

Nurikabe

Nome - Matricula

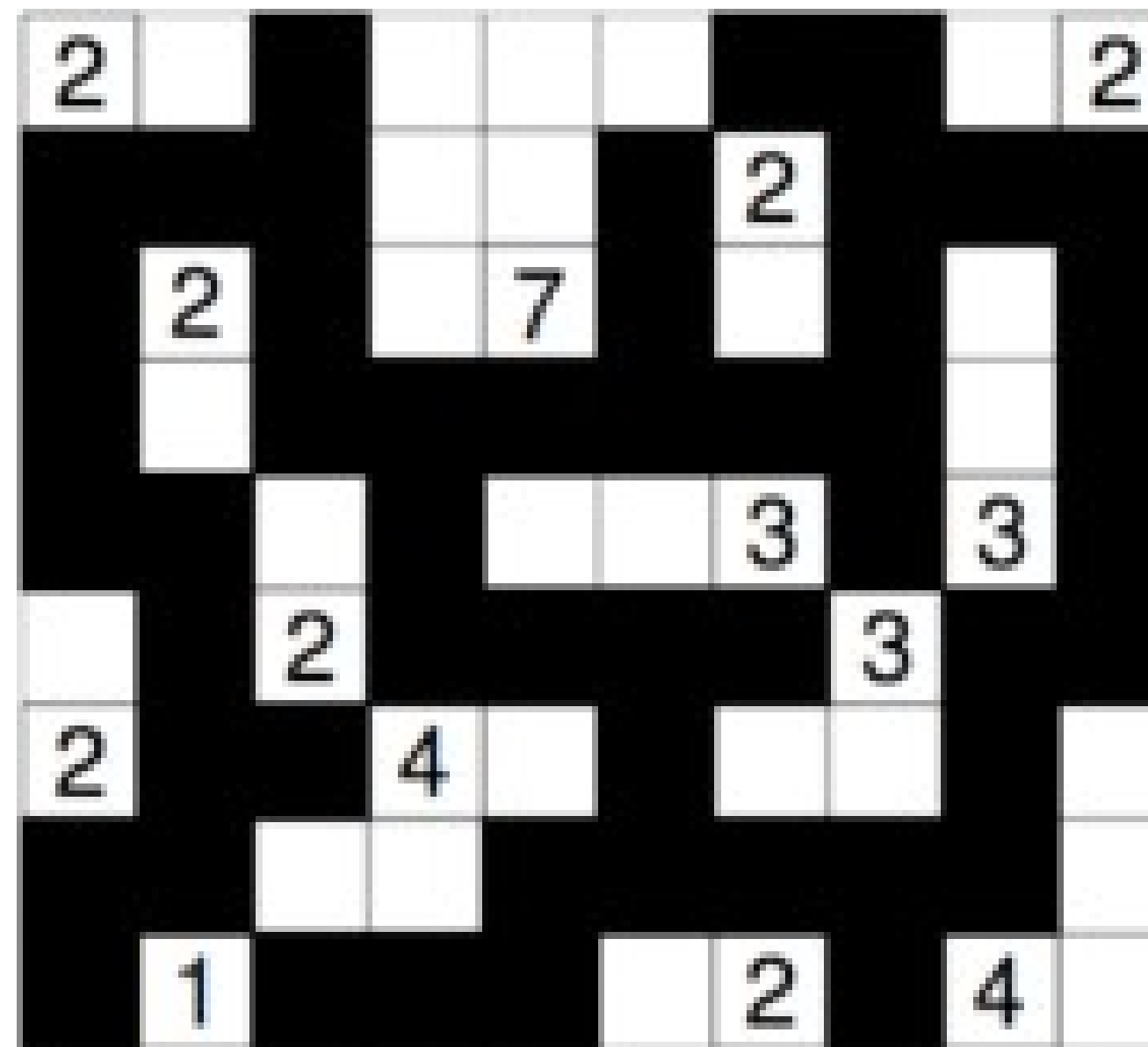
Daniel dos Santos Barros de Sousa - 211030980

Lucas Soares Rodrigues - 190126248

Marcos Gabriel Tavares - 170041042

Rodrigo Tiago Costa Lima - 180037242

Vinicius Vieira de Souza - 170115500

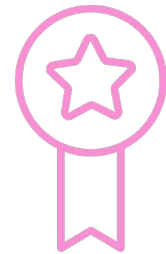


Do que se trata o domínio.

Nurikabe é um jogo de quebra-cabeça de determinação binária, onde se deve decidir para cada célula se ela é branca ou preta, tendo sido inventado no Japão.

Ele é um quebra-cabeça de formação de ilhas, sendo resolvido dividindo-se entre pistas com paredes, onde o objetivo é criar ilhas dividindo pistas com paredes de acordo com as seguintes regras:

Ilhas



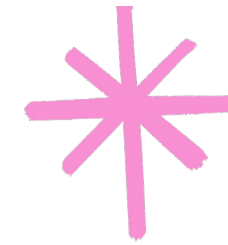
- Uma ilha deve ser formada por células brancas contínuas conectadas na quantidade indicada na célula branca (Incluindo a célula numerada).
- Cada ilha contém exatamente uma pista.
- Duas ilhas não podem estar conectadas
- Todas as ilhas estão isoladas umas das outras horizontal e verticalmente.
- Não se pode preencher células contendo números.
- As ilhas devem ser separadas por células pretas.

Paredes



- As células pretas não podem ser vinculadas para serem quadradas 2×2 ou maiores. (Regra não garantida neste domínio)
- Quando concluídas, todas as paredes formam um caminho contínuo.
- Todas as células pretas devem ser conectadas para formar uma parede contínua.

Análise crítica



- Diferente do que acontece na prática do jogo Nurikabe, o domínio em questão decide pintar as células brancas ao invés das células pretas. Acredita-se que essa decisão foi tomada com o intuito de “ganhar tempo” já que é comum, no jogo de Nurikabe, uma menor quantidade de células brancas.
- Devido ao que foi mencionado anteriormente. Como o domínio funciona pintando apenas as ilhas/grupos, não há garantia de que a solução obedeça a regra da parede 2x2 (Não deve haver uma parede preta 2x2).

Execução dos problemas

Existem 40 problemas no total. 20 são para a track OPT e 20 para a track SAT. Os problemas para a track OPT são em geral menores que os problemas da track SAT. Também realizamos um round de experimentos utilizando lama-first e madagascar com as flags padrão e os denominamos de Agile, feitos com os problemas SAT. Todos os experimentos foram realizados com um limite de tempo de 300s (5 Minutos).

OPT
 Time limit = 300s
 -S 1 -P 0

Problem/Planner	Fast-Downward	Madagascar	Julia	
1	0.232s	0.020s	21.762s	3x3 2 grupos
2	0.217s	0.021s	21.946s	3x3 3 grupos
3	0.512s	0.072s	21.820s	4x4 2 grupos
4	0.631s	0.159s	22.273s	4x4 3 grupos
5	2.444s	0.736s	22.363s	5x5 2 grupos
6	3.309s	2.034s	24.959s	5x5 5 grupos
7	21.999s	8.189s	25.444s	6x6 5 grupos
8	25.721s	84.291s	29.499s	6x6 7 grupos
9	123.631s	TLE	34.892s	7x7 6 grupos
10	126.599s	TLE	35.085s	7x7 8 grupos
11	TLE	TLE	TLE	8x8 8 grupos
12	TLE	TLE	98.998s	8x8 10 grupos
13	TLE	TLE	TLE	9x9 10 grupos
14	TLE	TLE	TLE	9x9 10 grupos
15	TLE	TLE	TLE	10x10 15 grupos
16	TLE	TLE	TLE	10x10 13 grupos
17	TLE	TLE	TLE	11x11 14 grupos
18	TLE	TLE	TLE	11x11 17 grupos
19	TLE	TLE	TLE	12x12 21 grupos
20	TLE	TLE	TLE	12x12 19 grupos
Total	10	8	11	

SAT

Time limit = 300s

flags	seq-sat-fdss-2018	seq-sat-lama-2011	
Problem/Planner	Fast-Downward	Fast-Downward	
1	1 plano - OPT 32	2 planos - OPT 32	6x6 6 grupos
2	3 planos - OPT 38	5 planos - OPT 38	6x6 7 grupos
3	6 planos - 42	6 planos - OPT 42	7x7 7 grupos
4	4 planos - 52	4 planos - 52	7x7 7 grupos
5	7 planos - 56	5 planos - 56	8x8 9 grupos
6	8 planos - 59	3 planos - 62	8x8 10 grupos
7	6 planos - 70	6 planos - 68	9x9 11 grupos
8	7 planos - 75	1 plano - 94	9x9 11 grupos
9	6 planos - 95	6 planos - 88	10x10 13 grupos
10	5 planos - 93	3 planos - 101	10x10 13 grupos
11	1 plano - 122	TLE	11x11 16 grupos
12	7 planos - 103	3 planos - 112	11x11 14 grupos
13	2 planos - 134	TLE	12x12 19 grupos
14	TLE	TLE	12x12 14 grupos
15	TLE	TLE	13x13 20 grupos
16	TLE	TLE	13x13 25 grupos
17	TLE	TLE	14x14 23 grupos
18	TLE	TLE	14x14 27 grupos
19	TLE	TLE	15x15 30 grupos
20	TLE	TLE	15x15 29 grupos
Total	13	11	

Agile
 Time limit = 300s
 -S 5 -P 2

Problem/Planner	Fast-Downward	Madagascar	
1	1.091s	2.177s	6x6 6 grupos
2	0.995s	12.731s	6x6 7 grupos
3	2.234s	18.930s	7x7 7 grupos
4	6.716s	TLE	7x7 7 grupos
5	5.088s	TLE	8x8 9 grupos
6	4.698s	TLE	8x8 10 grupos
7	8.055s	TLE	9x9 11 grupos
8	TLE	TLE	9x9 11 grupos
9	16.913s	TLE	10x10 13 grupos
10	47.464s	TLE	10x10 13 grupos
11	TLE	TLE	11x11 16 grupos
12	TLE	TLE	11x11 14 grupos
13	TLE	TLE	12x12 19 grupos
14	TLE	TLE	12x12 14 grupos
15	TLE	TLE	13x13 20 grupos
16	TLE	TLE	13x13 25 grupos
17	TLE	TLE	14x14 23 grupos
18	TLE	TLE	14x14 27 grupos
19	TLE	TLE	15x15 30 grupos
20	TLE	TLE	15x15 29 grupos
Total	9	3	



PROBLEMA

```
(define (problem random-3x3-2)
  (:domain nurikabe)
  (:objects
    pos-0-0 pos-0-1 pos-0-2 pos-1-0 pos-1-1 pos-1-2 pos-2-0 pos-2-1 pos-2-2 - cell
    n1 n2 - num
    g0 g1 - group
  )
  (:init
    (NEXT n0 n1)
    (NEXT n1 n2)
    (CONNECTED pos-0-0 pos-1-0)
    (CONNECTED pos-0-0 pos-0-1)
    (CONNECTED pos-0-1 pos-1-1)
    (CONNECTED pos-0-1 pos-0-2)
    (CONNECTED pos-0-1 pos-0-0)
    (CONNECTED pos-0-2 pos-1-2)
    (CONNECTED pos-0-2 pos-0-1)
    (CONNECTED pos-1-0 pos-2-0)
    (CONNECTED pos-1-0 pos-1-1)
    (CONNECTED pos-1-0 pos-0-0)
    (CONNECTED pos-1-1 pos-2-1)
    (CONNECTED pos-1-1 pos-1-2)
    (CONNECTED pos-1-1 pos-0-1)
    (CONNECTED pos-1-1 pos-1-0)
    (CONNECTED pos-1-2 pos-2-2)
    (CONNECTED pos-1-2 pos-0-2)
    (CONNECTED pos-1-2 pos-1-1)
    (CONNECTED pos-2-0 pos-2-1)
    (CONNECTED pos-2-0 pos-1-0)
    (CONNECTED pos-2-1 pos-2-2)
    (CONNECTED pos-2-1 pos-1-1)
    (CONNECTED pos-2-1 pos-2-0)
    (CONNECTED pos-2-2 pos-1-2)
    (CONNECTED pos-2-2 pos-2-1)
    (robot-pos pos-0-0)
    (moving)
    (SOURCE pos-0-0 g0)
    (SOURCE pos-2-1 g1)
    (available pos-0-2)
    (available pos-1-2)
    (part-of pos-1-0 g0)
    (part-of pos-0-1 g0)
    (part-of pos-2-2 g1)
    (part-of pos-1-1 g1)
    (part-of pos-2-0 g1)
    (remaining-cells g0 n2)
    (remaining-cells g1 n1)
  )
  (:goal
    (and
      (group-painted g0)
      (group-painted g1)
    )
  )
)
```

X SOLUÇÃO

```
(start-painting pos-0-0 g0 n2 n1)
(move-painting pos-0-0 pos-0-1 g0 n1 n0)
(end-painting g0)
(move pos-0-1 pos-1-1)
(move pos-1-1 pos-2-1)
(start-painting pos-2-1 g1 n1 n0)
(end-painting g1)
; cost = 7 (unit cost)
```

Ações e
predicados do
domínio.



Predicados do domínio

- (NEXT ?n1 - num ?n2 - num) - Um contador decrescente
- (CONNECTED ?c - cell ?c2 - cell) - Define o grafo
- (SOURCE ?x - cell ?g - group) - A posição inicial a partir da qual o robô deve começar a pintar
- (painted ?r - cell) - Uma célula foi pintada
- (available ?x - cell) - Uma célula não pertence a nenhum grupo
- (part-of ?x - cell ?y - group) - Se uma célula “pertence” a um grupo. Isso acontece quando uma célula adjacente pertence a tal grupo
- (blocked ?x - cell) - Uma célula pertence a dois grupos diferentes, portanto não pode ser pintada de nenhuma cor
- (remaining-cells ?x - group ?y - num) - Quantas células faltam pintar para cada grupo
- (robot-pos ?x - cell) - Posição do robô
- (moving) - O robô atualmente não está pintando
- (painting ?g - group) - O robô está atualmente pintando um grupo
- (group-painted ?g - group) - O robô pintou um grupo

Ações do domínio

move

start-painting

move-painting

end-painting

Move

```
(:action move
  :parameters (?from - cell ?to - cell)
  :precondition
  (and
    (CONNECTED ?from ?to)
    (moving)
    (not (painted ?to))
    (robot-pos ?from)
  )
  :effect
  (and
    (robot-pos ?to)
    (not (robot-pos ?from))
  )
)
```

Parâmetro:

- Célula de origem e célula de destino

Precondição:

- As células de origem e de destino estão conectadas
- O robô não está pintando
- A célula de destino não está pintada
- O robô está na célula de origem

Efeito:

- O robô está na célula de destino
- O robô não está na célula de origem

Start-Painting

```
(:action start-painting
  :parameters (?c - cell ?g - group ?n1 - num ?n2 - num)
  :precondition
  (and
    (NEXT ?n2 ?n1)
    (SOURCE ?c ?g)
    (moving)
    (robot-pos ?c)
    (remaining-cells ?g ?n1)
  )
  :effect
  (and
    (not (moving))
    (painting ?g)
    (painted ?c)
    (remaining-cells ?g ?n2)
    (not (remaining-cells ?g ?n1))
  )
)
```

Parâmetro:

- Célula que será pintada, grupo, n1, n2

Precondição:

- O robô não está pintando
- O robô está na célula que será pintada
- Falta pintar n1 células neste grupo

Efeito:

- O robô está pintando
- O robô está pintando este grupo
- A célula foi pintada
- Falta pintar n2 células no grupo

Move-Painting

```
[ :action move-painting
  :parameters {?from - cell ?to - cell ?g - group ?n1 - num ?n2 - num}
  :precondition
  (and
    (NEXT ?n2 ?n1)
    (CONNECTED ?from ?to)
    (not (painted ?to))
    (not (blocked ?to))
    (painting ?g)
    (remaining-cells ?g ?n1)
    (robot-pos ?from)
  )
  :effect
  (and
    (robot-pos ?to)
    (not (robot-pos ?from))
    (painted ?to)
    (remaining-cells ?g ?n2)
    (not (remaining-cells ?g ?n1))
    (forall (?cadj - cell)
      (when
        (and
          (CONNECTED ?to ?cadj)
          [available ?cadj]
        )
        (and
          (not [available ?cadj])
          [part-of ?cadj ?g]
        )
      )
    )
    (forall (?cadj - cell)
      (when
        (and
          (CONNECTED ?to ?cadj)
          [not [available ?cadj]]
          [not [part-of ?cadj ?g]]
        )
        (blocked ?cadj)
      )
    )
  )
]
```

Parâmetro:

- Célula de origem, célula de destino, grupo, n1, n2

Precondição:

- As células de origem e de destino estão conectadas
- A célula de destino não foi pintada
- A célula de destino não pertence a dois grupos diferentes
- O robô está pintando o grupo
- Falta pintar n1 células neste grupo
- O robô está na célula de origem

Efeito:

- O robô está na célula de destino
- O robô não está na célula de origem
- A célula de destino foi pintada
- Falta pintar n2 células neste grupo
- Não falta pintar n1 células no grupo

End-Painting

```
{:action end-painting
  :parameters (?g - group)
  :precondition
  (and
    (painting ?g)
    (remaining-cells ?g n0)
  )
  :effect
  (and
    (not (painting ?g))
    (moving)
    (group-painted ?g)
  )
}
```

Parâmetro:

- Grupo

Precondição:

- O robô está pintando este grupo
- Falta pintar n0 células neste grupo

Efeito:

- O robô não está pintando este grupo
- O robô não está pintando
- O robô pintou este grupo