**Topic : Software Modeling, Software Design**

1. What is software modeling?

(a) Developing models of software.

(b) Designing software applications before coding.

(c) Developing software diagrams.

(d) Developing software prototypes.

1. What is the Unified Modeling Language?

(a) A programming language for describing object-oriented models.

(b) A diagramming tool for drawing object-oriented models.

(c) A graphical language for describing object-oriented models.

(d) A standardized graphical language and notation for describing object oriented models.

1. What is a software architecture?

(a) The software inside a building.

(b) The structure of a client/server system.

(c) The overall structure of a software system.

(d) The software classes and their relationships.

1. What is a software design notation?

(a) Notes about the software design.

(b) A graphical or textual description of the software.

(c) Documentation of the software.

(d) A systematic approach for producing a design.

1. What is a software design concept?

(a) A graphical or textual description of the software.

(b) Documentation of the software.

(c) A fundamental idea that can be applied to designing a system.

(d) A systematic approach for producing a design.

1. What is a software design strategy?

(a) A graphical or textual description of the software.

(b) A fundamental idea that can be applied to designing a system.

(c) A systematic approach for producing a design.

(d) An overall plan and direction for developing a design.

1. What are software structuring criteria?

(a) Fundamental ideas that can be applied to designing a system.

(b) Systematic approaches for producing a design.

(c) Guidelines used to help in structuring a software system into its components.

(d) Overall plans for developing a design.

1. What is a software design method?

(a) A systematic approach for producing a design.

(b) Guidelines used to help in structuring a software system into its components.

(c) An overall plan for developing a design.

(d) A graphical or textual description of the software.

**Topic : UML**

1. What is true about UML stereotypes?

**(a) A stereotype is used for extending the UML language.**

(b) A stereotyped class must be abstract.

(c) The stereotype {frozen} indicates that the UML element cannot be changed.

(d) UML Profiles can be stereotyped for backward compatibility.

**Topic : Software Testing**

1. What is the goal of software validation?

(a) Building the system

(b) Building the right system

(c) Building the system right

(d) Testing the system

1. What is the goal of software verification?

(a) Building the system

(b) Building the right system

(c) Building the system right

(d) Testing the system

1. What is “white box” testing?

(a) Unit testing

(b) Integration testing

(c) Testing with knowledge of the system internals

(d) Testing without knowledge of the software internals

1. What is “black box” testing?

(a) System testing

(b) Integration testing

(c) Testing with knowledge of the system internals

(d) Testing without knowledge of the software internals

**Topic : Object Oriented Concepts, Class Diagram**

1. Which of the following are object-oriented concepts?

(a) Modules and interfaces

(b) Modules and information hiding

(c) Classes, information hiding, and inheritance

(d) Concurrency and information hiding

1. Which of the following is a characteristic of an object?

(a) A function or subroutine

(b) A module

(c) Groups data and procedures that operate on the data

(d) Groups a function and an algorithm

1. What is a class?

(a) An object instance

(b) The implementation of the object

(c) A collection of objects with the same characteristics

(d) A collection of objects with different characteristics

1. What is an operation (also known as method) of a class?

(a) Specification and the implementation of a function performed by a class

(b) Specification and the implementation of a subroutine provided by a class

(c) Specification and the implementation of a function or procedure provided by a class

(d) Specification and the implementation of an interface provided by a class

1. What is the signature of an operation?

(a) The operation’s name

(b) The operation’s function or subroutine

(c) The operation’s name, parameters, and return value

(d) The object’s interface

1. What is the interface of a class?

(a) The signature of a class

(b) The specification of operations provided by the class

(c) The internals of the class

(d) The implementation of the class

1. What is an attribute?

(a) A description of a class

(b) An internal property of a class

(c) A data item held by a class

(d) A parameter of a class

1. What is information hiding in software design?

(a) Hiding information so that it cannot be found

(b) Hiding a design decision that is considered likely to change

(c) Hiding information to make it secure

(d) Encapsulating data in a class

1. What is data abstraction?

(a) Another name for information hiding

(b) Encapsulating data so that its structure is hidden

(c) Storing data in a database

(d) Storing data in a data structure

1. What is inheritance?

(a) A mechanism for inheriting characteristics from a parent

(b) A mechanism for sharing and reusing code between classes

(c) A mechanism for sharing data between classes

(d) A mechanism for hiding information between classes

1. What is carried out during analysis modeling?

(a) Developing use case models

(b) Developing data flow and entity-relationship diagrams

(c) Developing static and dynamic models

(d) Developing software architectures

1. What is carried out during design modeling?

(a) Developing use case models

(b) Developing data flow and entity-relationship diagrams

(c) Developing static and dynamic models

(d) Developing software architectures

1. What is carried out during system testing?

(a) White box testing

(b) Black box testing

(c) Unit testing

(d) Integration testing

**Topic : Use Case Model**

1. How is an actor depicted on a use case

diagram?

(a) An oval

(b) A stick figure

(c) A box

(d) A dashed line

1. How is a use case depicted on a use case diagram?

(a) An oval

(b) A stick figure

(c) A box

(d) A dashed line

1. How is a class depicted on a class diagram?

(a) A box with one compartment

(b) A box with one or two compartments

(c) A box with one, two, or three compartments

(d) An oval

1. How is an association depicted on a class diagram?

(a) A solid line joining two class boxes

(b) A dashed line joining two class boxes

(c) A diamond touching the upper class box

(d) An arrowhead touching the upper class box

1. How is public visibility depicted for a class element on a class diagram?

(a) + sign

(b) − sign

(c) # sign

(d) ∗sign

1. What are the two kinds of UML interaction diagrams?

(a) Class diagram and sequence diagram

(b) Sequence diagram and communication diagram

(c) Class diagram and communication diagram

(d) Statechart and communication diagram

1. What does an interaction diagram depict?

(a) Objects and links

(b) Classes and relationships

(c) Objects and messages

(d) States and events

1. What does a statechart diagram depict?

(a) Objects and links

(b) Classes and relationships

(c) Objects and messages

(d) States and events

1. What is a UML package?

(a) A box

(b) A grouping of classes

(c) A grouping of use cases

(d) A grouping of model elements

1. What does a deployment diagram depict?

(a) The physical configuration of the system in terms of physical classes and physical connections between

the classes

(b) The physical configuration of the system in terms of physical objects and physical connections between

the objects

(c) The physical configuration of the system in terms of physical nodes and physical connections between

the nodes

(d) The physical configuration of the system in terms of physical computers and physical networks between

the computers

1. In UML deployment diagrams (indicate all correct answers):

a) artifacts are represented as pyramids.

b) Answer nodes are represented as cubes.

c) use cases are represented as cylinders.

d) information assets are represented as ellipses.

1. What is a use case?

(a) A case study involving users

(b) A sequence of interactions between the user and the system

(c) A sequence of interactions between the user and the objects in the system

(d) A sequence of user inputs to the system

1. What is an actor in a use case?

(a) An object inside the system

(b) A person who performs on stage

(c) An external entity that interacts with the system

(d) The customer to whom the system will be delivered

1. What is a primary actor?

(a) The actor who goes on stage first

(b) The actor that starts the use case

(c) An actor that participates in the use case

(d) An object inside the system

1. What is a secondary actor?

(a) The actor who goes on stage second

(b) The actor that starts the use case

(c) An actor that participates in the use case

(d) An object inside the system

1. What is an alternative sequence in a use case?

(a) A sequence that describes an error case

(b) A sequence that is different from the main sequence

(c) A sequence that describes interactions with a secondary actor

(d) A sequence that describes interactions with a primary actor

1. What can an inclusion use case be used for?

(a) To describe an inclusive use case

(b) To describe a lengthy interaction with an actor

(c) To describe functionality that is common to more than one use case

(d) To describe a use case that includes other use cases

1. What can an extension use case be used for?

(a) To describe a lengthy interaction with an actor

(b) To describe functionality that is common to more than one use case

(c) To describe the functionality of a use case that is extended by another use case(s)

(d) To describe a conditional part of a different use case that is only executed under certain circumstances

1. What can an activity diagram be used for in use case modeling?

(a) To depict the sequence of activities executed by all the use cases in the system

(b) To depict the sequence of external activities that the use case interacts with

(c) To depict the sequence of active objects in a use case

(d) To depict the activities in the main and alternative sequences of a use case

1. How can a nonfunctional requirement be described in a use case model?

(a) In a separate section of the use case description

(b) As a use case precondition

(c) As a use case postcondition

(d) In a separate document

1. Which of the following statements pertaining to actors in use cases are FALSE?

**a) Answer In presence of generalization a more general actor can always replace the specialized one.**

b) One user might be represented by multiple actors in the same use case.

c) Actor is a class of entities (human or computer), falling beyond the system

boundaries and interacting with the system.

d) One use case might involve multiple actors.

1. Which diagram is NOT commonly used for illustrating use cases?

(a) system sequence diagram

(b) activity diagram

(c) use case diagram

**(d) collaboration diagram**

1. Consider a beverage machine. If the actor is ‘customer’, and the scope is ‘machine’, what is most likely to be found in the main scenario of the use case ‘get drink’?

(a) - enter choice

- if drink available then show price

- put in coins

- if paid enough then deliver drink

**(b) - customer enters choice**

- machine shows price

- customer puts in coins

- machine delivers drink

(c) - enter choice

- show price

- put in coins

- deliver drink

(d) - ...

- machine sends price to LCD display

- customer put coins in slot

- coin mechanism verifies amount and tells machine controller

- machine controller activates boiler

1. What is a class?

(a) A course

(b) An object instance

(c) A client or server in the system

(d) A collection of objects with the same characteristics

1. What is an attribute?

(a) A relationship between two classes

(b) A parameter of an operation or method

(c) A data value held by an object in a class

(d) The return value from an operation

1. What is an association?

(a) A relationship between two classes

(b) A relationship between two objects

(c) A link between two classes

(d) A link between two objects

1. What is meant by the multiplicity of an association?

(a) The number of associations in a class

(b) The number of associations between two classes

(c) How many instances of one class relate to how many instances of another class

(d) How many instances of one class relate to a single instance of another class.

1. What is an association class?

(a) A class with multiple associations

(b) A class with one association

(c) A class that models an association between two or more classes

(d) A class that models an association between two or more objects

1. Which statements are implied by the class diagram in Figure :

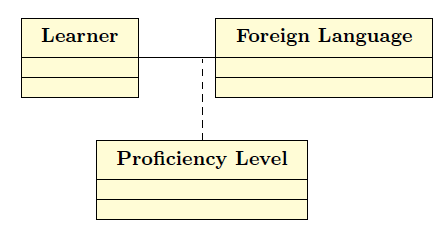
a) Every learner can study only one foreign language.

b) Every learner can have only one proficiency level per foreign language.

c) Every foreign language can be studied only by one learner.

d) Every proficiency level can be recorded for at most one (learner, foreign language)

pair.



1. What is a generalization/specialization hierarchy?

(a) A whole/part relationship

(b) An inheritance relationship

(c) An association between a generalized class and a specialized class

(d) A layered hierarchy

1. What is a composition hierarchy?

(a) A weak form of a generalization/specialization hierarchy

(b) A strong form of a generalization/specialization hierarchy

(c) A weak form of a whole/part relationship

(d) A strong form of a whole/part relationship

1. What is an aggregation hierarchy?

(a) A weak form of a generalization/specialization hierarchy

(b) A strong form of a generalization/specialization hierarchy

(c) A weak form of a whole/part relationship

(d) A strong form of a whole/part relationship

1. What does the system context class diagram define?

(a) The entity classes in the system

(b) How the system interfaces to other systems

(c) The boundary between the system and the external environment

(d) The context classes in the system

1. What is an entity class?

(a) A class on an entity/relationship diagram

(b) A class that stores data

(c) A class that interfaces to an external entity

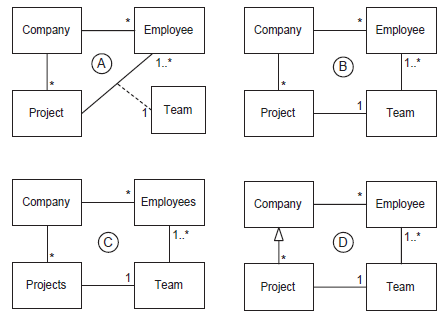
(d) An external class

Consider the following situation:

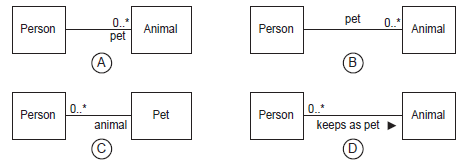
a company realizes projects; each project is executed by a team of employees.

Which would be a suitable conceptual UML diagram?

(a) diagram A (b) diagram B (c) diagram C (d) diagram D



1. How do you express that some persons keep animals as pets?



(a) diagram A (b) diagram B (c) diagram C (d) diagram D

1. What can UML interfaces be used for?

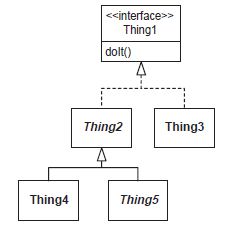
(a) to provide concrete classes with the stereotype <<interface>>

(b) to program in Java and C++, but not in C#

(c) to define executable logic that can be reused in several classes

(d) to specify required services for types of objects

1. Which is true about the method doIt()?



(a) The method doIt() must be implemented by Thing3 and possibly also by Thing4.

(b) The method doIt() must be implemented only by Thing5.

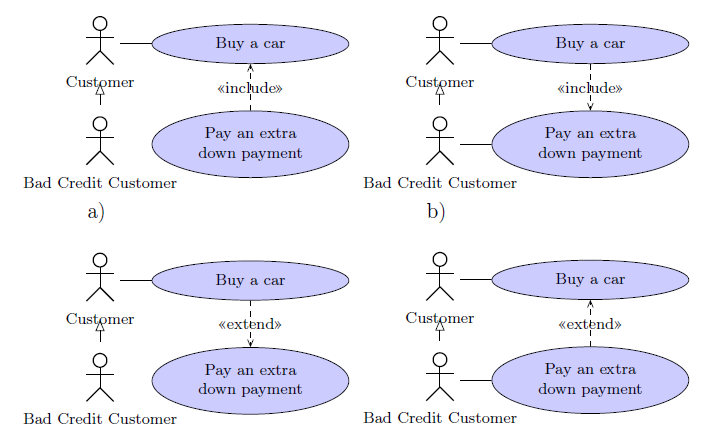
(c) The method doIt() must be implemented by Thing2, Thing3, Thing4 and Thing5.

(d) There is no need for any class to implement doIt(), because it is already implemented by Thing1.

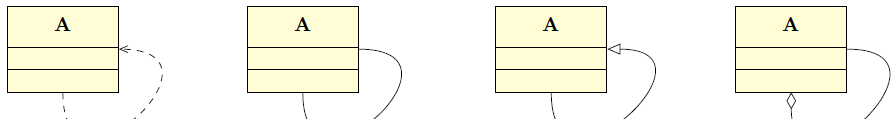
1. Read the following description. “Customers of the garage can buy cars. Customers

with a bad credit should pay an extra down payment”. Which of the following

diagrams represent this description? (D)



1. Which of the following class diagrams is wrong? (C)

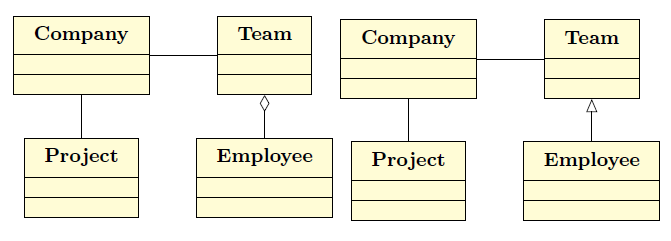


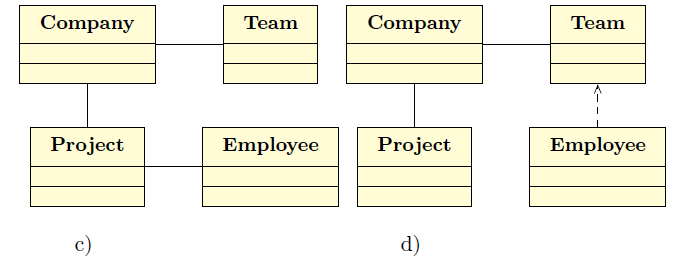
1. Read the following description and select the most appropriate class diagram from

Figure : “A company realizes projects; each project is executed by a

team of employees”. For the sake of simplicity, multiplicities and attributes are not

indicated.





1. What does a sequence diagram depict?

(a) The sequence of external objects communicating with each other

(b) Classes and their relationships

(c) Software objects and the sequence of their interactions

(d) The external objects communicating with the system

1. What is true about a Sequence Diagram? [2 answers]

[a] It describes the behaviour in many Use Cases.

[b] It describes the behaviour in a single Use Case.

[c] It describes the behaviour of a single object.

[d] It describes the behaviour of several objects.

1. I want to know in detail about the difference between alt and opt fragment in sequence diagram, they seem similar, I can’t distinguish them. Anyone knows about this thing?” Which of the following answers are correct?

a) Answer alt is more used for several choices, like a switch, while with opt code will

be executed or not!

b) Answer alt is used to described alternative scenarios. Only one of the options will

be executed. opt is used to describe an optional step.

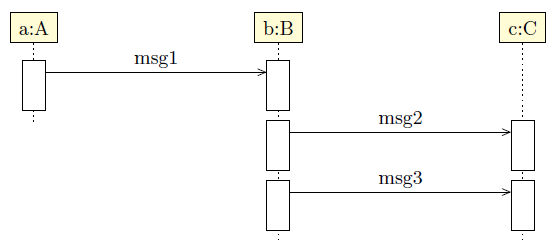
c) an opt fragment cannot be enclosed in other combined fragments such as strict

or par, while alt can be enclosed in such fragments.

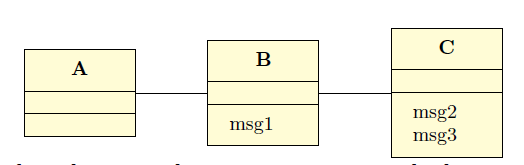
d) an alt fragment cannot be enclosed in other combined fragments such as strict

or par, while opt can be enclosed in such fragments.

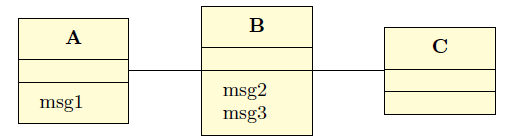
1. Consider the following sequence diagram.



Which of the following statements are TRUE?



a) The class diagram above is consistent with the sequence diagram.

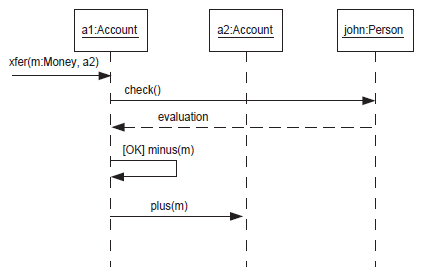


b) The class diagram above is consistent with the sequence diagram.

c) Message msg2 is received before msg3 has been sent.

d) Answer Message msg2 is sent after msg1 has been received.

1. Given the following diagram, which method(s) should be implemented for the Account class?



(a) xfer()

(b) xfer(), plus(), minus()

(c) check(), plus(), minus()

(d) xfer(), evaluation(), plus(), minus()

1. What is a state in a state machine?

(a) A recognizable situation that exists over an interval of time

(b) A condition that is True or False

(c) An input from the external environment

(d) An output from the system

1. What is an event in a state machine?

(a) A discrete signal that causes a change of state

(b) An input from the external environment

(c) An input that is True or False

(d) The result of a state transition

1. What is an action in a state machine?

(a) An occurrence at a point in time

(b) A cause of a state transition

(c) An interval between two successive events

(d) A computation that executes as a result of a state transition

1. What is an entry action in a state machine?

(a) An action that is performed when the state is entered

(b) An action that is performed when the state is left

(c) An action that starts executing when the state is entered and completes executing when the state is

left

(d) An action that executes as a result of a state transition

1. What is an exit action in a state machine?

(a) An action that is performed when the state is entered

(b) An action that is performed when the state is left

(c) An action that starts executing when the state is entered and completes executing when the state is

left

(d) An action that executes as a result of a state transition

1. What is a condition used for in a state machine?

(a) A conditional action

(b) A conditional state

(c) A conditional state transition

(d) A conditional event

1. What is a state transition into a composite state equivalent to?

(a) A transition into only one of the substates

(b) A transition into each of the substates

(c) A transition into none of the substates

(d) A transition into any one of the substates

1. What is a state transition out of a composite state equivalent to?

(a) A transition out of only one of the substates

(b) A transition out of each of the substates

(c) A transition out of none of the substates

(d) A transition out of any one of the substates

1. How does a composite state relate to a substate?

(a) A composite state is decomposed into substates.

(b) Composite states are composed into substates.

(c) A composite state transitions to a substate.

(d) A substate transitions to a composite state.

1. If two actions are shown on a given state transition, which of the following is true?

(a) The two actions are dependent on each other.

(b) The two actions are independent of each other.

(c) One action provides an input to the other action.

(d) The second action executes when the first action completes execution.

1. Which are valid events in a State diagram? [2 answers]

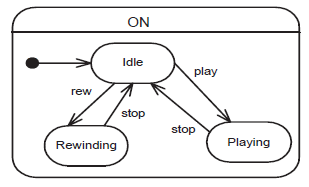
[a] if()

[b] when()

[c] close()

[d] after()

1. What is correct about the following State Diagram?



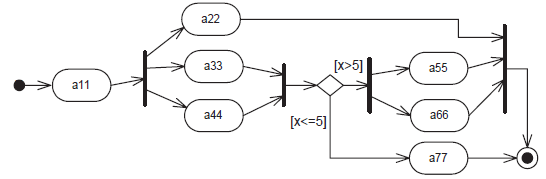
(a) ‘ON’ is a concurrent state.

(b) This State Diagram is invalid because it contains no final state.

(c) ‘play’, ‘stop’ and ‘rew’ are actions.

(d) ‘ON’ is a superstate.

1. Which of these activities COULD occur simultaneously? [2 answers]



[a] a44 and a66 [b] a44, a33 and a22 [c] a22 and a77 [d] a77 and a66

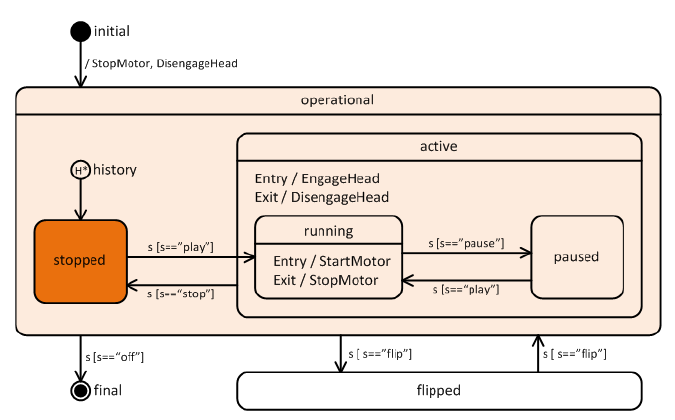
1. A controller for a cassette player is shown on Figure. The cassette

player can be controlled by buttons including play, pause, stop, flip and off. The

player has a motor that can be started or stopped, and a playback head that can

be engaged or disengaged. The head serves as an interface between the player and

the cassette.



Which of the following statements are TRUE?

a) If no previous history is available, then the stopped state is entered.

b) If the user presses play, then pause, followed by flip and flip, then the motor is

running.

c) Correctness of answer would be changed if H\*would have been replaced by

H.

d) Answer Whenever the motor is running, the head is engaged.

1. What do software quality attributes address?

(a) Software functional requirements

(b) Software nonfunctional requirements

(c) Software performance requirements

(d) Software availability requirements

1. What is maintainability?

(a) The extent to which software iscapable of being changed beforedeployment

(b) The extent to which software iscapable of being changed afterdeployment

(c) The extent to which software iscapable of being changed duringdevelopment

(d) The extent to which software iscapable of being changed afterdevelopment

1. What is modifiability?

(a) The extent to which software iscapable of being modified afterdeployment

(b) The extent to which software iscapable of being modified after initialdevelopment

(c) The extent to which software iscapable of being modified duringand after initial development

(d) The extent to which software iscapable of being changed beforedeployment

1. What is testability?

(a) The extent to which software iscapable of being developed

(b) The extent to which software iscapable of being tested beforedeployment

(c) The extent to which software is capableof being tested after deployment

(d) The extent which the software isunderstood

1. Traceability is the extent to which aproduct:

(a) Can be traced back to products ofprevious phases

(b) Traced back to the requirements

(c) Traced forward to implementation

(d) Deployed to a hardware configuration

1. What is scalability?

(a) The extent to which an applicationcan grow

(b) The extent to which the system iscapable of growing after its initialdeployment

(c) The extent to which the system iscapable of growing during development

(d) The extent to which the system iscapable of being scaled

1. What is reusability?

(a) The extent to which software implementationis reusable

(b) The extent to which software iscapable of being reused

(c) The extent to which SPL technologycan be introduced

(d) The extent to which the softwareis common among a programfamily

1. Which of the following is not performance-related?

(a) System response time

(b) System throughput

(c) System availability

(d) System capacity

1. Which of the following is not addressedby a secure system?

(a) System penetration

(b) Denial of service

(c) System scalability

(d) System authorization

1. Which of the following system problemsdoes availability address?

(a) Denial of service

(b) Single point of failure

(c) System throughput

(d) System penetration

1. What is a class interface?

(a) Specifies the internals of the operationsof a class

(b) Specifies the externally visibleoperations of a class

(c) Specifies the parameters of a classoperation.

(d) Specifies the signature of a classoperation

1. Which of the following is NOT anobject-oriented concept?

(a) Information hiding

(b) Class

(c) Subclass

(d) Subroutine

1. Which of the following is a class thatrealizes an interface?

(a) The class calls the interface.

(b) The class implements the interface.

(c) The class is called by the interface.

(d) The class is independent of theinterface.

1. Which of the following is an entityclass?

(a) An information hiding class

(b) A subclass

(c) A control class

(d) A data abstraction class

1. Which of the following is NOT allowedthrough inheritance?

(a) Subclass inherits attributes fromsuperclass.

(b) Subclass inherits operations fromsuperclass.

(c) Subclass redefines attributes inheritedfrom superclass.

(d) Subclass redefines operationsinherited from superclass.

1. Which of the following is true for anabstract class?

(a) It is used as a template for creatingobjects.

(b) It is used as a template for creatingsubclasses.

(c) It is used as a template for creatingclasses.

(d) It is used as a template for creatingsuperclasses.

1. In object-oriented design, polymorphismmeans that:

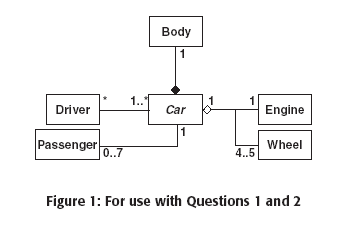
(a) Different classes may have the samename.

(b) Different classes may have the sameinterface name.

(c) Different classes may have the samesuperclass name.

(d) Different classes may have the sameoperation name.

With reference to Figure , what do you think is the most likely implementation of the relationship between Car and Engine? Choose only one option.



(a) A field, of type Car, in Engine.

(b) A class called CarEngine with one field of type Car and another field of type Engine.

(c) A field, of type Engine, in Car.

(d) A field, of type Engine, in Car and a field, of type Car, in Engine.

With reference to above Figure, which of the following statements are true? Choose all options thatApply.

A car always has the same body.

Some cars have spare wheels.

A car has one engine, and engines are not shared between cars.

All cars have either four or five wheels.

A car must have at least one driver.

Passengers cannot be drivers.

In a UML class diagram, how are objects distinguished from classes? Choose only one option.

Object labels are shown in italics.

Class labels have a box drawn around them.

Object labels are underlined.

PizzaBase Case Study

The PizzaBase restaurant wants to automate the ordering of pizzas by customers.Each table will be fitted with a touch-sensitive screen which customers can use to browse thepizzas on offer and select their choice.Two basic types of pizza will be offered: the Do-it-Yourself will have a base with tomato sauceonly and then customers can choose any number of toppings, at a fixed price per topping; thePrefab will come in several varieties, each with a fixed set of toppings. Every pizza can beordered with a deep crust or crispy base, and three sizes are available: 6 inch, 9 inch and 12 inch.Customers will also be able to order from a fixed set of drinks, such as cola and lemonadeflavours, each in large or small size. Once customers have confirmed their order, they will beshown the final price and, thereafter, the screen will display the progress of their food as it isbeing prepared and cooked. At the end of a meal, payment will be made in the conventional way.

With reference to the PizzaBase case study, which of the following is the most likely list ofattributes at the analysis stage? Choose only one option.

cola, base, price, size, lemonade, payment.

flavour, variety, payment, final, display, meal, tomato.

progress, variety, flavour, price, touchSensitive, size, drink.

(d) base, price, variety, size, progress, flavour.

With reference to the PizzaBase case study, which of the following is the most likely listof analysis classes? Choose only one option.

(a) Payment, Order, Drink, Topping, Pizza, Order, Restaurant, Base, Sauce.

(b) Customer, Table, Pizza, Topping, Drink, Restaurant, Order.

(c) PizzaBase, Cola, Restaurant, Lemonade, Customer, Do-it-Yourself, Prefab, Table,Order.

(d) Restaurant, Pizza, Topping, Display, Order, Payment, Touch.

(e) Screen, Order, Offer, Topping, Size, Meal, Pizza, Restaurant.

(f) Pizza, Customer, Cook, Table, Crust, Topping, Drink, Restaurant.

With reference to the PizzaBase case study, which of the following options are likely businessuse cases? Choose all options that apply.

Customer pays for meal.

Restaurant prepares meal.

Customer sees progress of food.

Customer chooses pizza.

Customer selects drink from display.

With reference to above Figure, which diagram is the best model of Pizzas in the PizzaBaserestaurant? Choose only one option.

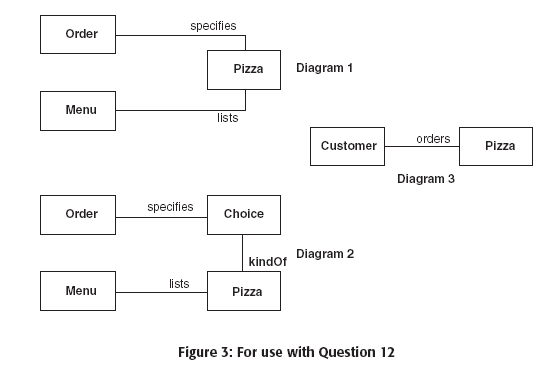
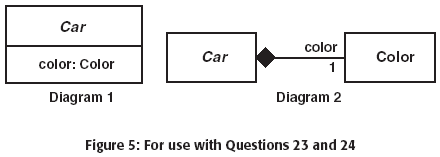


Diagram 1.

Diagram 2.

Diagram 3.

With reference to below Figure, what do Diagrams 1 and 2 illustrate? Choose only one option.



1: An aggregation, 2: A composition.

1: An attribute, 2: An aggregation.

1: An aggregation, 2: An attribute.

1: An attribute, 2: A composition.

1: A composition, 2: An attribute.

With reference to Figure, what is the difference between the two diagrams? Choose only one

option.

In Diagram 1, Color is public but in Diagram 2 Color is private.

Diagram 2 indicates that the car's Color can be removed and replaced.

Diagram 1 shows an abstract class and Diagram 2 shows a concrete class.

None, they mean the same thing.

Which of the following UML artifacts are used to show the distribution of processes, artifactsand model elements in a system? Choose only one option.

Interaction diagrams.

Sequence diagrams.

Deployment diagrams.

Communication diagrams.

State machine diagrams.

Class Diagrams.

Glossaries.

Which of the following terms best describes an object that is made up of other objects?   
Choose only one option.

Generalization.

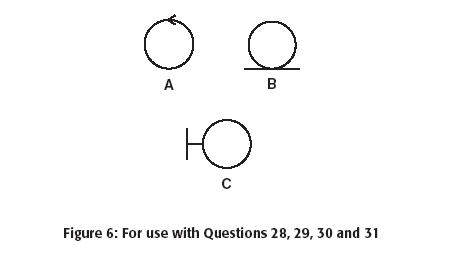
Inheritance.

Association.

Aggregation.

Specialization.

With reference to Figure, what kind of objects are A, B and C? Choose only one option.



A is an entity, B is a controller, C is a boundary.

A is a boundary, B is an entity, C is a controller.

A is an entity, B is a boundary, C is a controller.

A is a controller, B is an entity, C is a boundary.

A is a boundary, B is a controller, C is an entity.

A is a controller, B is a boundary, C is an entity.

With reference to Figure, which kind of icon would you use to represent a business object containing useful information? Choose only one option.

A (b) B (c) C

117. With reference to Figure, which kind of icon would you use to represent a communication path between systems or between a human and a system? Choose only one option.

(a) A (b) B (c) C

118. With reference to Figure, which kind of icon would you use to represent an object that coordinates a system process, creates objects or retrieves objects? Choose only one option.

(a) A (b) B (c) C

119. In UML diagrams, how are abstract classes distinguished from concrete classes? Choose onlyone option.

(a)Concrete classes are shown as boxes with dashed outlines.

(b)Labels on abstract classes are shown in italics.

(c)Labels on concrete classes are shown in italics.

(d)Abstract classes are shown as boxes with dashed outlines.

120. What does the term "polymorphism" refer to? Choose all options that apply.

(a)The ability of a variable to point at different classes of object at different times.

(b)The fact that a message with the same signature can invoke different methods at different times.

(c)All object-oriented programming languages are different.

(d)All object-oriented methodologies use a different notation.

121. What is "encapsulation"? Choose only one option.

(a)Depicting objects using state machine diagrams.

(b)Ensuring that the data inside an object can only be accessed via operations.

(c)Sealing the state of an object so that it cannot be changed.

(d)Putting objects into a collection.

122. In UML, which diagrams are used to show messages sent between objects? Choose all options that apply.

(a)Activity diagrams.

(b)Object diagrams.

(c)State machine diagrams.

(d)Sequence diagrams.

(e)Deployment diagrams.

123. What is an association class? Choose only one option.

(a) It describes the various kinds of relationship that can exist between classes.

(b) It adds attributes and/or behaviour to an association between two other classes.

(c) It associates an object with the class of which it is an instance.