

School Nova Computer Science 202

Homework 6 (due 11/7/2020)

Save your code as lastname_homework6.py and submit on Google Classroom

Task 1

Modify your previous homework solution using the class work code (that is, make sure that each PLANE's airport attribute is a reference to an actual AIRPORT object).

If your previous homework is incomplete, feel free to start with the class work code.

Task 2

For the AIRPORT class, implement an instance method that identifies the closest airport and returns the closest airport **object** (the actual object, not the id of the closest airport!).

Task 3

Modify the display methods for both PLANE and AIRPORT to display the id **and** location of the closest airport.

Task 4

Create a PLANE method that moves the plane to the closest airport. (Make sure to update the list of planes for both airports).

Task 5

Add an airport capacity attribute to each airport (the maximum number of planes that the airport can accommodate). Add airport capacity status attribute to each airport ("closed" when the maximum capacity is reached; "open" when the number of planes on the ground is below the capacity).

Task 6

Add a method that updates the airport capacity status based on the number of airplanes and airport capacity.

Task 7

Modify the display methods from task 3 to display the capacity and capacity status of the current and closest airport.

Task 8

Modify the PLANE method that moves a plane to the closest airport such that the plane cannot move to the "closed" airport even if it is geographically closest. In other words, the plane will move to the closest "open" airport only. Test your code generating the relevant number of airports, planes, and airport capacity values.

Task 9

What does the following code do?

```
planes.sort(key=lambda x: x.id, reverse=False)
```

Do the same for the airports list.

Task 10

Add passenger plane and cargo plane subclasses. Add a unique attribute to each of the subclass (use your imagination). How/where would you store your passenger planes and cargo planes objects? (you don't need to implement any solutions yet; just think about it).