

CSC148 Worksheet 6 Solution

Hyungmo Gu

April 18, 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*,

```

1  class Vehicle:
2      ...
3
4      def __init__(self, initial_fuel: int,
5                    initial_position: Tuple[int, int]) -> None:
6          ...
7
8          self.fuel = initial_fuel
9          self.position = initial_position
10

```

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

```

1      """
2      Initializing SuperDuperManager:
3      >>> s = SuperDuperManager()
4      >>> s._vehicles
5      {}
6
7      Adding Vehicles:
8      >>> s.add_vehicle('Car', '1', 100)
9      >>> s._vehicles['1'].__class__.__name__
10     'Car'
11     >>> s.add_vehicle('Helicopter'
12     >>> s._vehicles['1'].__class__.__name__
13     'Car'
14     >>> s.add_vehicle('Helicopter', '2', 100)
15     >>> s._vehicles['2'].__class__.__name__
16     'Helicopter'
17     >>> s.add_vehicle('UnreliableMagicCarpet', '3', 100)
18     >>> s._vehicles['3'].__class__.__name__
19     'UnreliableMagicCarpet'
20
21     Moving Vehicle:

```

```
22 >>> s.move_vehicle('1', 1, 1)
23 ""
```

Question 4

Question 5

Question 6

Question 7

Question 8