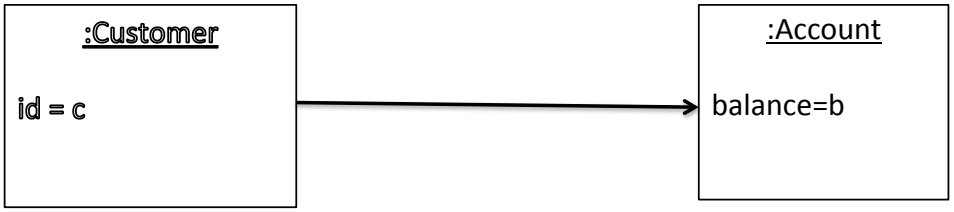
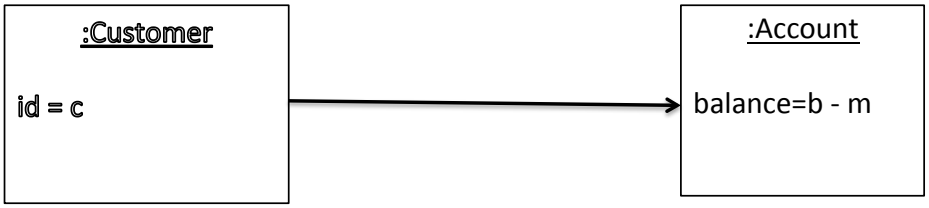
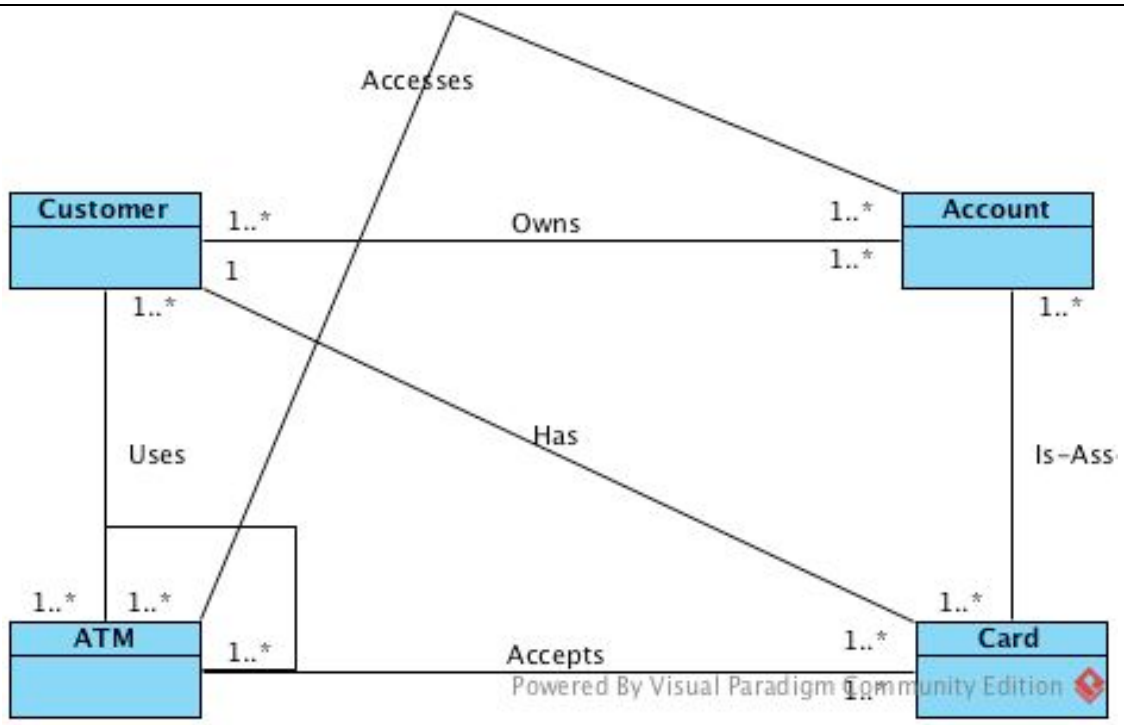
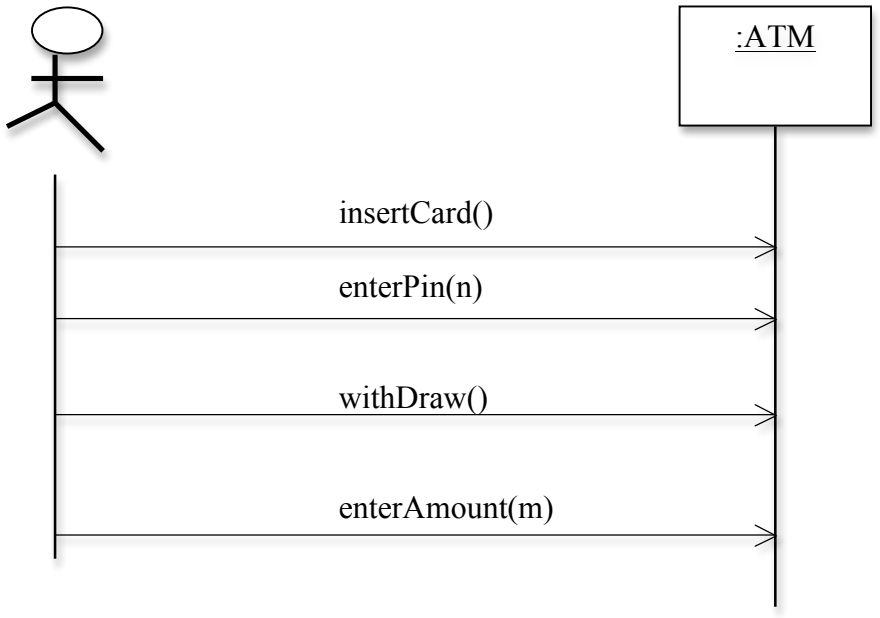
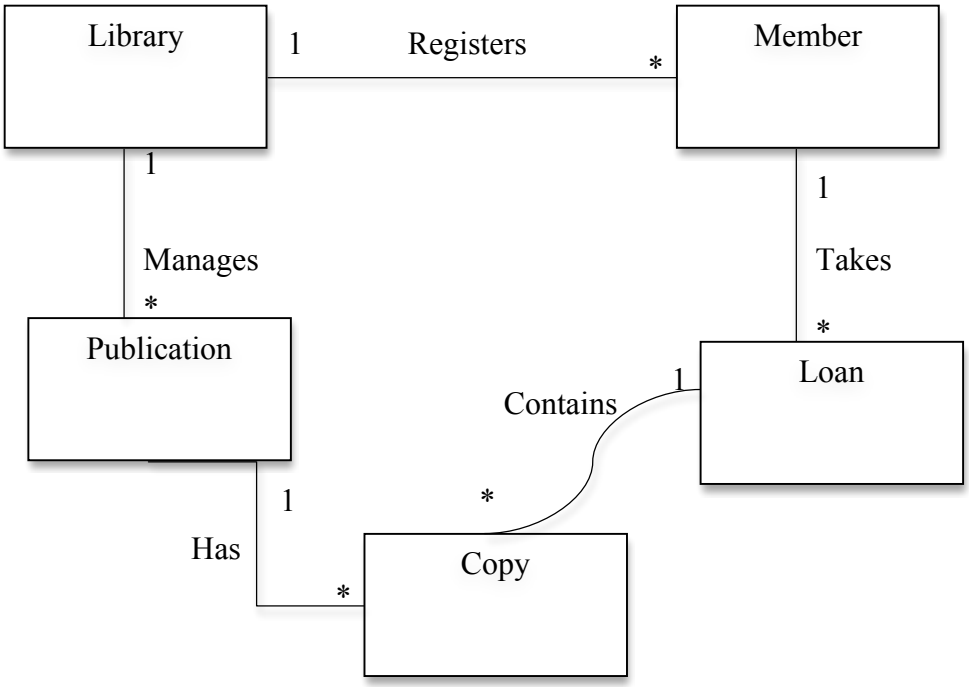
Question N°/Part	Module Title: Software Design ..... Exam date: …….....	Marks
1(a)	i. Use case identification through understanding of the application, use case decription and system functions. ii. Use case diagram and Use case sequence diagrams. iii. Identification of concepts as (conceptual classes) and understanding their attributes and association, Conceptual class diagram. iv. Identification of system operation and specification of their contracts in terms of pre and post conditions.	3 marks for each point
1(b)	i. -- Take each operation of each use case in turn. Extend the use case sequence diagram with design of the operation by assigning responsibilities to the objects of the classes in the conceptual class diagram, -- following the controller Pattern and the Expert Pattern. The high cohesion and low coupling patterns should be considered during the design for better design. ii. --The design will refine the conceptual class diagram to a designclass diagram; -- and the use case sequence daiagrams to object seuqnce diagrams or collaboration daiagrams.	3 marks for each item 3 marks each item
1(c)	i. Go through each of the use case sequence diagram, taking each operation in turns, in the order in which they occur in the sequence diagram. ii. Check if in the corresponding object sequence diagram (or collaboration diagram) if the preconditions are checked. iii. Check in the corresponding object sequence diagram (or collaboration diagram) if the object required to created in the post conditions of the operation actually created in the object sequence diagram, attributes required to be modified are actually modified, and links of associations required to be formed are actually formed. iv. Check if the object creaded, attributes modified and associations formed in for postconditions are actually in the conceptual class diagram. v. Check if the methods in the object sequence diagrams (or collaborations diagrams) are properly recorded in the design class diagrams. vi. Check if all operations and all operations are covered.	1 mark for each point
	Note: Credits are given to evidence of general understanding, rather than correctness of details	

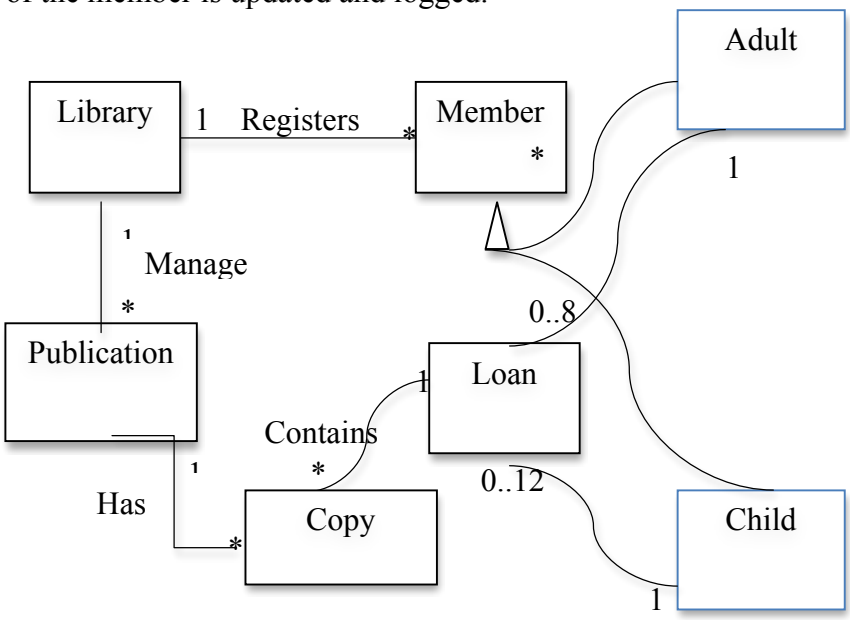
2(a)	i. Small Bank, Big Bank ii. Small Bank, Big Bank iii. Only the Big Back, as one customer should have at least two different accounts this is only allowed by the Big Bank	2 marks for each bullet point
2(b)	i. withDraw(c, m): Precondition: customer c with id equal to c exists customer c has an account, say a, the balance of a, a.balance, is not less than m Postcondition: the balance of a is changed to a.balance – m	4 marks
	ii. 	3 marks
	iii. 	3 marks
2(c)		

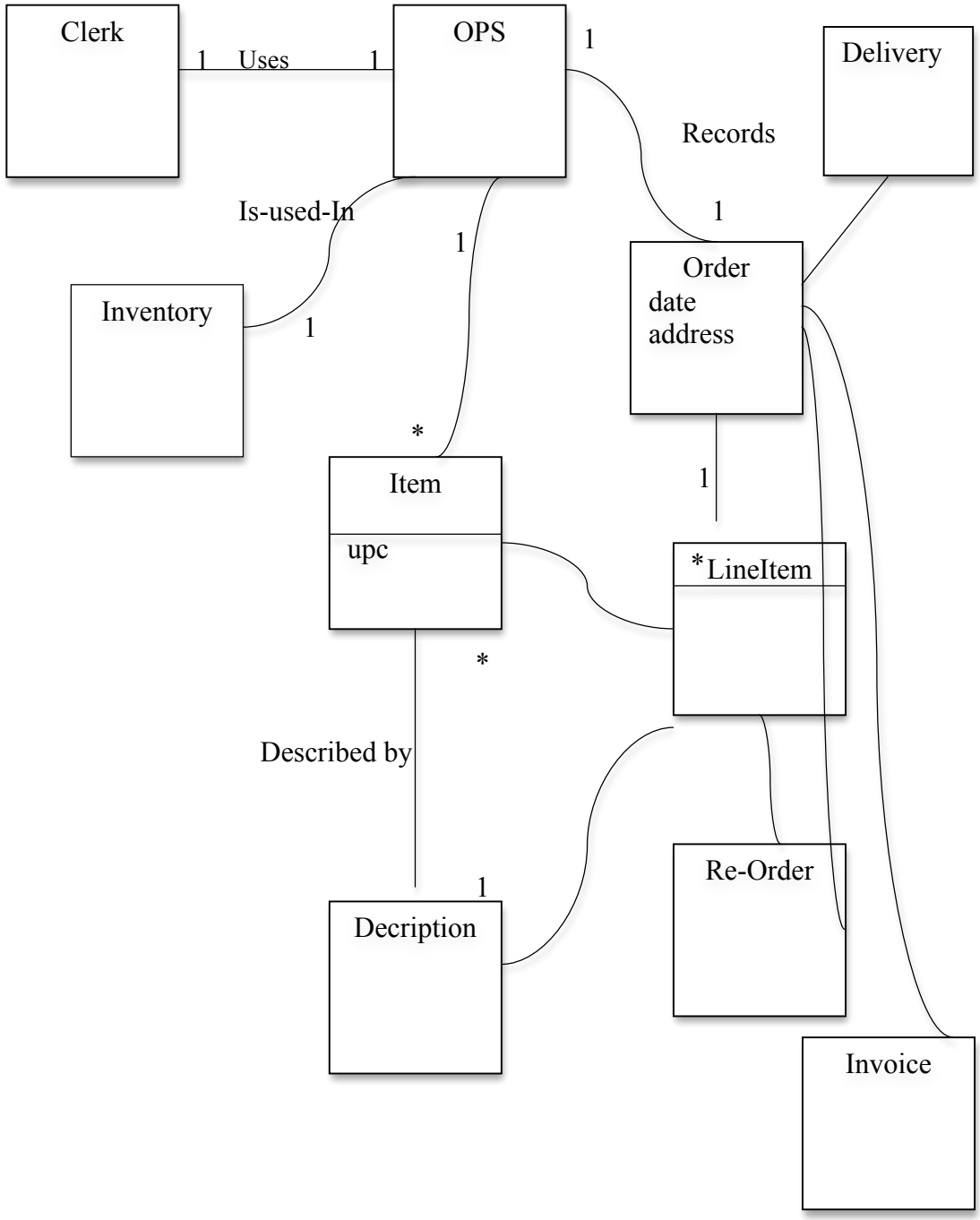


Attributes are omitted. Answers with fewer assocaitions as some can be derived.

2(d)	The sequence diagram contains two life lines, the Actor that a is a customer and the system ATM, respectively, and then the following operations in order: insertCard, enterPin(n), withDraw(), enterAmount(m)  	10 marks: 3 marks for lifelines, 4 for operations, and 3 for the order

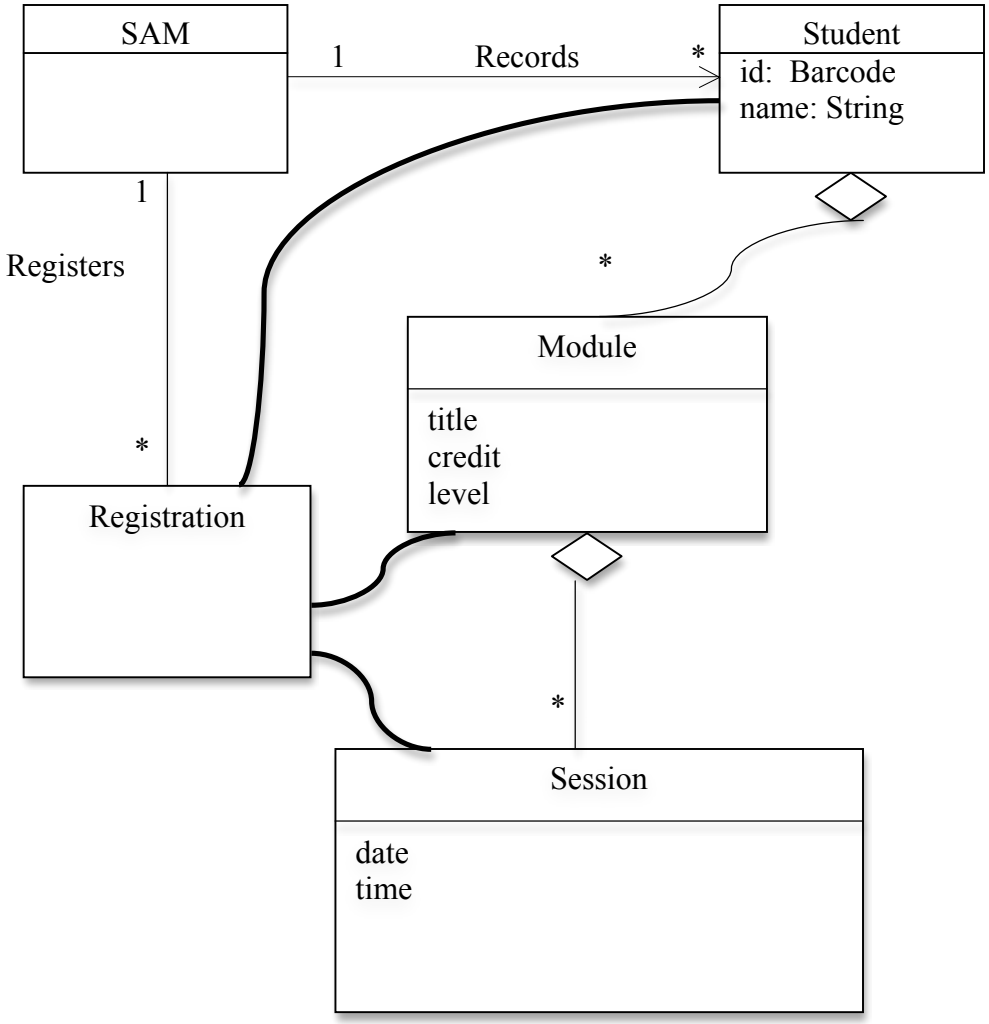
3(a)	i. Use case Borrow Copies 1. The Member comes to the checkpoint with the copies she is to borrow 2. The Librarian checks the validity of the membership of the memner 3. The Librarian enter each copy to the system; 4. a record of the borrow of copy was created and assocayed to to the the member's Loan 5. After entering the last copy, the Librarian indicaye the end of the entry, Loan of the member is updated and logged. ii. Some basic attributes should be added into the following  	10 marks in total : 3 for steps, 4 for operations, and 3 for orders 10 marks: 4 marks for classes, 4 marks for associations, and 2 marks for multiplicities

3(b)	i. Refined Use case: Borrow Copies 1. The Member comes to the checkpoint with the copies she is to borrow 2. The Librarian checks the validity of the membership of the memner 3. The Librarian enter each copy to the system; 4. a) if the member is a child and her loan currently has less than 12 copies, a record of the borrow of copy was created and assocayed to to the the member's Loan b) if the the member is an adult and her loan currently has less than 8 copies, a record of the borrow of copy was created and assocayed to to the the member's Loan 5. After entering the last copy, the Librarian indicaye the end of the entry, Loan of the member is updated and logged. ii.  Note: look for the super-subclass relation	1 mark for steps 1&2, 2 marks for step 3, 2 marks for step 4(a), 1 mark for step 4(b), 1 mark for step 5: 7 marks in total 4 marks for the inheritance relation, and 4 marks for the rest of diagram

4(a)		Note not all names and multiplicities are given in the diagram. They will be checked during the marking: 4 marks for classes, 2 marks for attributes, 4 marks for associations and 3 marks for multiplicities

4(b)	The use case sequence diagram is in the shape of the soltion for 2(d), but the operations only include enterItem(upc, qty), endEntry() But a loop is expected	10 marks: 3 marks for lifelines, 4 for operations, and 3 for the order
4(c)	enterItem(): Precondition: upc is known by OPS Postcondition: if it is the fistitem entered, an Order was created, and Delivery was acreated, and the delivery is assocated to the Order; if qty is not bigger than the inventory of the product in stock, a LineItem was created, and subtotal was set according to price (according to specification through upc matching); if qty is bigger than the inventory, a Reorder of the item is creaded. endEntry() Precondition: true Postcondition: An Invoicewas created; the Invoice was associated to the Order; the Order and its associated Delivery, Reoder and Involice were logged to the inventory	2 marks one mark for each condition One mark for each condition

5(a)

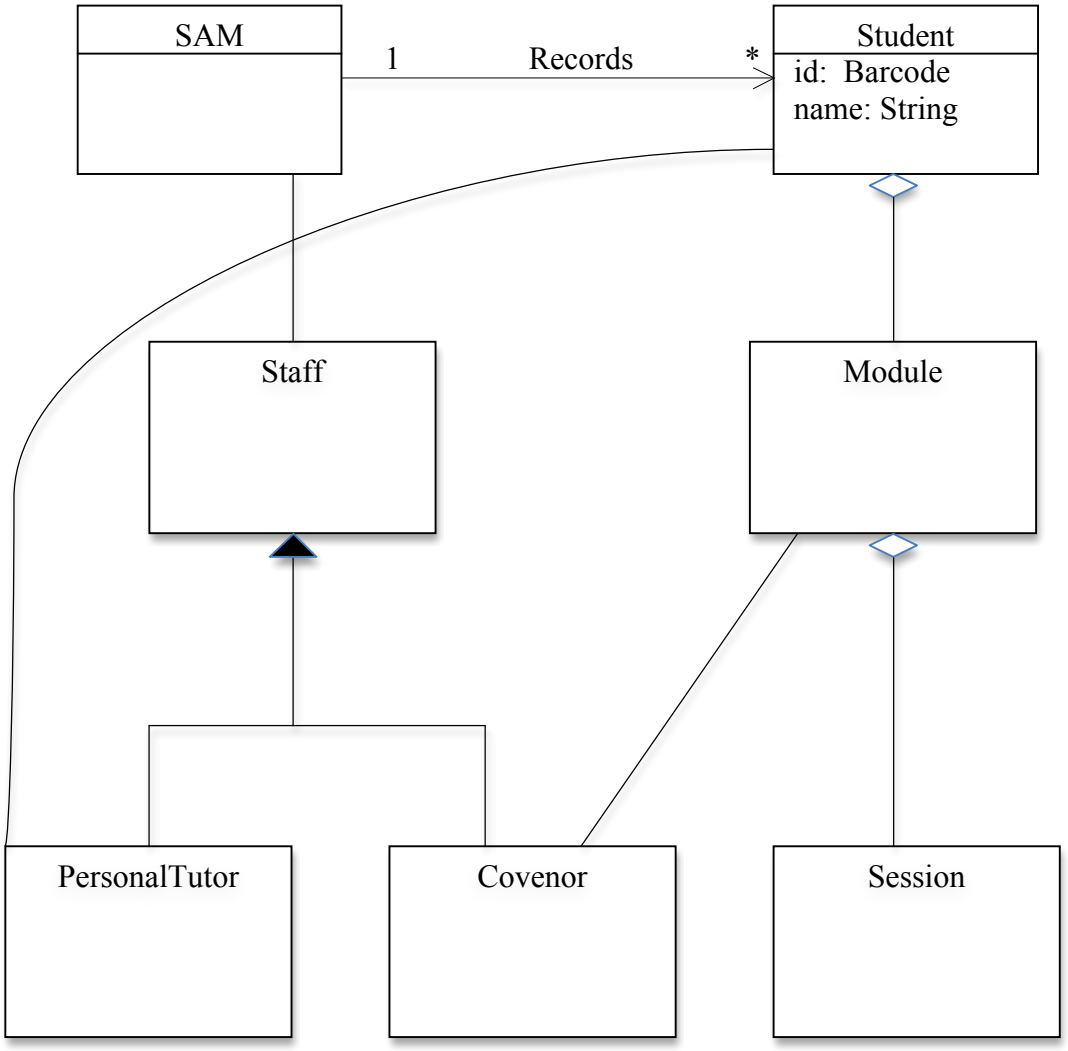


Note: look for th aggregation assocations

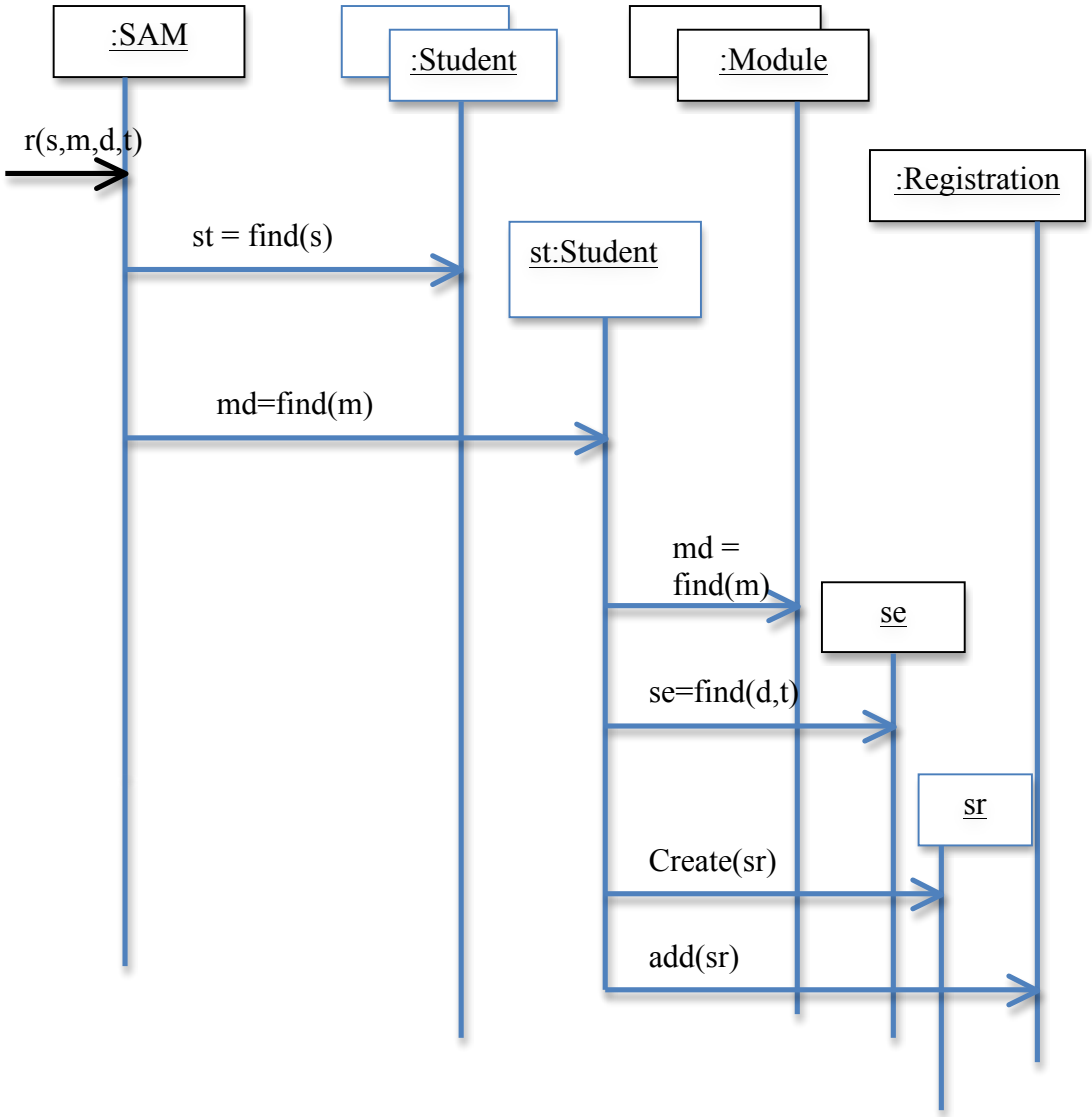
This is the simplest possible, multiplicities are both 1 to many.

Better models can include Canlendar, and a class ``Venue'' for better timetabling

4 marks for classes, 4 marks for associations, 3 marks for attribites, and 2 marks for aggreagtion

5(b)	It is acceptable if the inheritance is not used, instead two separate classes "Convenor" and "PersonalTutor" are introduced and associated Module and Student approrately. The registration class in 5(a) and its associations should be added into the following model, and details of the classes are the same as part 5(a) **Note: Look for the Generalistion-Specialization relation**.  	4 marks for inheritance; 4 marks for aggregation, 2 marks for classes and 2 marks for associations

5(c)



Note: there can be different designs

4 marks for liflines; 3 marks for operations, and 3 marks for order