

In [19]:	<pre>%autosave 0 Autosave disabled In [124. #Exercício Prático 2 import random import numpy as np import time def reactive_agent_half_blind(ini_pos, ini_dust, row_size, column_size, elapsed_time): score = 0 #posição do agente inicial: if (ini_pos == 2): #direita ini_row = 0 ini_col = 1 elif (ini_pos == 1): #esquerda ini_row = 0 ini_col = 0 else: #aleatório ini_row = random.randint(0,row_size-1) ini_col = random.randint(0,column_size-1) #células do ambiente inicializadas cells = np.full((row_size, column_size), 0) #sujeira inicial if (ini_dust == 2): #direita cells[0][1] = 1 elif (ini_dust == 1): #esquerda cells[0][0] = 1 elif (ini_dust == 12): #tudo sujo cells[0][1] = 1 cells[0][0] = 1 elif (ini_dust == 0): #totalmente limpo cells[0][1] = 0 cells[0][0] = 0 else: for i in range(2): random_dust = random.randint(0,1) if random_dust == 1: cells[random.randint(0,row_size-1)][random.randint(0,column_size-1)] = 1 if(row_size == 1): for i in range(0,elapsed_time): #sensor reactive_agent_sensor = cells[ini_row][ini_col] #estado inicial #print("Ambiente") #print(cells) #operação de limpeza if(reactive_agent_sensor == 0): ini_col = ini_col #print("posição: ", ini_row, ini_col) #print("Célula limpa.") #print("Procurando nova célula para limpar...") elif(reactive_agent_sensor == 1): #print("posição: ", ini_row, ini_col) #print("Célula suja.") #print("Limpando Célula do Ambiente...") cells[ini_row][ini_col] = 0 #print("Limpeza finalizada!") score = score +1 #print(cells) else: print("ERRO! Célula com dados desconhecidos!") #print(cells) #limites do ambiente if(ini_col+1 == column_size): right = 0 left = 1 elif(ini_col-1 < 0): left = 0 right = 1 #operação de movimento if(left == 0): if(ini_col < column_size-1): ini_col = ini_col+1 score = score-1 elif(right == 0): if(ini_col > 0): ini_col = ini_col-1 score = score-1 #aleatoriedade de sujeira random_dust = random.randint(0,1) if random_dust == 1: cells[random.randint(0,row_size-1)][random.randint(0,column_size-1)] = 1 #print("-----") else: print("ERRO! Ambiente 3D! Projeto conta apenas com ambiente 2D!") return score def reactive_agent_bulls_eye(ini_pos, ini_dust, row_size, column_size, elapsed_time): score = 0 #posição do agente inicial: if (ini_pos == 2): #direita ini_row = 0 ini_col = 1 elif (ini_pos == 1): #esquerda ini_row = 0 ini_col = 0 else: #aleatório ini_row = random.randint(0,row_size-1) ini_col = random.randint(0,column_size-1) #células do ambiente inicializadas cells = np.full((row_size, column_size), 0) #sujeira inicial if (ini_dust == 2): #direita cells[0][1] = 1 elif (ini_dust == 1): #esquerda cells[0][0] = 1 elif (ini_dust == 12): #tudo sujo cells[0][1] = 1 cells[0][0] = 1 elif (ini_dust == 0): #totalmente limpo cells[0][1] = 0 cells[0][0] = 0 else: for i in range(2): random_dust = random.randint(0,1) if random_dust == 1: cells[random.randint(0,row_size-1)][random.randint(0,column_size-1)] = 1 if(row_size == 1): for i in range(0,elapsed_time): #estado inicial #print("Ambiente") #print(cells) #sensor for i in range(1): for j in range(2): if cells[i][j] == 0: #noop ini_col = ini_col elif cells[i][j] == 1: if ini_col != j: ini_col = j score = score-1 else: #noop ini_col = ini_col #limites do ambiente if ini_col > 1: ini_col = 1 elif ini_col < 0: ini_col = 0 #print("posição: ", ini_row, ini_col) #print("Célula suja.") #print("Limpando Célula do Ambiente...") cells[ini_row][ini_col] = 0 #print("Limpeza finalizada!") score = score +1 #aleatoriedade de sujeira random_dust = random.randint(0,1) if random_dust == 1: cells[random.randint(0,row_size-1)][random.randint(0,column_size-1)] = 1 #print("-----") else: print("ERRO! Ambiente 3D! Projeto conta apenas com ambiente 2D!") return score In [125. #AGENTE 1 - Ambiente Parcialmente Observável! #posição inicial/sujeira inicial #1-esquerda/tudo limpo reactive_agent_half_blind(1, 0, 1, 2, 1000) Out[125. -573 In [126. #2-esquerda/tudo sujo reactive_agent_half_blind(1, 12, 1, 2, 1000) Out[126. -567 In [127. #3-esquerda/sujo na direita reactive_agent_half_blind(1, 2, 1, 2, 1000) Out[127. -550 In [128. #4-esquerda sujo na esquerda reactive_agent_half_blind(1, 1, 1, 2, 1000) Out[128. -566 In [129. #5-direita/tudo limpo reactive_agent_half_blind(2, 0, 1, 2, 1000) Out[129. -574 In [130. #6-direita/tudo sujo reactive_agent_half_blind(2, 12, 1, 2, 1000) Out[130. -559 In [131. #7-direita/sujo na direita reactive_agent_half_blind(2, 2, 1, 2, 1000) Out[131. -581 In [132. #8-direita/sujo na esquerda reactive_agent_half_blind(2, 1, 1, 2, 1000) Out[132. -572 In []: Média Global do Half-Blind: -567,75 In [110. #AGENTE 2 - Consegue ver todo o ambiente! #posição inicial/sujeira inicial #1-esquerda/tudo limpo reactive_agent_bulls_eye(1, 0, 1, 2, 1000) Out[110. 274 In [111. #2-esquerda/tudo sujo reactive_agent_bulls_eye(1, 12, 1, 2, 1000) Out[111. 251 In [112. #3-esquerda/sujo na direita reactive_agent_bulls_eye(1, 2, 1, 2, 1000) Out[112. 259 In [113. #4-esquerda sujo na esquerda reactive_agent_bulls_eye(1, 1, 1, 2, 1000) Out[113. 251 In [114. #5-direita/tudo limpo reactive_agent_bulls_eye(2, 0, 1, 2, 1000) Out[114. 244 In [115. #6-direita/tudo sujo reactive_agent_bulls_eye(2, 12, 1, 2, 1000) Out[115. 258 In [116. #7-direita/sujo na direita reactive_agent_bulls_eye(2, 2, 1, 2, 1000) Out[116. 253 In [117. #8-direita/sujo na esquerda reactive_agent_bulls_eye(2, 1, 1, 2, 1000) Out[117. 246 In []: Média Global do Bull's Eye: +254,5</pre>
----------	--