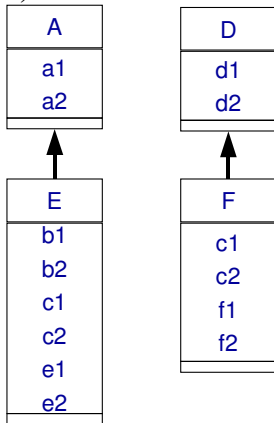


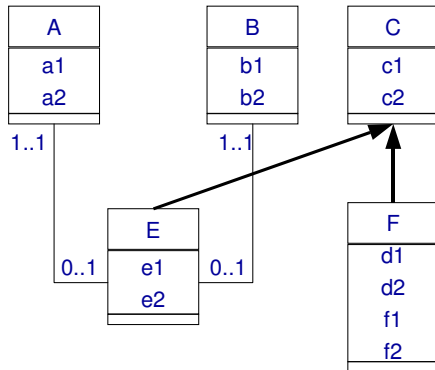
ΑΠΑΝΤΗΣΕΙΣ

Άσκηση 1

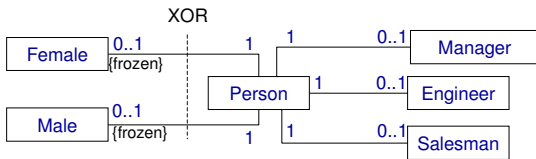
a)



b)



Άσκηση 2



Άσκηση 3

[A] -- x----- rel ----- y - [B]

y:0..n no constraint

y:1..n // mandatory
context A inv: not self.rel->isEmpty()

ή
context A inv: self.rel->Size () >= 1

y:0.1 // single valued
context A inv: self.rel->Size () <= 1

y:1..1
context A inv: self.rel->Size () = 1

Άσκηση 4

a) context Person

inv: (self.wife->notEmpty() implies not (self.wife = self)) and
(self.husband->notEmpty() implies not (self.husband = self))

b) context Person

inv: not (self.wife->isEmpty()) implies self.wife.sex="female"

c) context Person

inv: not (self.husband->isEmpty()) implies self.husband.sex="male"

d) context Person

inv: (self.wife->notEmpty() or self.husband->notEmpty()) implies self.age > 18

e) context Person::marryWife(p:Person)

pre : (self.sex="male" or self.sex=nil) and (p.sex="female" or p.sex=nil) and
self.wife->isEmpty() and p.husband->isEmpty()

post: self.wife=p and p.husband=self and self.sex="male" and p.sex="female"

context Person::marryHusb(p:Person)

pre : (self.sex="female" or self.sex=nil) and (p.sex="male" or p.sex=nil) and
p.wife->isEmpty() and self.husband->isEmpty()

post: p.wife=self and self.husband=p and self.sex="female" and p.sex="male"

context Person::divorce()

pre : self.wife->notEmpty() or self.husband->notEmpty()

post : (self.sex="male" implies (self.wife->isEmpty() and
self.wife@pre.husband->isEmpty())) and

(self.sex="female" implies (self.husband->isEmpty() and
self.husband@pre.wife->isEmpty()))

Άσκηση 5

context Order

inv: self.oline1 ->forall(orderLine1, orderLine2 : OrderLine |
(orderLine1 <> orderLine2) implies (orderLine1.olproduct.id <>
orderLine2.olproduct.id))

Άσκηση 6

1. context Flight

inv: self.passenger->size() <= self.airplane.passengerCapacity

2. context Flight

inv: crew->forall(c : Person | c.employee = self.airline)

3context Flight

inv: self.crew->size() = self.airplane.numCrewRequired

4.context FlightSegment

inv: not arrive.isBefore(depart)

5.context Flight

inv: Sequence{ 1..schedule->size()-1 }->forall(i |
schedule->at(i+1).depart.differenceMinutes(schedule->at(i).arrive) >= 30)

Επειδή οι φοιτητές δεν είχαν διδαχθεί την λειτουργία “at” , θεώρησα σωστές και λύσεις της μορφής:

context FlightSegment inv: depart.differenceMinutes(arrive) >=30

6.context Flight::assignCrew(p:Person):Boolean

pre: p.employee = self.airline

pre: p.qualifiedFor->includes(self.airplane)

post: result = self.crew->includes(p)

Άσκηση 7

1. context ViewPort :: resize (newHeight:Integer): void

pre: 0 <= newHeight and newHeight <= self.maxSize

post: self.height = newHeight

2.context ScrollBar :: moveHandle (p:Integer) : int

pre: 0 <= p and p <= (view.file.size() - 1)

post: self.handlePosition = p

3.context ViewPort:

inv: self.contents->forAll(i: int | (0 <= i < self.height) and (i < file->size() - scroll.handlePosition))

implies self.contents[i] = file->getLine(scroll.handlePosition + i))

4.context ViewPort

inv: self.contents[0] = file.getLine(scroll.handlePosition)

5.context ScrollBar

inv: if view.file->size() <= view.height then self.handleSize = 1

else self.handleSize = view.height / view.file.size()

endif