Spike: Spike_08
Title: Navigation with Graphs
Author: Muhammad Abubakar

Goals / deliverables:
The goal of this spike was to implement a navigation system for agents using graphs to control the movement of agents in the box world, ensuring that the agents follow a defined path and stop at a target position.

Technologies, Tools, and Resources used:
Python 3.9.7, PyCharm, Pyglet 1.5.27

Tasks undertaken:
The following tasks were undertaken during this spike:
- Analysed the existing codebase for the graph-based navigation system.
- Designed and implemented the Agent class, which includes movement logic and path following.


```python
def update(self, delta):
    if not self.path or not self.path.path:
        return

    target_box = self.world.boxes[self.path.path[0]]
    target_pos = target_box._vc

    target_vector = Vector2D(target_pos.x, target_pos.y)
    pos_vector = Vector2D(self.pos.x, self.pos.y)

    direction = (target_vector - pos_vector).normalise()
    distance = self.speed * delta
    travel = direction * distance

    if (pos_vector + travel - target_vector).length() < distance:
        self.pos = target_pos
        self.path.path.pop(0)
        if self.path.path:
            self.box_idx = self.path.path[0]
        else:
            self.box_idx = None
    else:
        self.pos = Vector2D(pos_vector.x + travel.x, pos_vector.y + travel.y)
```

```python
class Agent:
    ⚲ Muhammad Abubakar
    def __init__(self, world, box_idx, speed):
        self.world = world
        self.box_idx = box_idx
        self.pos = world.boxes[box_idx]._vc
        self.speed = speed
        self.path = None

    1 usage (1 dynamic)   ⚲ Muhammad Abubakar
    def set_path(self, path):
        if self.path is None:
            self.path = path
```

The Agent class represents a moving object in the box world that travels along a predefined path through a series of box indices. The agent travels at a predetermined rate and updates its position based on the time difference. If the agent does not already have a path, the set_path function establishes one, while the update method computes the target location, direction, distance, and travel vectors and changes the agent's position accordingly. If the agent achieves its purpose, it proceeds to the next accessible box on the route. The draw method is in charge of showing the agent graphically as a circle in the world.

- Integrated the Agent class into the existing box world class, allowing for multiple agents to navigate the environment.

```python
def set_start(self, idx):
    '''Set the start box based on its index idx value. '''
    # remove any existing start node, set new start node
    if self.target == self.boxes[idx]:
        print("Can't have the same start and end boxes!")
        return
    if self.start:
        self.start.marker = None
    self.start = self.boxes[idx]
    self.start.marker = 'S'
    self.agents.append(Agent(self, idx, random.randrange(50, 200)))
```

```python
def update(self, delta):
    for agent in self.agents:
        agent.update(delta)
```

self.agents.append(Agent(self, idx, random.randrange(50, 200))) creates a new instance of the Agent class and appends it to the list self.agents. The speed of the agent is chosen randomly between 50 and 200 using random.randrange(50, 200). This line is executed when setting a new start box for the agent's path, ensuring that a new agent is created and added to the list of agents in the current world.

What we found out:
The implementation of the Agent class was successful, allowing agents to navigate through the environment following the defined path. However, some issues were encountered, such as the agents not stopping precisely at the target position and occasional crashes due to the path update logic.

Code:
The code for this spike can be found in the *16 – Spike – Navigation with Graphs* directory.

```
commit bb537d0ddb112e6b07f31e67d0dc4f7fcc77c8d0 (origin/master, origin/HEAD)
Author: Muhammad Abubakar <Muhammad.Abubakar@itconnexion.com>
Date:   Wed May 10 12:28:54 2023 +1000

    custom plan

commit 0e1cf694f918b57d4d5c04a5f386f884a3f9b0ba
Author: Muhammad Abubakar <Muhammad.Abubakar@itconnexion.com>
Date:   Wed May 10 12:28:37 2023 +1000

    Implemented the multiple agents for path following

commit 03c91fe6a1f0cf395c834dbc8fcea1adfb0c5ae7
Author: Muhammad Abubakar <Muhammad.Abubakar@itconnexion.com>
Date:   Tue May 9 14:47:04 2023 +1000

    spike extension report
```

Open issues/risks:

One potential risk is the system's robustness when dealing with a larger number of agents and more complex environments. Additionally, the current implementation may not handle dynamic changes in the environment, such as obstacles appearing or disappearing during runtime.

Recommendations:

Based on the results of the spike, it is recommended to:

- Based on the results of the spike, it is recommended to:
- Further refine the Agent class's movement and path update logic to ensure a smooth and accurate navigation experience.
- Consider adding checks and conditions to handle situations where the environment changes during runtime, such as implementing a dynamic pathfinding algorithm.
- Test the system in more complex scenarios with a larger number of agents and obstacles to ensure scalability and performance.
- Explore alternative navigation techniques, such as potential fields or reinforcement learning, to compare their effectiveness and suitability for the given problem.