# -\*- coding: utf-8 -\*-

"""

Created on Tue Feb 21 11:05:29 2023

@author: xiaoyu

"""

import random

import matplotlib

from matplotlib import pyplot as plt

import operator

import time

import agentframework as af

# set the random seed

random.seed(0)

# set parameters

n\_agents = 10

# initialise agents

agents = []

for i in range(n\_agents):

# create an agent

agents.append(af.Agent())

print(agents[i])

print(agents)

def get\_distance(x0, y0, x1, y1):

x = x1 - x0

y = y1 - y0

#return 0.5 \*\* ( x\*x + y\*y )

return ( x\*x + y\*y ) \*\* 0.5

print(get\_distance(0,0,3,4)) # test result should be 5

max\_distance = 0

for i in range(len(agents)):

a = agents[i]

for j in range(len(agents)):

b = agents[j]

distance = get\_distance(a.x, a.y, b.x, b.y)

print("distance between", a, b, distance)

max\_distance = max(max\_distance, distance)

print("max\_distance", max\_distance)

# Variables for constraining movement.

# The minimum x coordinate.

x\_min = 0

# The minimum y coordinate.

y\_min = 0

# The maximum x coordinate.

x\_max = 99

# The maximum y coordinate.

y\_max = 99

# Plot

for i in range(n\_agents):

plt.scatter(agents[i].x, agents[i].y, color='black')

# Plot the coordinate with the largest x red

lx = max(agents, key=operator.attrgetter('x'))

plt.scatter(lx.x, lx.y, color='red')

# Plot the coordinate with the smallest x blue

sx = min(agents, key=operator.attrgetter('x'))

plt.scatter(sx.x, sx.y, color='blue')

# Plot the coordinate with the largest y yellow

ly = max(agents, key=operator.attrgetter('y'))

plt.scatter(ly.x, ly.y, color='yellow')

# Plot the coordinate with the smallest y green

sy = min(agents, key=operator.attrgetter('y'))

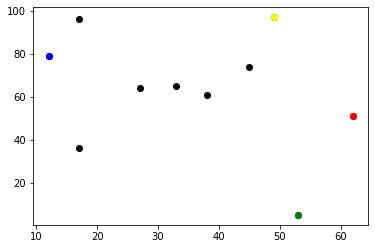
plt.scatter(sy.x, sy.y, color='green')

plt.show()

start = time.perf\_counter()

end = time.perf\_counter()

print("Time taken to calculate maximum distance", end - start, "second")



# -\*- coding: utf-8 -\*-

"""

Created on Tue Feb 21 11:05:29 2023

@author: xiaoyu

"""

import random

import matplotlib

from matplotlib import pyplot as plt

import operator

import time

import agentframework as af

# set the random seed

random.seed(0)

# set parameters

n\_agents = 10

# initialise agents

agents = []

for i in range(n\_agents):

# create an agent

agents.append(af.Agent())

print(agents[i])

print(agents)

def get\_distance(x0, y0, x1, y1):

x = x1 - x0

y = y1 - y0

#return 0.5 \*\* ( x\*x + y\*y )

return ( x\*x + y\*y ) \*\* 0.5

print(get\_distance(0,0,3,4)) # test result should be 5

max\_distance = 0

for i in range(len(agents)):

a = agents[i]

for j in range(len(agents)):

b = agents[j]

distance = get\_distance(a.x, a.y, b.x, b.y)

print("distance between", a, b, distance)

max\_distance = max(max\_distance, distance)

print("max\_distance", max\_distance)

# Variables for constraining movement.

# The minimum x coordinate.

x\_min = 0

# The minimum y coordinate.

y\_min = 0

# The maximum x coordinate.

x\_max = 99

# The maximum y coordinate.

y\_max = 99

# Move agents

n\_iterations = 1000

for n\_iterations in range(n\_iterations):

for i in range(n\_agents):

# Change agents(i) coordinates randomly

agents[i].move(x\_min, y\_min, x\_max, y\_max)

# Plot

for i in range(n\_agents):

plt.scatter(agents[i].x, agents[i].y, color='black')

# Plot the coordinate with the largest x red

lx = max(agents, key=operator.attrgetter('x'))

plt.scatter(lx.x, lx.y, color='red')

# Plot the coordinate with the smallest x blue

sx = min(agents, key=operator.attrgetter('x'))

plt.scatter(sx.x, sx.y, color='blue')

# Plot the coordinate with the largest y yellow

ly = max(agents, key=operator.attrgetter('y'))

plt.scatter(ly.x, ly.y, color='yellow')

# Plot the coordinate with the smallest y green

sy = min(agents, key=operator.attrgetter('y'))

plt.scatter(sy.x, sy.y, color='green')

plt.show()

start = time.perf\_counter()

end = time.perf\_counter()

print("Time taken to calculate maximum distance", end - start, "second")

