Pay attention to following topics in which questions are frequently asked

**Use cases**

Use case multiplicity.

Ex: 2 instances of players are required to play tennis game

Ownership of use case can be either by class or package.

Use case generalization, concrete actors involved in use case.

Directed Relationship (include/extend) based questions.

**Sequence diagram**

Occurrence specification count

Execution specification count

Valid and invalid Execution trace with partial ordering constraint

**Actions & Activities**

No of actions in given diagram which may include Accept Event Action, Send Signal Action

Difference between action and activity (notation wise)

Invocation of Activity (notation with fork symbol)

Object node and control node difference with respect to transformation/Multicasting

Pseudo state properties

No of object nodes present in given diagram with respect to Pins.

**State Machines**

Transition with respect to deferred events.

Transition with guard constraints.

Group transition from composite states

Pseudo state and their properties (how differs from normal state)

When state machine is created for object?

Run to completion paradigm

**Classifiers**

Values section, mainly focus on Real Literal

Enumeration – Properties

Data type vs. classifiers

Generalization based questions

Visibility based on inheritance, package, namespace

Static property – notation

Notation for all properties.

When substitution allowed? Which relationship allows it?

Abstraction/Usage/Dependency – directed relationship

Features with respect to association member end

Properties

Composition vs Aggregation based question are FAQ

Association and association class

Package merge – out of syllabus

**Why We model**

PPTs are enough. However refer other sources for on safe side