

Clinical Lab System: Statement

The task is to develop a computerized data management system for a clinical test analyzer. An analyzer can carry out tests on body fluids such as blood, urine, and swab specimens. An analyzer is capable of carrying out tests on several samples simultaneously.

The technician enters a batch of samples from a single patient by first entering the patient's identification and then indicating, one at a time, the tests that need to be performed on the samples. By a "batch end" message, s/he informs the system that there are no more samples for the current patient. When all the tests for a patient have been performed by the analyzer, they are collected together into a patient report, which is sent to the technician.

The system can process analyses for several patients simultaneously, but all test samples for a particular patient must be entered before any tests can begin. The technician may ask for a report reflecting the current status of a patient's tests before they are all completed. The tests for a patient may also be aborted, in which case a patient report containing just the test results collected so far is generated and all further tests on samples from the same patient are ignored.



Clinical Lab System

Environment Model

Show by a Environment Model the interaction between the technician, the system and the analyzer.

Provide message declarations.

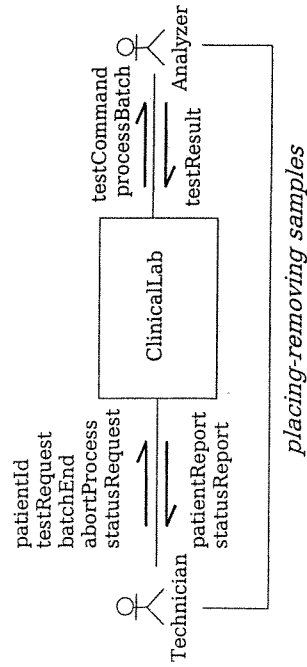
Write down some possible/forbidden message sequences; show both input and output messages. (Could also be answered based on the Protocol Model.)

Protocol Model (Full Exercise)

Devise a Protocol Model. Discuss. Especially explain what is in the problem statement but not in the Protocol Model.



Clinical Lab System



Clinical Lab System: Message Types

type Test is enum {... names of lab tests on blood, urine, and swab.};
 type State is enum {finished, ongoing};
 type Measure is record (kind: Test, value: Real);

patientId (id: Integer, name: String);
 testRequest (id: Integer, kind: Test);
 batchEnd (id: Integer);
 abortProcess (id: Integer);
 statusRequest (id: Integer);
 testCommand (id: Integer, kind: Test);
 testResult (id: Integer, result: Measure);
 patientReport (id: Integer, results: Collection (Measure));
 statusReport (id: Integer, status: State);
 processBatch (id: Integer);



Clinical Lab System

Possible:

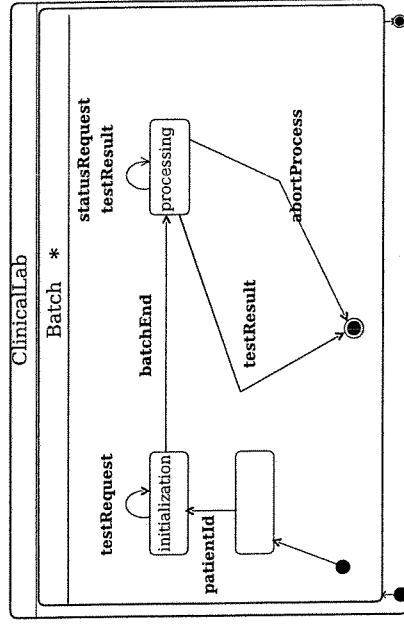
patient id John, test request 1 for John, patient id Bob, test request 2 for John, test request 1 for Bob, test request 3 for John, batch end for John, test result 1 for John, status report for John, status report for John, batch end for Bob, test result 1 for Bob, patient report for Bob, test result 3 for John, test result 2 for John, patient report for John.

Note that any type of interleaving between patient analyses is possible. Impossible:

... test request 1 for John, test result 1 for John, batch end for John,...



Clinical Lab System



Clinical Lab System: Answers

1. Based only on the Protocol Model, do we know if a status request is about a patient or a test?

No. The message parameters should tell us; according to the statement a status report is about a patient, and the patient must therefore be a parameter of a status request.

2. Is it possible to issue a status request for some patient between two test requests for another one?

Yes, since there are as many independent concurrent Batch submachines as needed.

3. Is it possible to issue a status request while another one is still pending?

No, the status report is delivered immediately after the status request. Reminder: a system operation is immediate.



Clinical Lab System: Answers

4. Is it possible to enter the ID of a new patient, even though the test requests of the previous one are not yet completed.

Yes, patient batches are independent of one another.

5. Is it possible to start the batch of a new patient, even though not all test results of the previous one are available.

Yes, again all activities of patient batch processing are independent of one another.

6. Does the Protocol Model guarantee that the test results of a patient arrive in the same order the test requests were issued?

No.



Clinical Lab System: Answers

7. Does the Protocol Model guarantee that the test results of two different patients are not interleaved?
No, and the analyzer will indeed produce the results in disorder, some tests taking more time than others.
8. Does the Protocol Model show who is the generator of a message.
Give examples illustrating your answer.
No, testResult come from the analyzer, whereas batchEnd comes from the technician.
9. Does the Protocol Model show when the system terminates.
No.



DVM: Statement

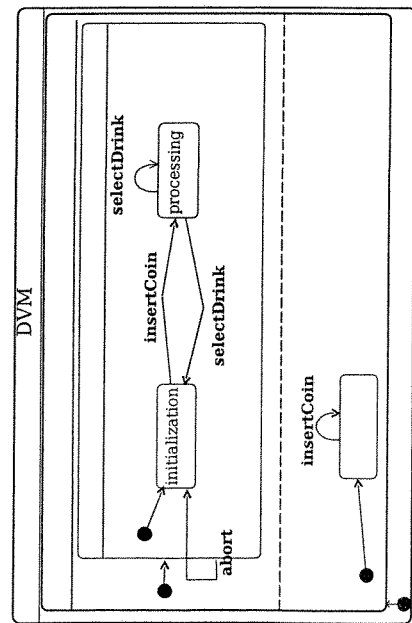
The task is to develop a model of a Drink Vending Machine. The user has to be able to insert money and to select a drink. If the amount of money inserted by the user is greater than the price of the selected drink the system must deliver the selected drink and return the change to the user.

Furthermore, it is always possible for the user to insert additional coins into the machine or to abort the operation of buying a drink. In the latter case, the amount of money inserted into the machine since the last drink has been delivered will be returned to the user.

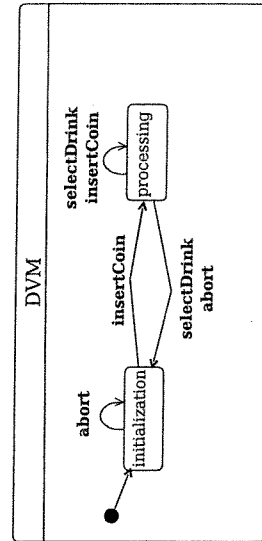
For this problem statement you are asked to provide a Protocol Model



DVM, variant 1



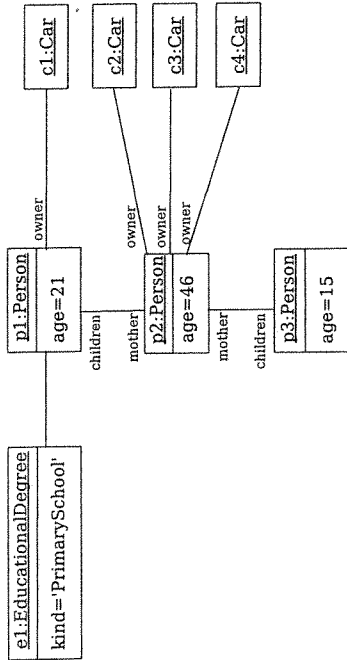
DVM, variant 2



OCL – Person/Car/Educational Degree

- Draw an object diagram which satisfies all multiplicity constraints

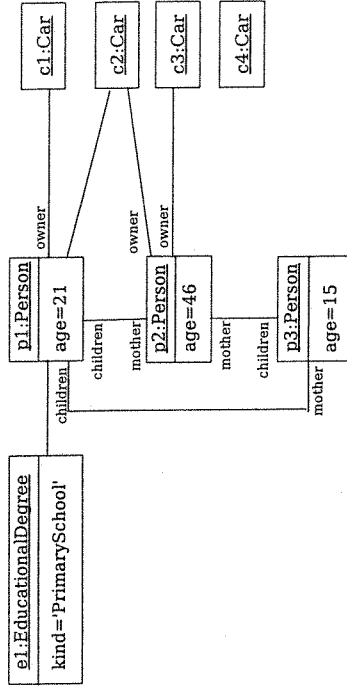
Object Diagram 1



OCL – Person/Car/Educational Degree

- Draw an object diagram with at least one broken multiplicity constraint

Object Diagram 1



OCL – Person/Car/Educational Degree

- Give an informal description of both invariants
 - Inv 1:
 - There is no person with negative age and
 - Nobody can own more than 4 cars and
 - If someone has a mother then she must be older than him/her and
 - If someone is older than 16 then he/she must have some educational degrees and the first one of them must be primary school
 - Inv 2:
 - Cars can not be owned by minors (younger than 18)

OCL – Person/Car/Educational Degree

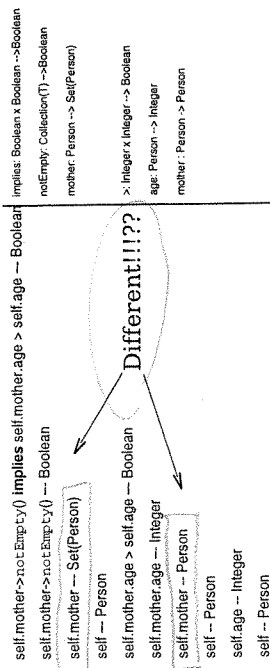
- Apply the evaluation algorithm on both given invariants in both object diagrams
- We evaluate the first invariant in object diagram 1 for object p1

Expression	Evaluation
self.age	21
21 > 0	TRUE line 1
self.car	{c1}
{c1} > size()	1 line 2
1 < 5	TRUE
self.mother	{p2}
{p2} > notEmpty()	TRUE
self.mother	p2
p2.age	46
46 > 21	TRUE line 3
TRUE implies TRUE	TRUE line 4
21 > 16	TRUE
self.educationalDegree	{e1}
{e1} > notEmpty()	TRUE line 5
{e1} > first()	e1
e1.kind	PrimarySchool
PrimarySchool = 'PrimarySchool'	TRUE line 6
TRUE and TRUE	TRUE line 5 and line 6
TRUE implies TRUE	TRUE line 4 and (line 5 and line 6)
TRUE and TRUE	TRUE line 1 and line 2

Case 2 [self.p2]

OCL - Person/Car/Educational Degree

For the line `self.mother` list all subexpressions with their type



OCL - Person/Car/Educational Degree

- How is the line `self.mother` evaluated for objects not having a link for association with role mother?

`self.mother --> notEmpty()` --> false
 (false implies ...) --> true

This means that this line will always evaluate to true for objects that don't have a link mother

OCL - Person/Car/Educational Degree

- Give an invariant equivalent to the second invariant which starts with `context Person inv:...`

`context Person inv:`
`self.car --> notEmpty implies self.age >= 18`

OCL - Person/Car/Educational Degree

- Formalize that mothers have always more cars than their children

`context Person inv:`

`self.mother --> notEmpty() implies self.mother.car --> size() > self.car --> size()`

...or...

`context Person inv:`

`self.children --> forAll(c: Person | c.car --> size() < self.car --> size())`

