CSC148 Worksheet 6 Solution

Hyungmo Gu

April 19, 2020

Question 1

- The two classes already defined are: Vehicle and SuperDuperManager

  The additional classes required to create for this exercise are: Car, Helicopter and MagicCarpet

  Car, Helicopter and MagicCarpet are child classes of Vehicle.

Question 2

a. The following are attributes possessed by all vehicles

- type
- initial_position
- moves_to
- move_diagonally
- fuel_usage

Correct Solution:

The following are attributes possessed by all vehicles

- position
- fuel

b. No. Referencing the following code in *worksheet_6_starter_code.py*,

```python
class Vehicle:
    ...

    def __init__(self, initial_fuel: int,
                 initial_position: Tuple[int, int]) -> None:
        ...

        self.fuel = initial_fuel
        self.position = initial_position

```

we can come up with the following examples.

- Vehicle(100, (10,20))
- Vehicle(50, (5, 10))

Here, we can see the two vehicles have different value of fuel and initial position.

c. *fuel_needed* not implemented because each child classes have different fuel consumption rate, and the method is to be defined by the child classes by overriding it.

d. The following methods must be defined in each of its subclasses

- Car
  - fuel_needed
  - move
- Helicopter
  - fuel_needed
  - move
- MagicCarpet
  - __init__
  - move

Correct Solution:

- Car
  - __init__
    - Necessary because the parameter *position* must be set as optional
    - Necessary because *self.position* must default to (0,0) if the argument of position not given.
  - fuel_needed
    - Necessary because vehicle uses fuel

    - Necessary because needs to define the fuel cost based on it not being able to moving diagonally.

- Helicopter
  - __init__
    - Necessary because the parameter *position* must be set as optional
    - Necessary because *self.position* must default to (3,5) if the argument of position not given.
  - fuel_needed
    - Necessary because vehicle uses fuel
    - Necessary because needs to define the fuel cost based on it being able to move diagonally.

- MagicCarpet
  - __init__
    - Necessary to set the parameters *initial_fuel*, *initial_position* as optional
    - Necessary to randomize the value of *self.position.*
  - move
    - Necessary to set the parameters *new_x* and *new_y* as optional.
    - Necessary to randomize the value of new position.

Question 3

```
    """
    Initializing SuperDuperManager:
    >>> s = SuperDuperManager()
    >>> s._vehicles
    {}

    Adding Vehicles:
    >>> s.add_vehicle('Car', '1', 100)
    >>> s._vehicles['1'].__class__.__name__
    'Car'
    >>> s.add_vehicle('Helicopter', '1', 100)
    >>> s._vehicles['1'].__class__.__name__
    'Car'

    >>> s.add_vehicle('Helicopter', '2', 100)
    >>> s._vehicles['2'].__class__.__name__
    'Helicopter'

    >>> s.add_vehicle('UnreliableMagicCarpet','3',100)
    >>> s._vehicles['3'].__class__.__name__
    'UnreliableMagicCarpet'

    Moving Vehicle:
    >>> s._vehicles[’1’].position
    (0,0)
    >>> s.move_vehicle(’1’, 1, 1)
    >>> s._vehicles[’1’].position
    (1,1)

    >>> s._vehicles[’2’].position
    (3,5)
    >>> s.move_vehicle(’2’, 1, 1)
    >>> s._vehicles[’2’].position
    (4,6)

    >>> s._vehicles[’3’].position
    (4,8)
    >>> s._vehicles[’3’].position
    (12,4)
    >>> s.move_vehicle(’3’, 1, 1)
    >>> s._vehicles[’3’].position
    (100,100)

    Get Vehicle Position:
    >>> s.get_vehicle_position(’1’)
    (1,1)

    >>> s.get_vehicle_position(’2’)
    (4,6)

    >>> s.get_vehicle_position(’3’)
    (50,200)

    Get Vehicle Fuel:
    >>> s.get_vehicle_fuel(’1’)
    98

    >>> s.get_vehicle_fuel(’2’)
    99

    >>> s.get_vehicle_fuel(’2’)
    100
    """
```

Question 4

Question 5

Question 6

Question 7

Question 8