Specification of Information Systems (233030) Examination

28th January, 2009

Explain your answers, yet keep your explanations precise. Long-winded answers are not appreciated.

1. a. What is the difference between a subject domain and the context of the system as shown in a context diagram? Use an example to explain your answer. `context is where messages at system interface come from or go to; s.d. is subject (or topic) of these messages.`

   b. The subject domain of a reactive system changes when we add functions to, or delete functions from the system. Explain this and give an example of each. `page 21 question 7.`

   c. Can the subject domain of a company be the same as that of an information system owned by the company? Explain by means of examples. `page 21 question 9.`

2. Figure 1 shows a point-of-sale-terminal (POST) connected to a radio, that connects wireless to another radio that is attached to a scanner, which is used to scan items.

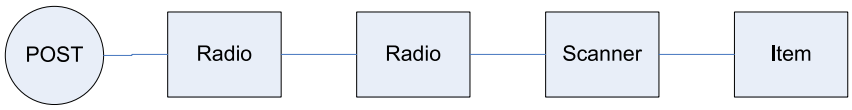

Figure 1: A context diagram.

   (a) Give two reasons for dropping the two radio entities from the diagram. `(1) no errors made, (2) no significant delay, (3) message always arrives at destination. See also page 216 figure 17.2 and slide remarks about chapters 15, 18, 19.`

   (b) If an item is scanned, is this an action or an event? Why? `It is an action by the scanner (by the sales clerk really) but an event for the system. It is an event for the system because it does not cause the event but must respond to it`

   (c) Scanning takes time. Should scanning be modelled as a process, a state, or an event? Why? `It is an event for the POST. For the scanner it is a process but for the POST it is an atomic event with no significant intermediate state and with a single time stamp.`

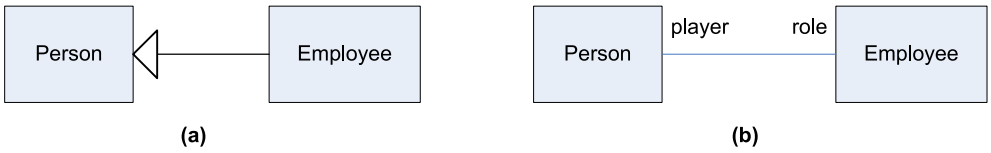

Figure 2: Two entity-relationship diagrams.

3. Figure 2 shows two ways to model the relationship between a person and an employee.

   (a) Each model has an implication for how we count entities. Explain this difference between the two models. `In (a) each employee is identical to an person; in (b) there is a many-many relationship.`

   (b) Add a cardinality property to diagram (b) to reduce this difference in meaning. After you have added this property to diagram (b), do the two diagrams still have the same implications for the way we count the entities? Explain your answer. `After you add the cardinality property 1 for player (each employee is a role played by exactly 1 person) you still have the possibility that in two different states of the world, the employee role is played by different persons. That is not possible in (a).`

   (c) The two diagrams also express existence dependencies. Which existence dependencies are expressed? `An employee cannot exist without being a person (a) or without being played by a person (b).`

4.) Figure 3 shows a context diagram in which a button X can send a push-X event to a controller, and receive X-on and X-off actions from the controller; the effect of X-on is that the button lights up and the effect of X-off is that the button light switches off. Button Y has a similar interface.

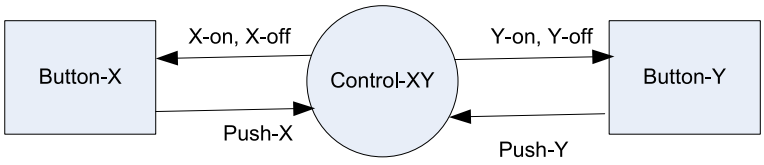

Figure 3: A context diagram.

   (a) Make a statechart for the behavior of the controller using two parallel substates.

   (b) Based on the statechart of question (a), make a statechart that represents another another controller, that allows X to light on only when Y is off; if Y is on, pushing X must have no effect. `Use in(State).`

   (c) Represent the system of (b) by another statechart, that has no parallel substates but has only one state and uses local variables X and Y. `See page 146 exercise 12.`

5. Figure 4 shows a decomposition of an elevator controller. For ease of representation, the external entity Elevator doors is represented twice, as is the component *Arrival sensing*.

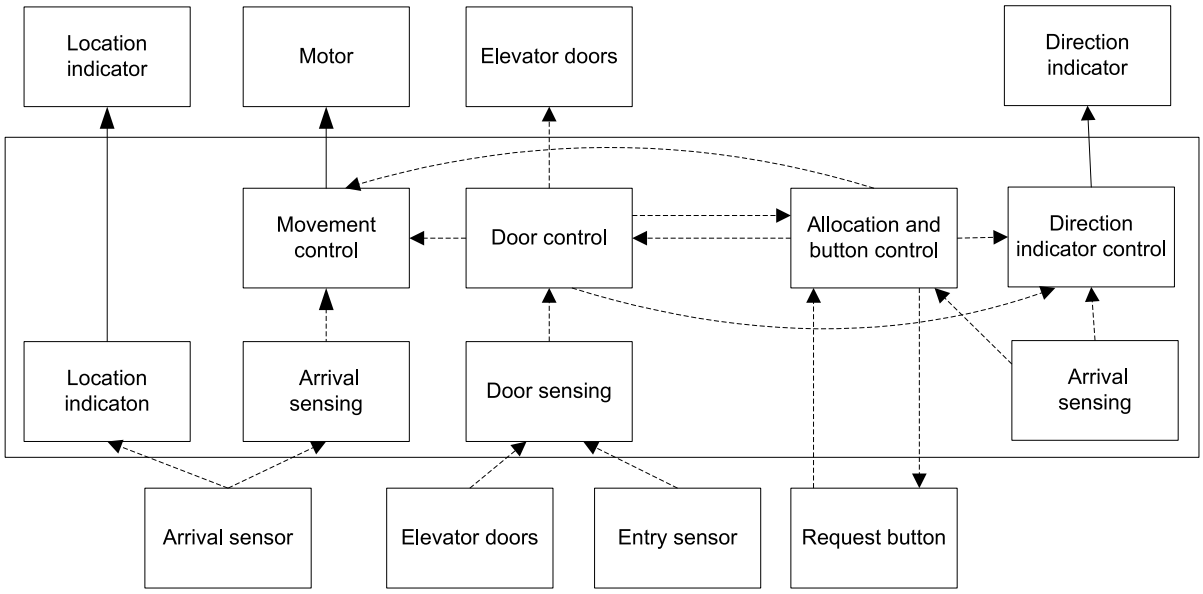

Figure 4: A decomposition of an elevator controller.

(a) Draw a context diagram of the elevator controller. `Just omit everything inside the controller.`

(b) The diagram shows data flows and event flows. What is the difference between the two? `Pages 187-188`

(c) A requirements-level architecture can be designed using the following guidelines:

G1 Functional decomposition
G2 Subject-oriented decomposition
G3 Event-oriented decomposition
G4 Device-oriented decomposition
G5 User-oriented decomposition
G6 Behavior-oriented decomposition

Explain each of these guidelines. `Pages 241 and further`

(d) Classify each of the components in figure 4 according to the above guidelines. `Page 254 exercise 8`

Problem	a	b	c	d	
1	6	4	4		14
2	6	4	4		14
3	6	6	6		18
4	8	8	8		124
5	4	4	6	6	26
					90

Grade = (10 + points)/10