

# Trabalho Prático nº1 – Agentes

Licenciatura em Engenharia Informática – 3º ano

Inteligência Artificial

Turma 1

Autores:

Duarte Garcês al70860

Tomás Antunes al70577

Docente

Paulo Moura Oliveira

## Índice

Introdução.....	3
Etapas e resolução: .....	4
Conclusão .....	14
Referências bibliográficas .....	14

## Introdução

O presente relatório é elaborado no âmbito da unidade curricular Inteligência Artificial, com vista à conclusão do trabalho prático nº1 de Agentes. Pretende-se promover a aquisição de conhecimentos e desenvolvimento de competências fundamentais à modelação e simulação computacional de sistemas com agentes racionais utilizando a ferramenta NetLogo.

## Etapas e resolução:

```

to setup
  clear-all
  __clear-all-and-reset-ticks ;para o gráfico resetar a cada setup|
  set head 0 ;usado para saber a direção em que o serpentina segue (cima ou para baixo)

  set quantidade 0
  set num_toupeiras 0
  set num_nascimentos 0 ;declaração do valor 0 inicial a todas as variáveis usadas
  set num_mortes 0
  set age 0

  ask patches [
    set pcolor brown ;fundo castanho
  ]

  create-plantadores 1 [
    set shape "turtle"
    set color black ;criação do planta-relva
    setxy -16 -16
    set size 2.5
  ]

  ask plantadores [
    if modo_de_plantar = "Serpentina" [
      set heading 0
    ]
    if modo_de_plantar = "Para_a_frente" [ ;definição da direção de acordo com cada modo de plantação
      set heading one-of [0 90 180 270]
    ]
    if modo_de_plantar = "Manutenção" [
      set heading 0
    ]
  ]
  reset-ticks
end

to Frente_20
  let i 0
  while [i < 20] [ ;andar 20 passos de cada vez o plantador
    plantar
    set i i + 1
  ]
end

```

Figura 1 – Setup



Figura 2 – Limpar campo

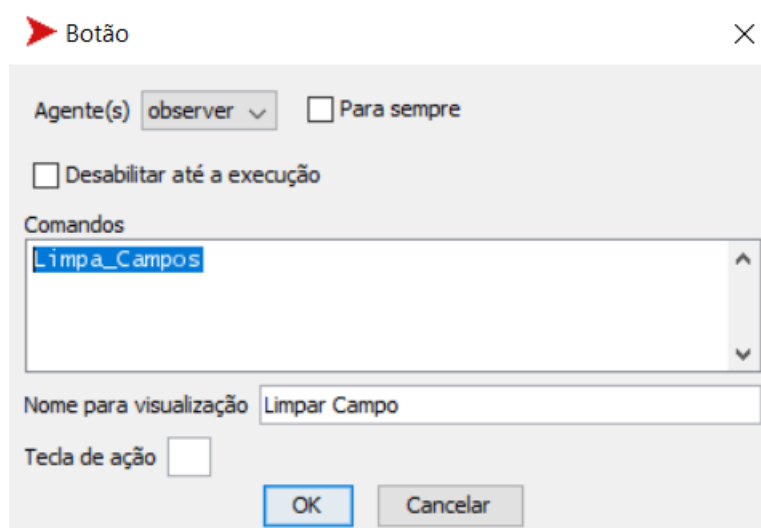


Figura 2.1 – Configuração do botão Limpar Campo

```
to Limpa_Campos
  ask patches[
    set pcolor brown
  ]
end
```

Figura 2.2 – Código de Limpar campo

- 1) Altere a localização da origem do ambiente (0,0) para o canto inferior esquerdo.



Figura 3 – Origem do Ambiente (-16,-16)

- 2) Insira um botão que permite ao planta-relva fazer a plantação de todo o campo.



Figura 4 – Botão para fazer a Plantação de todo o campo

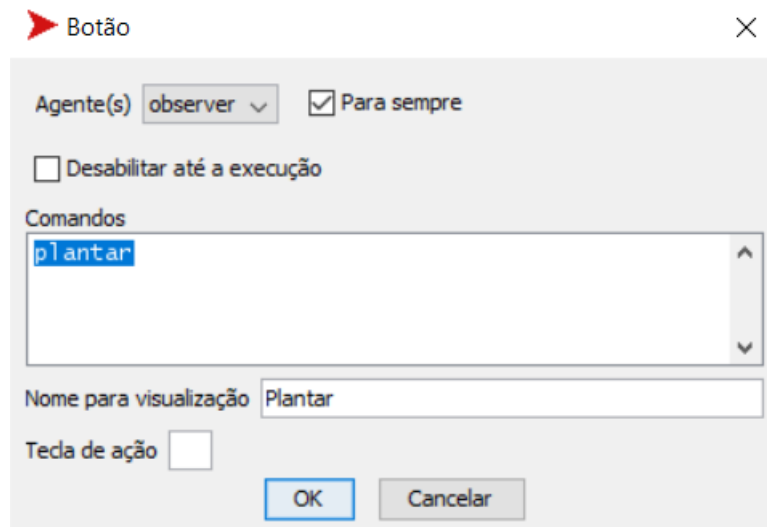


Figura 4.1 – Configuração do botão plantar

```
if modo_de_plantar = "Serpentina"
[
  ask plantadores [
    if [pcolor] of patch-ahead 1 = brown [
      ask patch-here[
        set pcolor lime
      ]
      forward 1
    ]
    if [pcolor] of patch-ahead 1 = lime [
      set head head + heading
      set heading 90
      ask patch-here[
        set pcolor lime
      ]
    ]

    forward 1

    if head = 0 [
      right 90
    ]
    if head = 180 [
      left 90
    ]
    set head 0
  ]
]
```

Figura 4.2 – Código de Plantar “Serpentina”

```

if modo_de_plantar = "Para_a_frente" [
  ask plantadores [
    if [pcolor] of patch-ahead 1 = brown [
      ask patch-here[
        set pcolor lime
      ]
      forward 1
    ]
    if [pcolor] of patch-ahead 1 = lime [
      ask patch-here[
        set pcolor lime
      ]
      forward 1
    ]
  ]
]
]

```

Figura 4.3 – Código de Plantar “Para\_a\_frente”

A plantação pode ser feita de (pelo menos) dois modos:

- Para\_a\_frente: o planta-relva é inicializado com uma direção aleatória entre quatro ângulos (0º, 90º, 180º, 270º).
- Serpentina: O planta-relva é inicializado com uma direção de ângulo 0º até atingir o limite do campo, invertendo então o sentido.





Figura 5 – “Frente\_20”

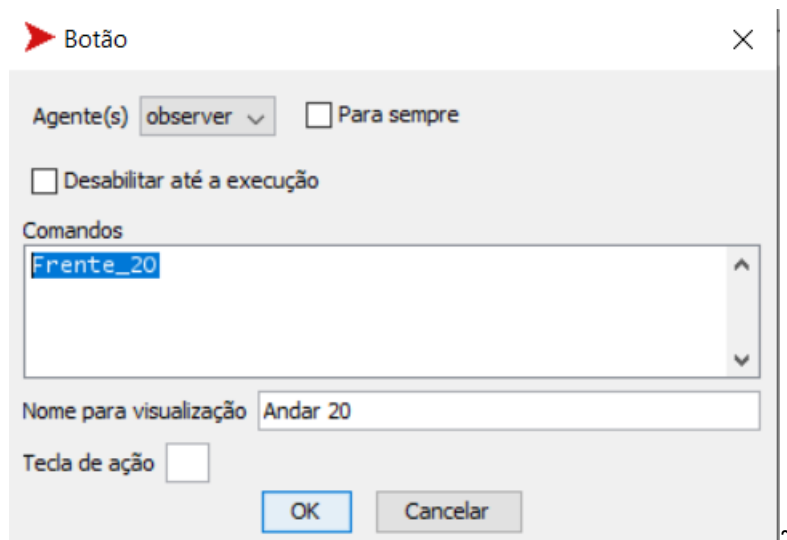


Figura 5.1 – Configuração botão “Andar 20”

```
to Frente_20
  let i 0
  while [i < 20] [
    plantar
    set i i + 1
  ]
end
```

Figura 5.2 – Código de Plantar “Andar 20”

- 3) Seleccionador que permite escolher o modo de funcionamento do planta-relva (agente). O seleccionador comporta 3 modos: “Para\_a\_frente”, “Serpentina”, “Manutenção”.

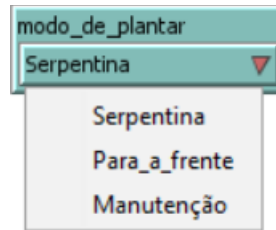


Figura 6 – Seleccionador Modo de Plantar

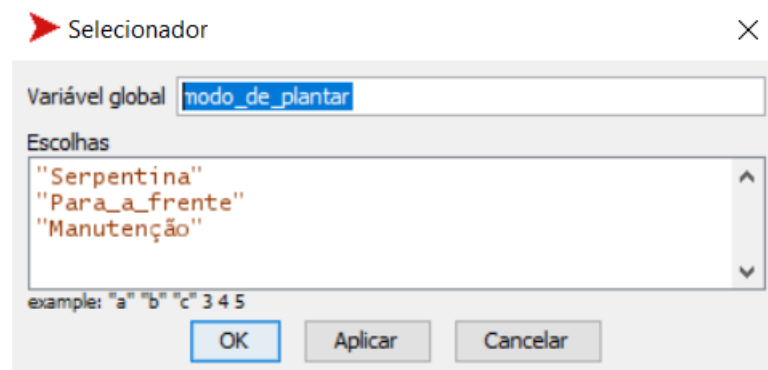


Figura 6.1 Configuração seleccionador Modo de Plantar

- 4) Altere o modelo de forma a que existam duas espécies de agentes: planta-relva e toupeira.

```
breed [plantadores plantador]
breed [toupeiras toupeira]
```

Figura 7 – As duas espécies de ‘Agentes’

- 5) Insira um botão Inserir\_Toupeira, cujo programa permite inserir uma toupeira no campo numa localização e com direção aleatórias. Cada vez que o botão é pressionado é inserida uma toupeira.



Figura 8 – Botão Inserir Toupeira

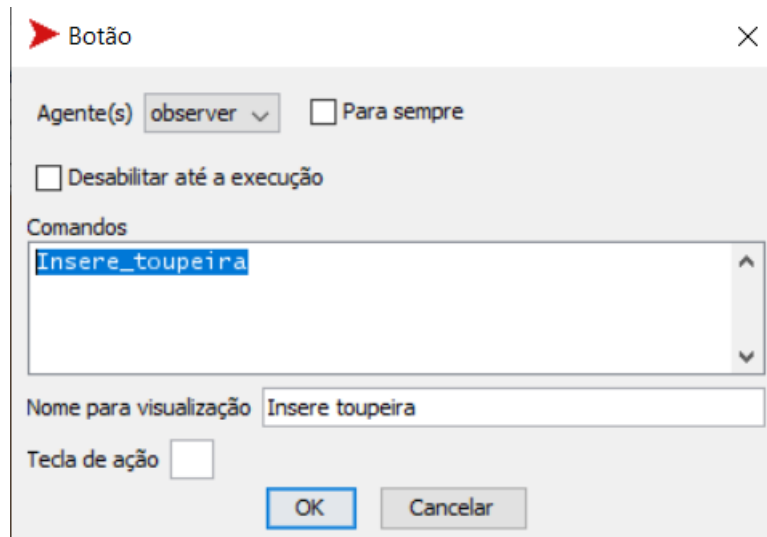


Figura 8.1 – Configurar botão Inserir\_Toupeira

```
to Inserir_toupeira
  create-toupeiras 1 [
    set shape "mole"
    set heading one-of [0 90 180 270]
    set color gray
    setxy random-pxcor random-pycor
    set size 1
    set num_toupeiras num_toupeiras + 1
  ]

  ask toupeiras[
    if shape = "mole" [
      if [pcolor] of patch-ahead 1 = lime [
        ask patch-here[
          set pcolor brown
        ]
      ]
      if [pcolor] of patch-ahead 1 = brown [
        ask patch-here[
          set pcolor brown
        ]
      ]
    ]
  ]
end
```

Figura 8.2 – Código de Plantar “Inserir Toupeira”

7) Insira um deslizador Prob\_Toupeira. A circulação das toupeiras é determinada por um valor de probabilidade. Quanto maior a probabilidade, maior atividade deve ter a toupeira.



Figura 9 – Deslizador de probabilidade

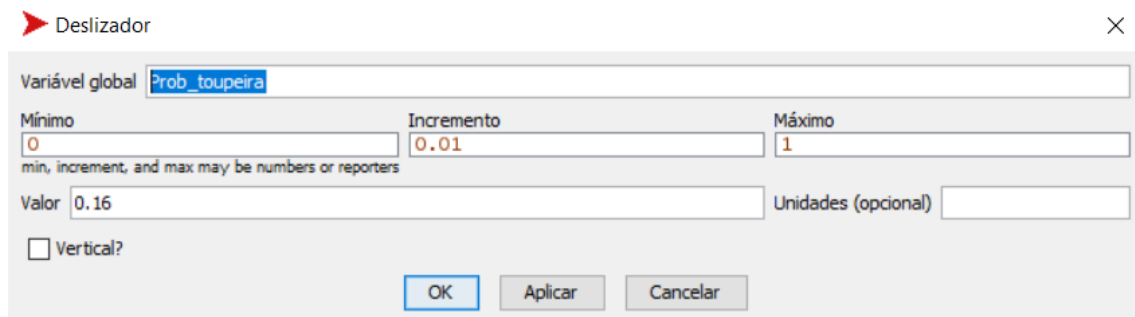


Figura 9.1 – Configuração deslizador de probabilidade

8) Configuração do modo “Manutenção”. Neste modo o planta-relva circula o campo a repor a relva nas células danificadas pelas toupeiras. Este pode mover-se sempre em frente ou numa direção aleatória.

```
if modo_de_plantar = "Manutenção"
[
  ask plantadores [
    if [pcolor] of patch-ahead 1 = brown [
      ask patch-here[
        set pcolor lime
      ]
      forward 1
    ]
    if [pcolor] of patch-ahead 1 = lime [
      ask patch-here[
        set pcolor lime
      ]
      set heading one-of [0 90 180 270]
      forward 1
    ]
  ]
]
```

Figura 10 – Código de Modo de “Manutenção”

9) Se o planta-relva chocar com uma toupeira na mesma célula, a toupeira morre.

```
ask patches[]
if count plantadores-here = 1 [
  if count toupeiras-here = 1 [
    ask toupeiras-here [
      die
      set num_toupeiras num_toupeiras - 1
      set num_mortes num_mortes + 1
    ]
  ]
]
```

Figura 11 – Código de Morte da Toupeira

10) Se duas toupeiras chocarem na mesma célula, dada uma certa idade, nasce uma nova toupeira.

```
ask patches[
  if count toupeiras-here = 2 [
    if age != 0 [
      ask toupeiras-here [
        hatch 1 [
          set age 0
          set heading one-of [0 90 180 270]
          set num_toupeiras num_toupeiras + 1
          set quantidade 0
          set num_nascimentos num_nascimentos + 1
          forward one-of [1 2 3]
        ]
      ]
    ]
  ]
]
```

Figura 12 – Nascimento das toupeiras

## Conclusão

A fim de desenvolver competências fundamentais relativas à modelação e simulação computacional de sistemas com agentes, assim foi realizado o trabalho prático nº1 utilizando a ferramenta NetLogo. Não surgiu qualquer tipo de obstáculo à eficiência e à ética de trabalho do grupo, visto que este debateu e optou sempre pela melhor solução conjunta.

## Referências bibliográficas

Wilensky, U. 1999-2016. NetLogo, [NetLogo Home Page \(northwestern.edu\)](http://northwestern.edu/~netlogo/).