b', 'b', 'c']
['0', '0', 'a', '0', '0', 'c']
['0', '0', 'a', 't', 't', 'c']
['0', '0', '0', 'd', 'e', 'e']
['f', 'g', 'g', 'd', '0', '0']
['f', '0', '0', 'd', 'h', 'h']
```

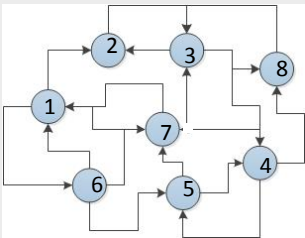


# Visualisatie



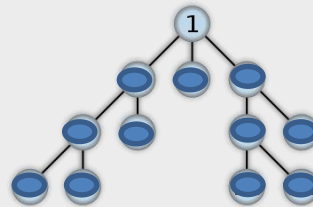
# Methodes en algoritme

## Random



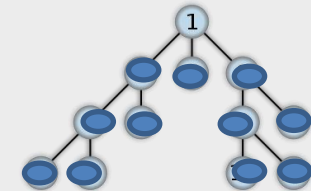
1. Start bij begin board
2. Bepaal alle mogelijke moves voor het board
3. D.m.v. random functie, kies één move
4. Maak move
5. Is dit de oplossing?
  1. Ja → stoppen
  2. Nee → naar stap 2.

## Breadth-first



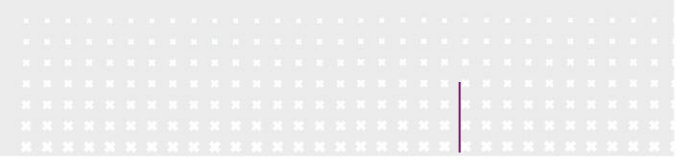
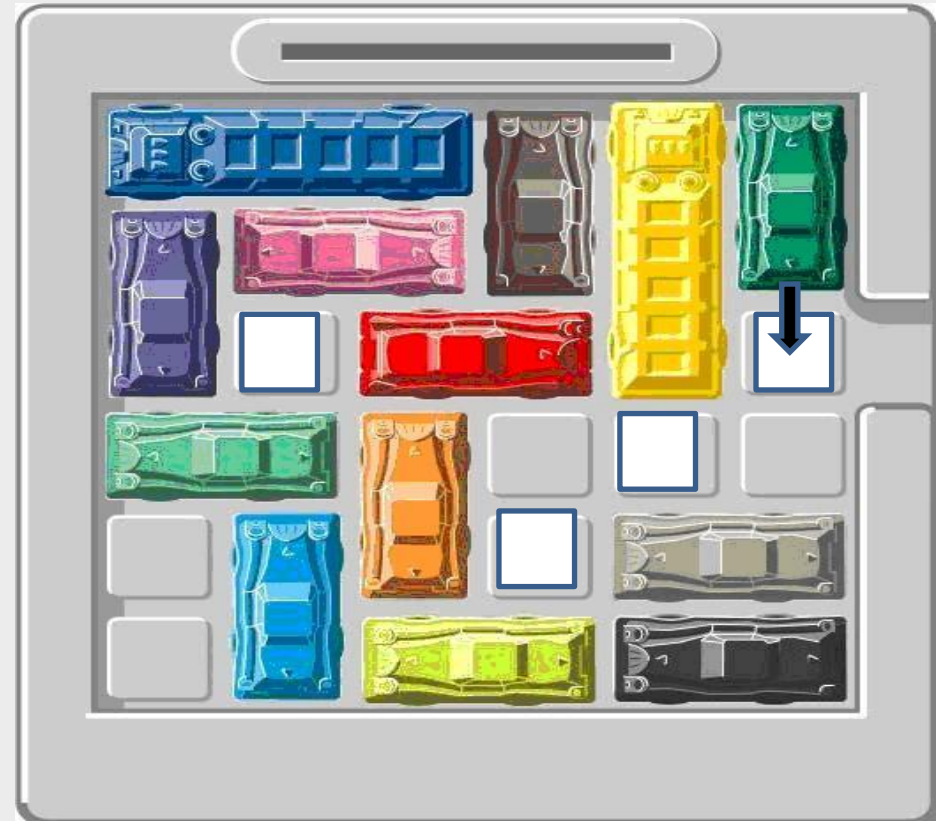
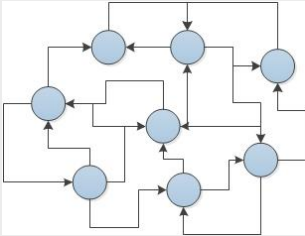
1. Start bij begin board
2. Bepaal alle mogelijke moves voor alle boards op dezelfde diepte
3. Voor elke move:
  1. Maak move
  2. Is dit de oplossing?
    1. Ja → stoppen
    2. Nee → doorgaan
  3. Huidige staat onthouden.
4. Naar boards in de volgende diepte.

## Depth-first



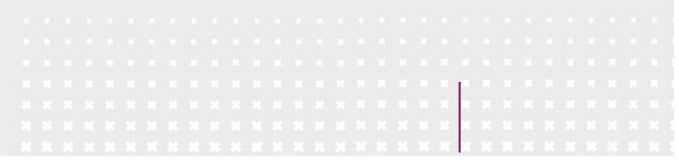
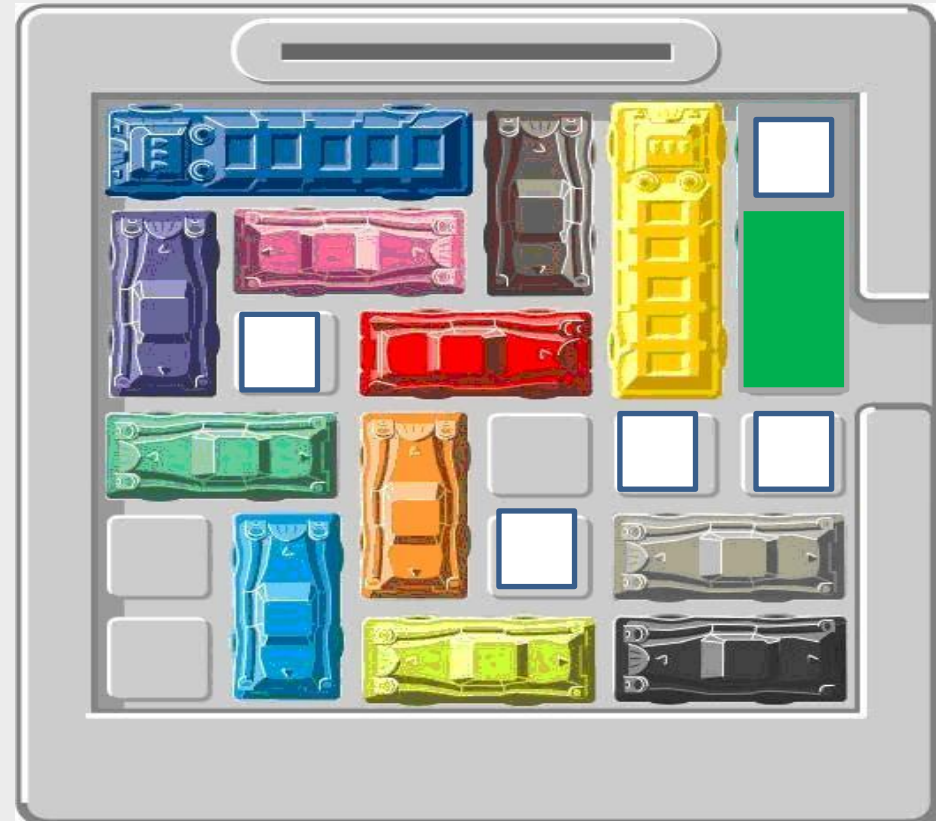
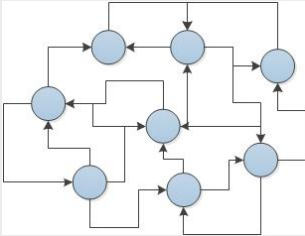
1. Start bij begin board
2. Bepaal alle mogelijke moves voor het board
3. Zijn er moves?
  1. Ja → Neem de eerste
  2. Nee → Ga een niveau 'omhoog'
4. Is dit de oplossing?
  1. Ja → stoppen
  2. Nee → Ga naar stap 2

Random



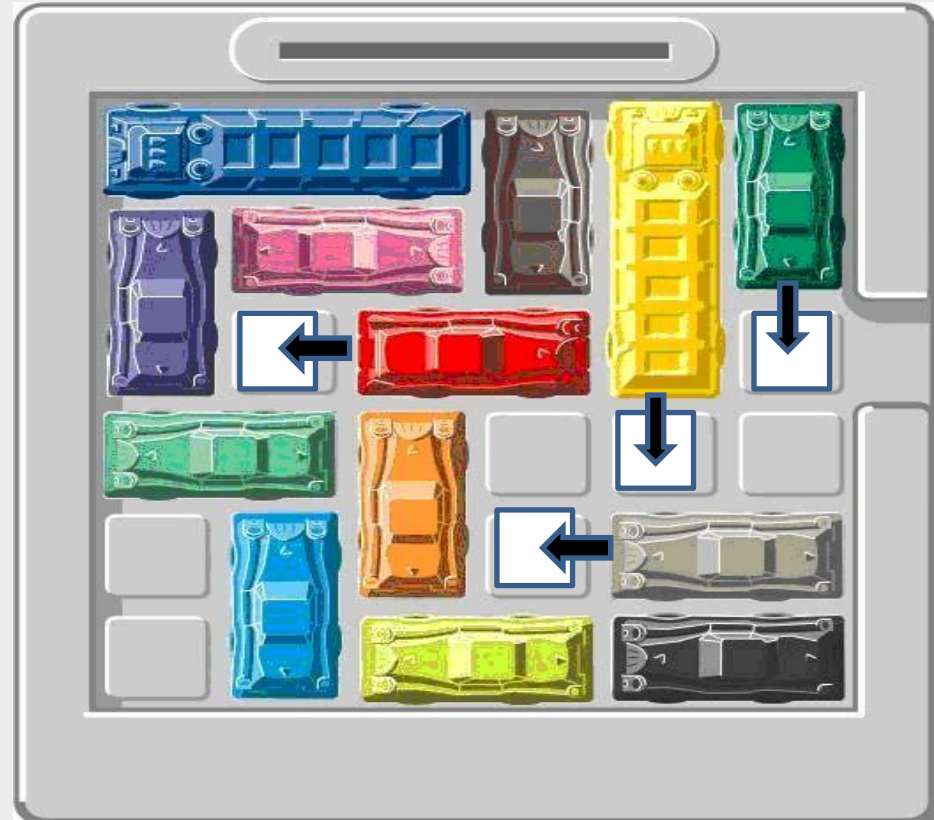
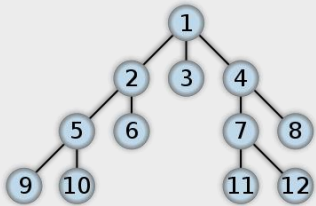


## Random

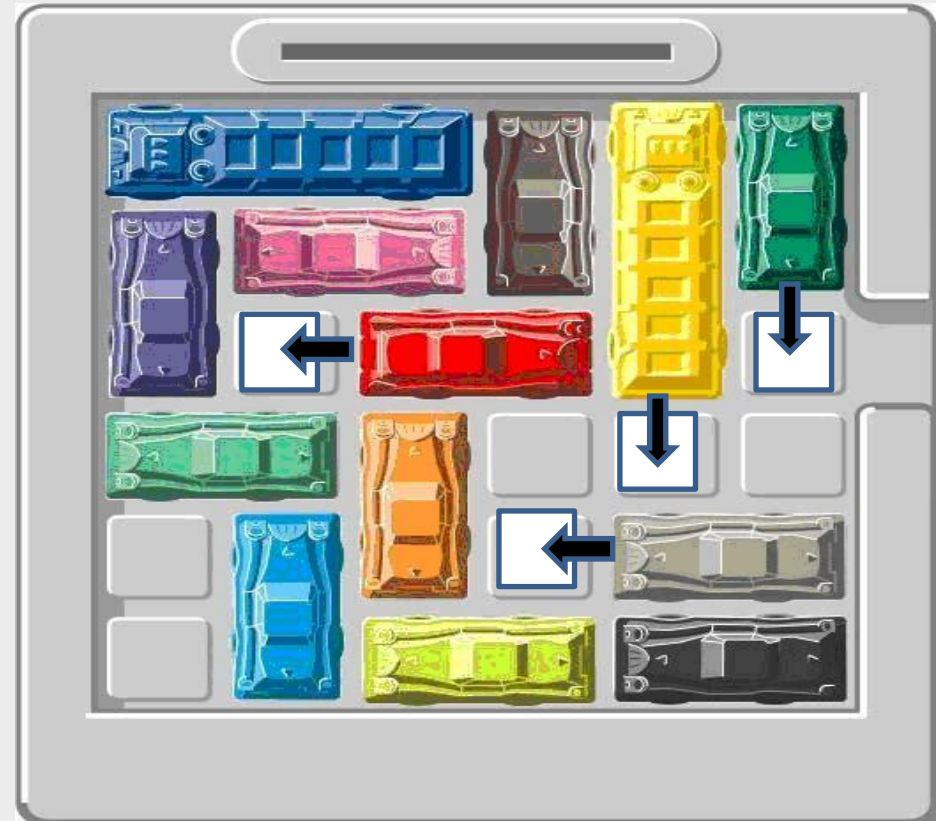
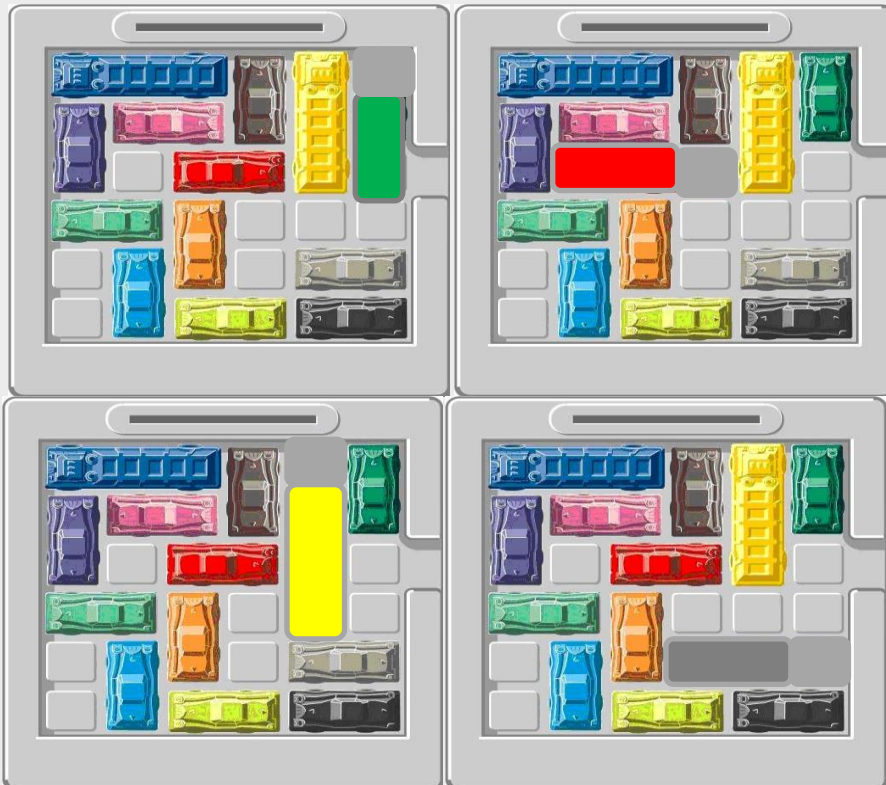
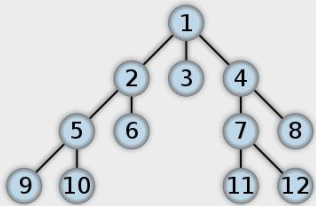




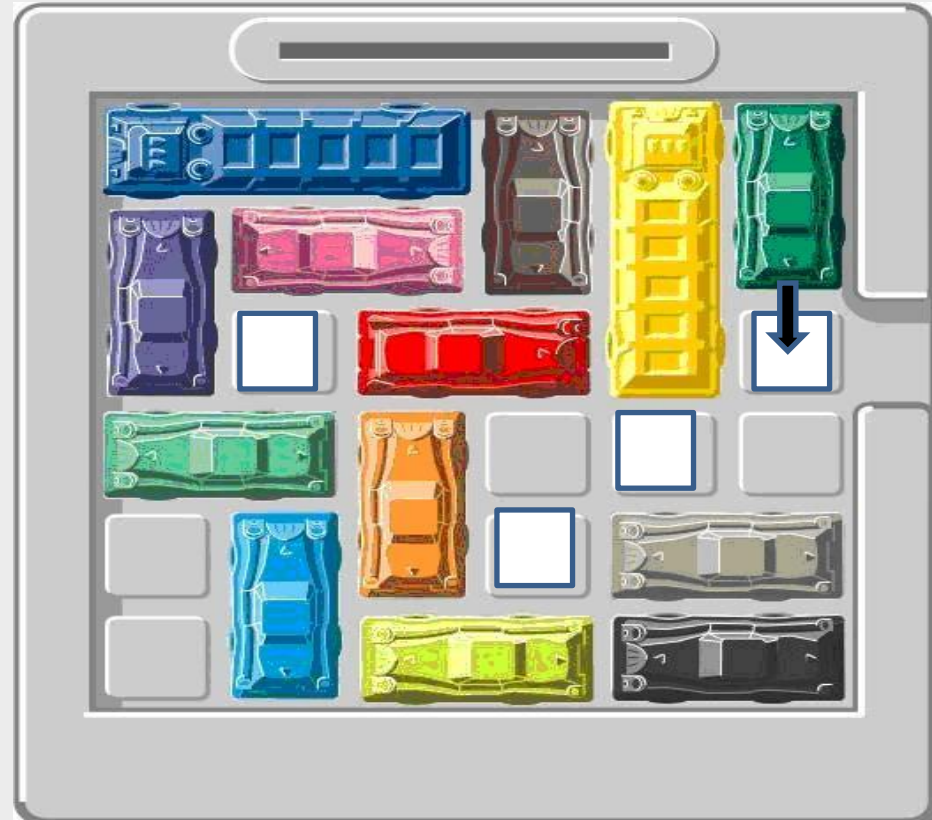
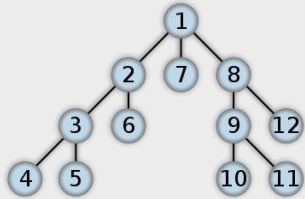
## Breadth-first



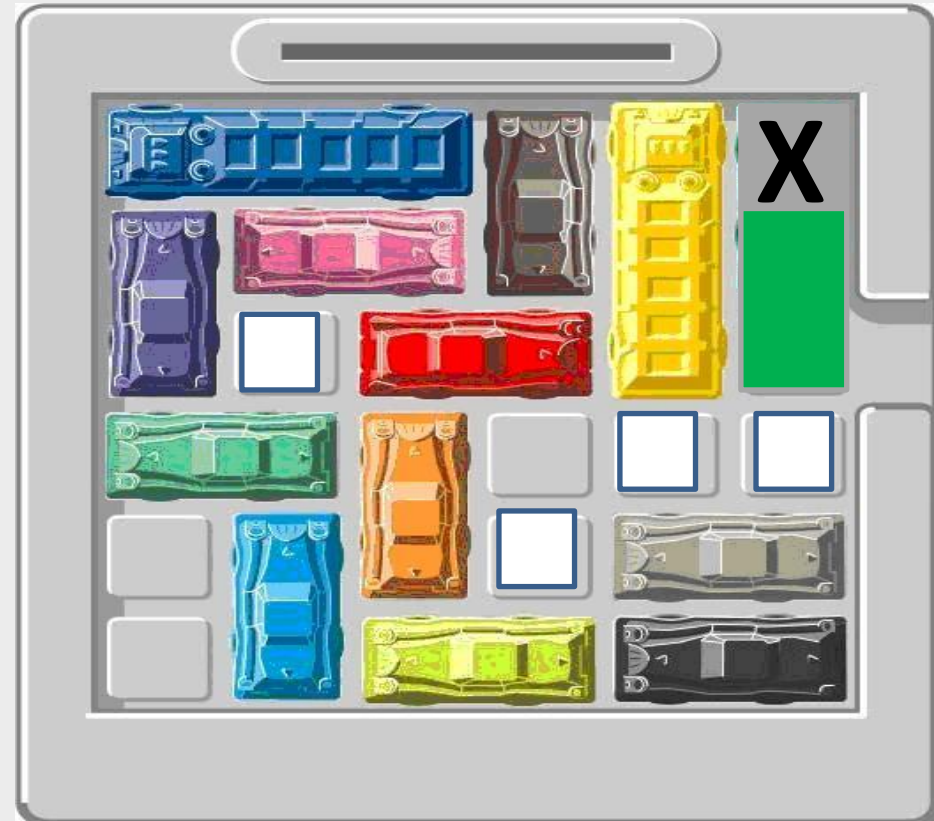
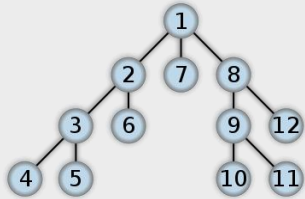
## Breadth-first



## Depth-first

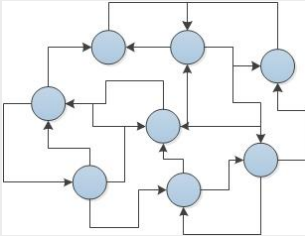


## Depth-first

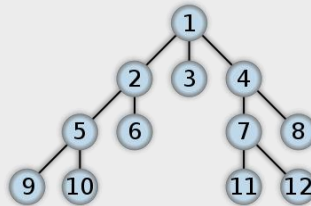


# Resultaten

## Random

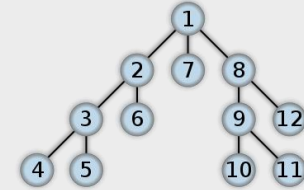


## Breadth-first



Board	Minimaal moves (absoluut)	Boards (cumulatief)
test	6	1871

## Depth-first



Current results:  
questionable

Board	Minimaal moves gevonden
1	51
2	71
3	37
4	323
5	276
6	95
7	818
Test	6

# Conclusie

Enorme statespace:

- Breath first werkt wel, is nog langzaam
- Breadth first zal kortste oplossing vinden → lang draaien
- Snelle random solver: niet gegarandeerd minimale stappen

