Dhirubhai Ambani Institute of Information and Communication Technology

Gandhinagar, Gujarat

Name : Ghori Zeel Jivrajbhai

Student ID : 202201287

Course : IT 314 Software Engineering

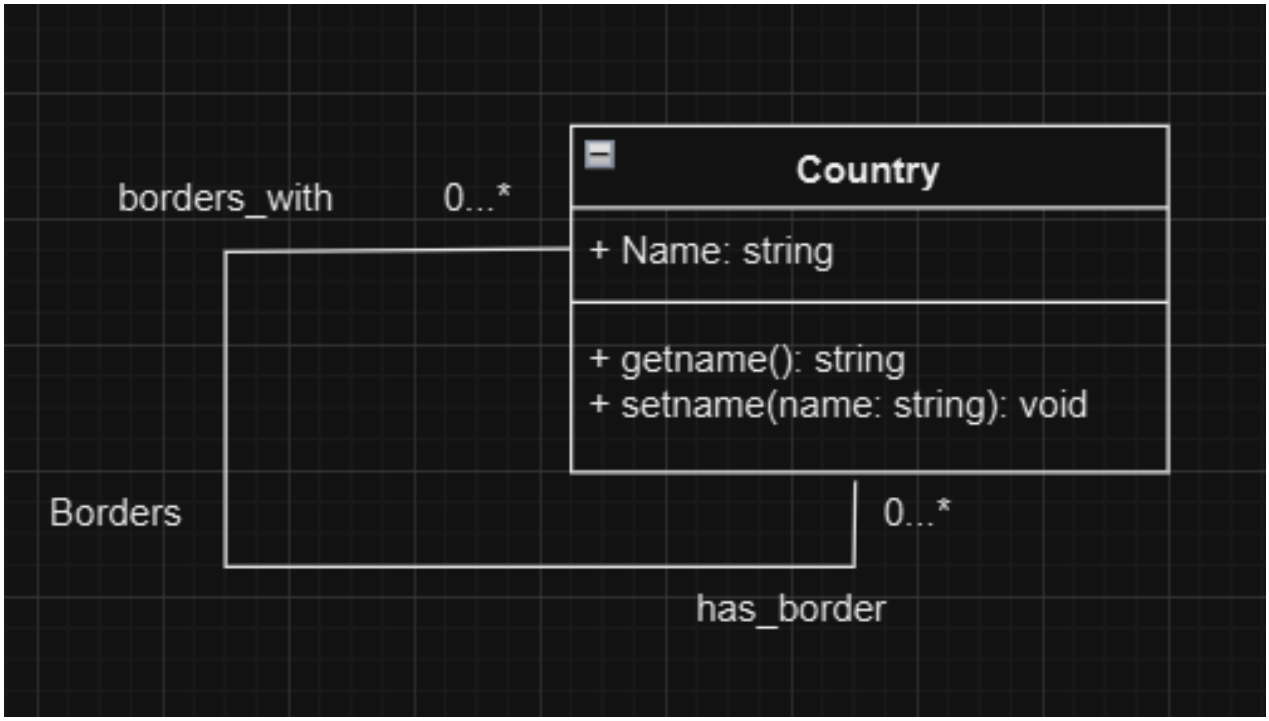
Professor : Saurabh Tiwari

Semester : Autumn 2024

Lab 4 : Class Diagram

Q.1 Prepare a class diagram for the following object diagram that shows a portion of Europe.

Figure-1



Q.2 Prepare a class diagram for object diagram given in Figure -2. Explain your multiplicity decisions. What is the smallest number of points required to construct a polygon? Does it make a difference whether or not point may be shared between polygons? Your answer should address the fact that points are ordered.

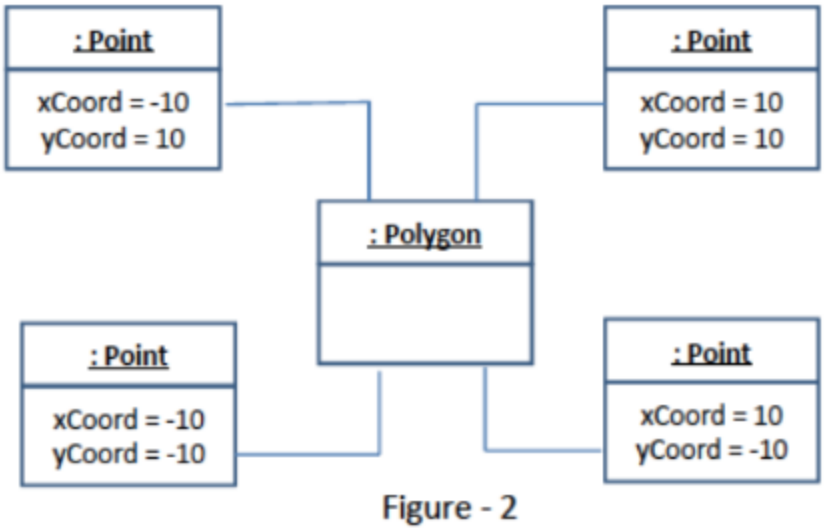
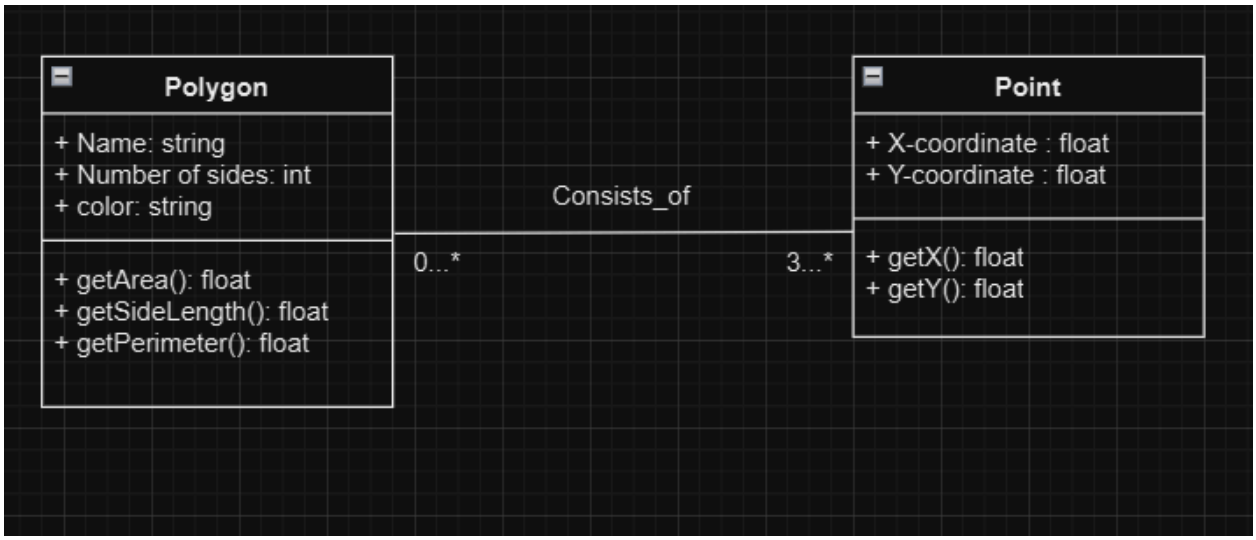


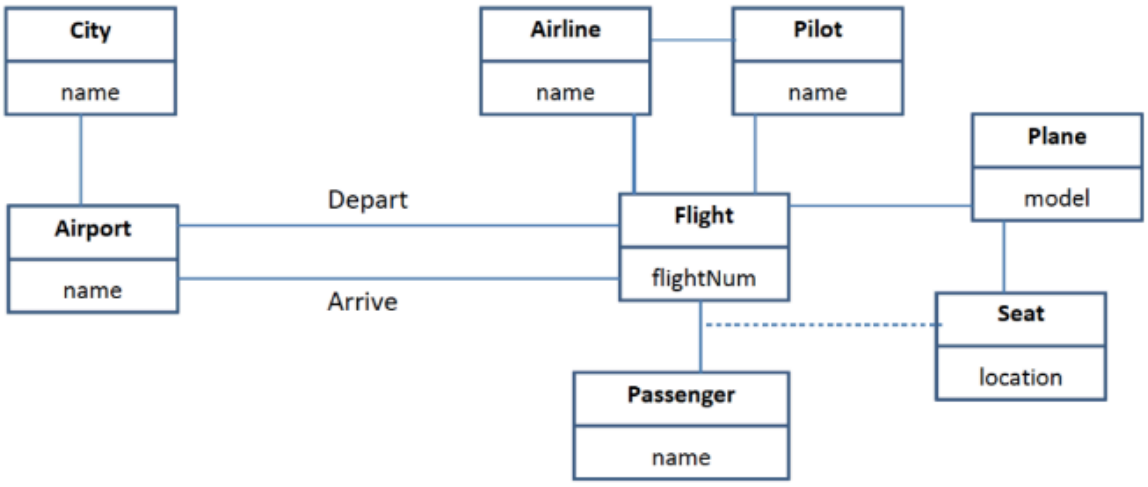
Figure - 2

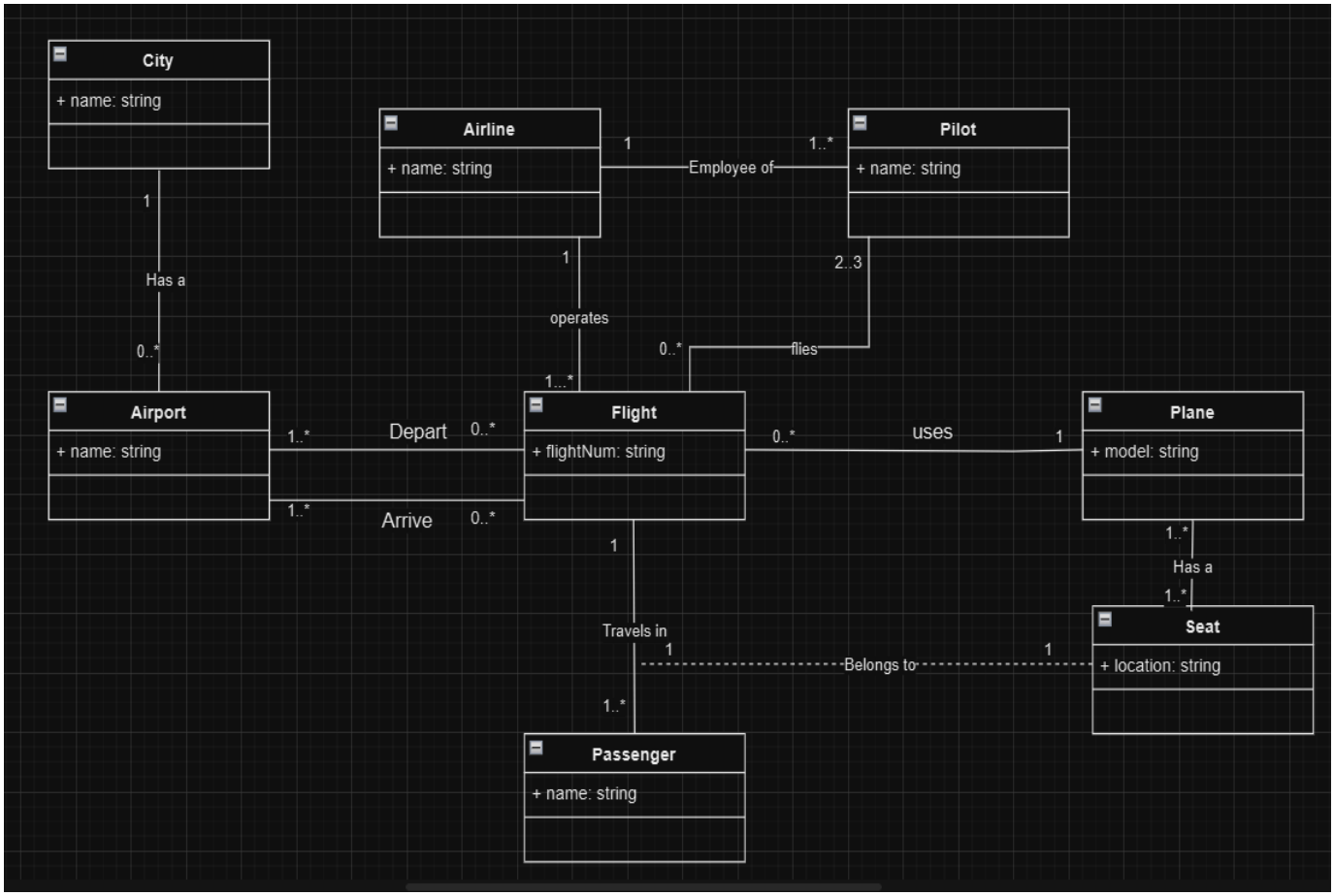


- A polygon can have a minimum of three points and maximum infinite points. Hence multiplicity is 3…* .
- Similarly a point can be or can not be a part of polygon and can be part of multiple polygons(shared). Hence its multiplicity is 0…*.

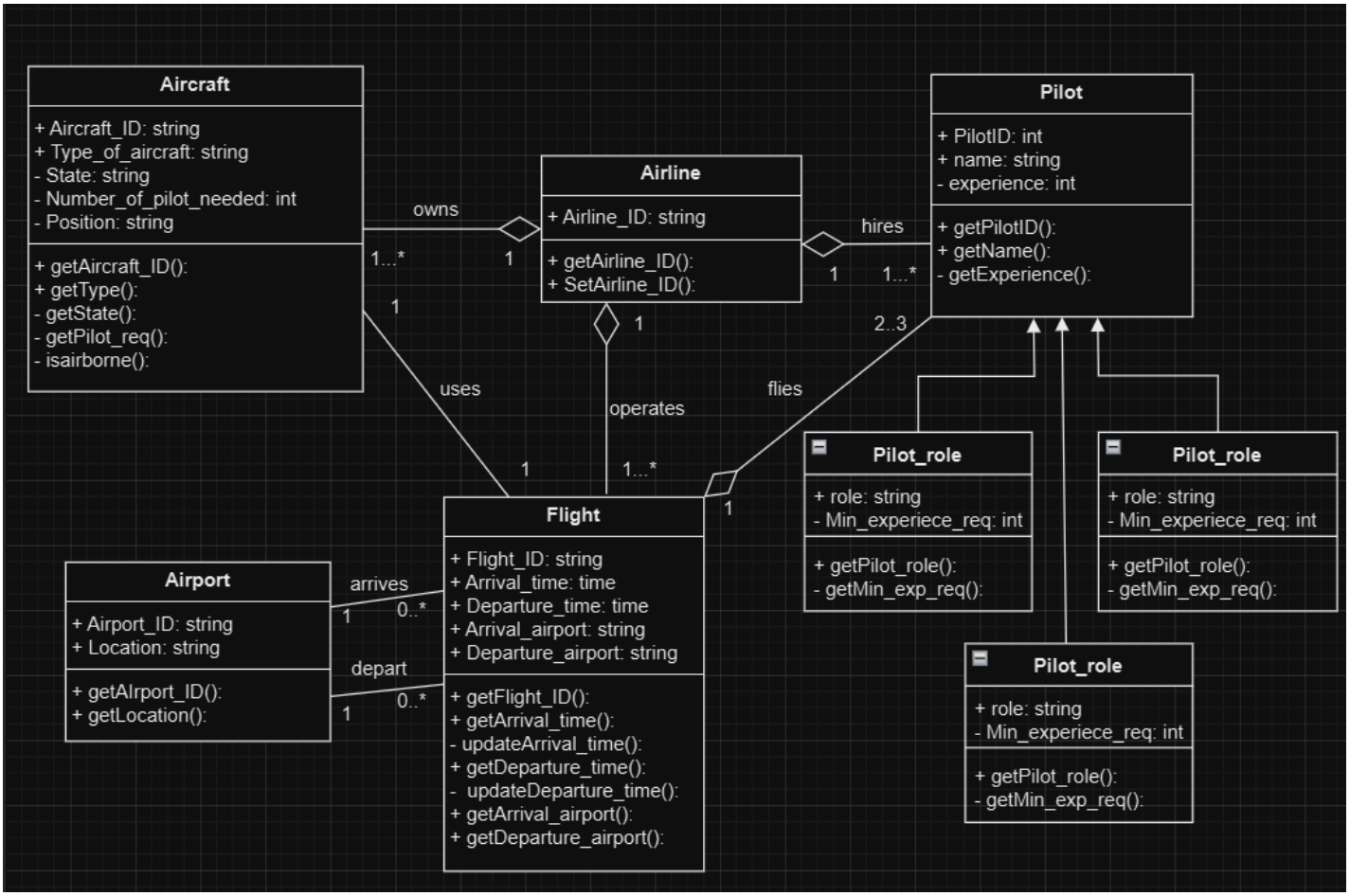
- Yes, it makes a difference that points could be shared between polygons, as it changes multiplicity. If points cannot be shared between polygons, then each polygon must have its own distinct set of points, meaning no common edges or vertices.
  0…1 → if not shared
  0…* → if shared
- The fact that ports are ordered matters for defining the shape of a polygon.

Q.3 Figure 3 is a partially completed class diagram of an air transportation system. Add multiplicities in the diagram. Also add association names to unlevelled associations.





Q.4 We want to model a system for management of flights and pilots. An airline operates flights. Each airline has an ID. Each flight has an ID a departure airport and an arrival airport: an airport as a unique identifier. Each flight has a pilot and a co-pilot, and it uses an aircraft of a certain type; a flight has also a departure time and an arrival time. An airline owns a set of aircrafts of different types. An aircraft can be in a working state or it can be under repair. In a particular moment an aircraft can be landed or airborne. A company has a set of pilots: each pilot has an experience level: 1 is minimum, 3 is maximum. A type of aeroplane may need a particular number of pilots, with a different role (e.g.: captain, co-pilot, navigator): there must be at least one captain and one co-pilot, and a captain must have a level 3.


Aircraft
+ Aircraft_ID: string
+ Type_of_aircraft: string
- State: string
- Number_of_pilot_needed: int
- Position: string
+ getAircraft_ID():
+ getType():
- getState():
- getPilot_req():
- isairborne():
owns
1...*
1
Airline
+ Airline_ID: string
+ getAirline_ID():
+ SetAirline_ID():
hires
1
1...*
Pilot
+ PilotID: int
+ name: string
- experience: int
+ getPilotID():
+ getName():
- getExperience():
1
uses
1
operates
1...*
2..3
flies
1
Flight
+ Flight_ID: string
+ Arrival_time: time
+ Departure_time: time
+ Arrival_airport: string
+ Departure_airport: string
+ getFlight_ID():
+ getArrival_time():
- updateArrival_time():
+ getDeparture_time():
- updateDeparture_time():
+ getArrival_airport():
+ getDeparture_airport():
Airport
+ Airport_ID: string
+ Location: string
+ getAIrport_ID():
+ getLocation():
arrives
1
0..*
depart
1
0..*
Pilot_role
+ role: string
- Min_experiece_req: int
+ getPilot_role():
- getMin_exp_req():
Pilot_role
+ role: string
- Min_experiece_req: int
+ getPilot_role():
- getMin_exp_req():
Pilot_role
+ role: string
- Min_experiece_req: int
+ getPilot_role():
- getMin_exp_req():