**OMG-OCUP2-FOUND100 Exam Overview**

| | |
|---|---|
| **Exam Number** | OMG-OCUP2-FOUND100 |
| **Exam Duration** | 120 minutes in English-speaking countries (exception: city of Quebec) and 150 minutes in all others. |
| **Exam Fee** | US$250 (or local equivalent) in English-speaking countries (exception: city of Quebec) and US$260 (or local equivalent) in all others. |
| **Exam Type** | Multiple choice (text and UML diagrams) |
| **Exam Pass Score** | >=60 of 90 questions answered correctly (>=67%) |
| **Exam Prerequisite(s)** | None |
| **Exam Specification** | **Unified Modeling Language (UML) v.2.5.1** |
| **Recommended Exam Study Guides** | 1. OCUP 2 Certification Guide: Preparing for the OMG Certified UML 2.5 Professional 2 Foundation Exam (Chonoles) *Includes practice questions by the exam designer. |
| | 2. UML 2.0 in a Nutshell (Pitman) |
| | 3. UML 2 for Dummies (Schardt) |
| **Additional Reading** | **The Value of Modeling (IBM Software Group)** |
| | **Why Model? (Epstein)** |
| | **Business Modeling: A Practical Guide to Realizing Business Value-Excerpt from Chapter 7: Model Value Analysis (Zahavi)** |
| | **Why Domain Modeling (Wirfs-Brock)** |
| | **Model Organization with Packages and the Package Diagram (Baker)** |
| | **Concurrency in UML (Stachecki)** |
| **Useful Knowledge** | Modeling using UML, BPMN, SysML, or Realtime software development principles. |
| **Exam Training Required** | None |
| **Exam Training Options (not required)** | **inprogress** (Poland: **2 and 3-day courses**) |
| | **NobleProg** (**Canada**, **China**, **Germany**, **India**, **North America**, **Poland**, **UAE** and **UK**) |
| | **oose** (Germany: **1-day course** and **5-day course**) |
| | **RedPill** (Poland: **2-day course**) |
| **Exam Voucher Program** | Visit the **Pearson VUE Voucher Store** for a 10% discount/10 vouchers or contact **certificationinfo@omg.org** or call **+1-781-444-0404 Ext. 144** for a 15% discount/25 vouchers, a 20% discount/50 vouchers and a 25% discount/100+ vouchers. Vouchers can be transferred. Vouchers expire one year after purchase. **Contact Pearson VUE** to honor a previously purchased voucher price. |
| **Exam Registration** | **Pearson VUE**: create an account, locate a test center, view available tests, (re)schedule a test (online or at a test center), cancel your exam (contact Pearson VUE >=24 hours prior to exam for a full refund or you forfeit the full exam price), view exam scores and **Contact Pearson VUE**. |
| **Testing Accommodations** | If you have a hearing, learning, physical or visual disability you may contact us at **certificationinfo@omg.org** to provide instructions on testing accommodations. |
| **Online Exam Check-In & Requirements** | Visit **Pearson VUE Online Proctoring** for detailed info. Log in at least 30 minutes early (online verification may take 15-20 minutes). Late arrivals will not be allowed to take the exam. |
| **Test Center Check-In & Requirements** | Arrive at least 30 minutes early. Late arrivals will not be allowed to take the exam. Two forms of ID (at least one with photo and both with signature): alien registration card, bank card, credit card, employee badge, government issued, green card, military, passport, school and state ID. Do not bring any items (personal or otherwise) other than the two forms of ID to a test center. **Pearson VUE Test Center Coronavirus Guidelines** |
| **Exam Languages** | Offered in English. You cannot use a translating app during the exam. |

| | |
|---|---|
| **Review Your Answers** | Before completing an exam you will be presented with a review screen to review your answers to all questions. |
| **Exam Score Reports** | Pass or fail, you will be provided with a score report on computer screen immediately following the exam whether on-site at test center or online. A hardcopy will be provided to you before leaving test center with your score in each major section. If you fail, you can review those sections where you scored poorly to assist you when you decide to retake the exam. You can also review your exam scores via your **Pearson VUE account**. |
| **Certification Kit** | Those who pass the exam will receive a certification kit within 4-6 weeks of taking the exam. The kit will include a certification letter, certificate, digital certification logo download instructions, **guidelines** and how to opt-into the **OMG Certified Professionals Directory**. Certifications are associated with individuals and not companies. |
| **Lost Certificate** | Contact **certificationinfo@omg.org** with your full name, mailing address (if applicable) and candidate ID number. US$15 cost to mail each certificate or to email a .pdf version. |
| **Updating Contact Information** | You must first update your contact information via your **Pearson VUE account** and then contact **certificationinfo@omg.org** to update the **OMG Certified Professionals Directory**. |
| **Certification Expiration** | Your certification expires 5 years from the date you passed the exam. The same or a higher-level certification must be taken prior to the expiration date to extend a certification by 5 years. |
| **Retaking the Exam** | You can retake the exam 30 days after you last took the exam. However, an exam cannot be retaken more than 3 times within a 12-month period. The cost of a retaken exam is US$175 (or local equivalent) in English-speaking countries (exception: city of Quebec) and US$185 (or local equivalent) in all others. Contact **certificationinfo@omg.org** to request a discounted exam retake voucher. |
| **Original OCUP Certification Valid?** | Yes, but only the new OCUP2 certification will demonstrate the modeling knowledge and skills required in today's complex IT environment. |
| **Still Have Questions?** | **certificationinfo@omg.org** |

**General Areas Tested in OMG-OCUP2-FOUND100 Exam**

| | |
|---|---|
| **Class Diagram** | 25% |
| **Activity Diagram** | 20% |
| **Sequence Diagram** | 15% |
| **Why We Model** | 15% |
| **State Machine Diagram** | 10% |
| **Object Diagram** | 5% |
| **Package Diagram** | 5% |
| **Use Case Diagram** | 5% |
| **Total** | **100%** |

**Comprehensive Areas Tested in OMG-OCUP2-FOUND100 Exam**

| STRUCT | CLASSIFICATION | COMMON STRUCTURE | PACKAGES | SIMPLE CLASSIFIERS | STRUCTURED CLASSIFIERS | VALUES |
|---|---|---|---|---|---|---|
| | | | | | | |

| | | | | | |
|---|---|---|---|---|---|
| AggregationKind (Composition, Aggregation) Behavioral Feature Feature Generalization InstanceSpecification Operation Parameter Property Slot Structural Feature | Comment Constraint Dependency MultiplicityElement Namespace PackageImport Type VisibilityKind (public, private, protected) | Package (excludes Package Merge) | DataType Enumeration EnumerationLiteral Interface InterfaceRealization PrimitiveType Reception Signal | Association Class | LiteralBoolean LiteralInteger LiteralNull LiteralReal LiteralString LiteralUnlimitedNatural Opaque Expression |
| **ACTIONS** | **ACTIVITIES** | **COMMON BEHAVIOR** | **INTERACTIONS** | **STATE MACHINES** | **USE CASES** |
| AcceptEventAction Action CallBehaviorAction CallOperationAction InputPin, OutputPin OpaqueAction Pin SendSignalAction | Activity ActivityFinalNode ActivityParameterNode ControlFlow DecisionNode FlowFinalNode ForkNode InitialNode JoinNode MergeNode ObjectFlow ObjectNode | CallEvent OpaqueBehavior SignalEvent Trigger | DestructionOccurrenceSpecification ExecutionOccurrenceSpecification ExecutionSpecification Interaction Lifeline Message MessageEnd MessageOccurrenceSpecification MessageSort OccurrenceSpecification | FinalState Pseudostate (choice, junction, initial) State StateMachine Transition | Actor Extend Include UseCase |

*(Left vertical label: BEHAVIORAL DIAGRAMS)*

The following serves mainly as a basis for description of the **OMG-OCUP2-INT200** and **OMG-OCUP2-ADV300** exam coverage and is **not** required for the **OMG-OCUP2-FOUND100** exam. It is **not** required, but if you wish, please refer to the Unified Modeling Language (UML) v.2.5.1 specification for a more in-depth look at the corresponding chapters and sections cited below.

## CHAPTER 6: GENERAL TOPICS

**Abstract Syntax**

- Every first-level subsection of the UML specification starts with a UML diagram labeled *Abstract Syntax*. The OCUP 2 exams do not ask about these diagrams explicitly, but they are good examples of the language you're studying(!) and represent the relationships linking the elements to be presented in the sections that follow in a particularly clear and concise way. Learn to read them - this will provide an advantage to your study.

## CHAPTER 7: COMMON STRUCTURE

- 7.2 Root concepts - *All*
- 7.3 Templates - *Templates are Excluded* from Foundation and Intermediate levels; Templates and the many elements that support them will be covered at Advanced level *only*. This exclusion encompasses elements and attributes defined for Templates here and later on (String Expressions and Name Expressions, e.g.; most have "Template" somewhere in their names). There are many of these scattered throughout the specification but we will not point out on the Foundation and Intermediate Coverage

Maps, for each one, that it is excluded. This exclusion applies even within subsections denoted "All" in the coverage list.

- 7.4 Namespaces - *All except* as noted below:
    - *Except*: ownedRule constraints, nested nameSpaces, circle-plus notation, ElementImport
    - *Except*: StringExpression (used in Templates) and nameExpression
- 7.5 Types and Multiplicity - *All except* Cardinality, isOrdered, isUnique, multiplicity string
- 7.6 Constraints - *All except* owner
- 7.7 Dependencies - *All except* Usage, Abstraction, Realization

## CHAPTER 8: VALUES

- 8.2 Literals - *All*
- 8.3 Expressions
    - Opaque Expressions - *All Except* opaque expressions defined by a UML behavior

## CHAPTER 9: CLASSIFICATION

- 9.2 Classifiers
    - 9.2.3 Semantics
        - Classifiers: *All Except* the mentioned relations to Collaboration and UseCase
        - Generalization: *All Except* the detail in this section about Substitutability
- 9.4 Features
    - 9.4.3 Semantics
        - Features: *All*
        - Structural Features: *All Except* execution scope, isReadOnly
        - Behavioral Features: *All Except* concurrency
        - Parameters: *All Except* the effect property, redefines (*ParameterSet* is not covered in OCUP 2)
- 9.5 Properties
    - 9.5.3 Semantics
        - *Includes* Property as an attribute of a Classifier and as the parts of Structured Classifiers; context for the Property; basic definition of derived Property (isDerived=true); aggregation *Except* details about composite aggregation
        - The semantics of the defaultValue, isStatic, and isDerived properties will be covered in more detail at Intermediate level, but their notation and basic aspects (that is, the syntax) are covered at Foundation level
- 9.6 Operations
    - 9.6.3 Semantics
        - Operations: *Includes* basic definition including return Parameter
- 9.8 Instances
    - 9.8.3 Semantics
        - All except InstanceSpecification partially representing the instance it corresponds to, classification of the instance by zero or more than one Classifier, type restrictions on a defining ValueSpecification, and snapshots

## CHAPTER 10: SIMPLE CLASSIFIERS

- 10.2 DataTypes - *All*
- 10.3 Signals - Signals and Receptions will be covered in this exam as used in Sequence diagrams; see Chapter 17. The semantics covered in this chapter will be covered at Intermediate level.
- 10.4 Interfaces

- o  10.4.3 Semantics - *All except* ownership of a ProtocolStateMachine.

## CHAPTER 11: STRUCTURED CLASSIFIERS

- 11.4 Classes
  - o  11.4.1 Summary
    - ▪ *Include* purpose of a class. NOTE: The summary also points out that "Class is the concrete realization of EncapsulatedClassifier and BehavioredClassifier". Although this metamodel-based aspect will not be tested explicitly until Advanced level, it provides much insight to candidates who take the time to understand what it means, and what it indicates.
  - o  11.4.3 Semantics
    - ▪ Classes: *Includes* Basic aspects of Class. *Excludes* detailed aspects of, e.g., attributes, namespace, isActive, which will be covered at Intermediate level.
- 11.5 Associations
  - o  11.5.1 Summary: *Excludes* AssociationClass
  - o  11.5.3 Semantics
    - ▪ Associations: *Includes* basic aspects of Associations, including composite aggregation, navigability. Excludes Associations with more than two memberEnds, Association defining a collection, subsetting, specialization, navigableOwnedEnd, qualifier, derived Association

## CHAPTER 12: PACKAGES

- 12.1 Summary: *All Except* Profiles, which will be covered at Advanced level
- 12.2 Packages
  - o  12.2.3 Semantics
    - ▪ Package: *Includes* basic definition. *Excludes* merging, specifying the URI.
    - ▪ Package Merge: NOTE *that Package Merge is* Not *covered in OCUP 2 at any level.*
    - ▪ NOTE: For more on packages, download the White Paper *Model Organization with Packages and the Package Diagram* in the references section of the Primary Coverage Map page.

## CHAPTER 13: COMMON BEHAVIOR

- 13.2 Behaviors:
  - o  13.2.3 Semantics:
    - ▪ Behaviors: *All Except* behavior as a class, reentrant.
    - ▪ Behavior Parameters: *All Except* defaultValue. Includes streaming at awareness level only. *ParameterSets* are not covered in OCUP 2.
    - ▪ Behavioral Features and Methods: Includes Operations and Receptions; excludes method, context, resolution process
- 13.3 Events
  - o  At Foundation level, Events are treated in the context of specific diagrams (sequence, activity, state machine, primarily). Detailed aspects of Events (and of Behaviors in general), described in Section 13.3 and its subsections, will be covered at the Intermediate and Advanced levels.

## CHAPTER 14: STATEMACHINES

- StateMachine coverage at Foundation level:

- o StateMachine coverage at Foundation level Includes only the single-region Behavior State Machine. All aspects of this StateMachine are included *Except* the following:
  - specification of a method of a behavioredClassifier (that is, an Operation or Reception corresponding to a BeharioralFeature); regions; encapsulated composite States; submachine States; history (deep or shallow); deferred events and the event pool; the pseudostates join, fork, entrypoint, exitpoint, and terminate; transition kind=local; high-level (group) transitions; conflicting transitions; firing priorities; transition selection and execution sequence; StateMachine redefinition; and ProtocolStateMachines.
  - o Also *Excluded* is the alternative graphical representation illustrated in Figure 14.32.

## CHAPTER 15: ACTIVITIES

- 15.2 Activities
  - o 15.2.1 Summary: *All*
  - o 15.2.3 Semantics:
    - Activities: *Includes* the Token model - object tokens and control tokens - but *Excludes* object tokens over ControlFlow edges and isControlType, which will be covered at Advanced level. Also *Excludes* the null token, token movement details resulting from offer and acceptance, named edges, Activities as classes.
    - Activity Nodes: *All Except* concurrent execution, and one token offered to multiple targets
    - Activity Edges: *All Except* object tokens passing over activity edges, contention, weight, token ordering.
    - Object Flows: *Include* the basic definition of Object Flow.
    - Activity Execution: *Include* precondition and postcondition constraints.
- 15.3 Control Nodes
  - o 15.3.3 Semantics
    - Initial Node: *All Except* additional concurrent flows and CentralBufferNodes
    - Final Nodes: *All Except* isSingleExecution
    - Fork Nodes: *All Except* handling of unaccepted token offers
    - Join Nodes: *All Except* joinSpec and isCombinedDuplicate
    - Merge Nodes: *All*
    - Decision Nodes: The basic behavior of Decision Nodes is covered at this Foundation level, but the distinction between the decisionInputFlow and the primary incoming edge is *Not* covered until Intermediate and Advanced.
- 15.4 Object Nodes
  - o 15.4.1 Summary: *All Except* CentralBufferNodes and DataStoreNodes
  - o 15.4.3 Semantics:
    - Object Nodes: *Includes* Basic token input and output.
    - ActivityParameterNodes: *All Except* ordering
- 15.5 Executable Nodes
  - o 15.5.3 Semantics
    - Executable Nodes: *All Except* isControlType, multiple concurrent executions

## CHAPTER 16: ACTIONS

- 16.1 Summary: *Includes* the basic definition of Action as contained in Activities.
- 16.2 Actions:
  - o 16.2.3 Semantics
    - Actions: *Includes* basic definition of Action including input and output on pins, and localPrecondition and localPostcondition. *Excludes* context BehavioredClassifier,

StructuredActivityNodes, streaming, multiple instances, non-locally-reentrant and non-reentrant behavior, and effects of violations of localPrecondition and localPostcondition.
- Opaque Actions: *All Except* interpretation of body strings
- Pins: *Includes* basic definition of input and output pins, ordering (basic aspects only), multiplicity. Excludes multiplicity requirements on output pins for termination, ValuePins, ActionInputPins.
- Pins: *All Except* attributes ordering and isOrdered (NOTE: At Foundation level, values on a pin may be ordered but the exam will not link this causally with a value of either of these attributes), StructuredActivityNodes and pins during execution, multiplicity requirements on output pins for termination, ValuePins, ActionInputPins
- 16.3 Invocation Actions
  - 16.3.3 Semantics
    - Call Actions: *Includes* CallAction, basic definition. CallBehaviorAction, CallOperationAction
    - Send Actions: *Includes* basic definition, SendSignalAction.
- 16.10 Accept Event Actions
  - 16.10.1 Summary: *All*
  - 16.10.3 Semantics
    - Accept Event Action: *All*
    - Accept Call Actions: Definition, Pins, triggering and basic returning values. Excludes triggering by an asynchronous call, method behavior caveat

## CHAPTER 17: INTERACTIONS

- 17.1 Summary
  - 17.1.1 Overview: General uses as discussed, trace, allowed and disallowed traces (but disallowed or invalid traces will be tested at Advanced level only), discussion relating to sequence diagrams.
  - 17.1.2 Basic Trace Model: *All Except* Interaction equivalence
  - 17.1.3 Partial ordering constraints on valid and invalid traces: *All Except* coregion or parallel operator effect
  - 17.1.4 Interaction Diagram Variants: The Sequence Diagram is tested at Foundation level. The Communication Diagram will be tested at (this) Intermediate level. The Interaction Overview Diagram will be tested at Advanced level. Neither the Timing Diagram nor Interaction Tables will be tested in OCUP 2.
- 17.2 Interactions
  - 17.2.3 Semantics
    - Interactions: *All Except* generalizing, redefining, and specializing an Interaction, and use of a formal Gate
    - Occurrence Specifications: *All*
    - Execution Specifications: *All*
- 17.3 Lifelines
  - 17.3.3 Semantics
    - Lifelines: *Includes* definition and use of Lifelines in modeling *Except* local ordering. Parallel combined fragment will be tested at Intermediate level, and coregion will be tested at Advanced.
- 17.4 Messages
  - 17.4.3 Semantics
    - Messages: *Includes* signature as Operation or Signal, messagesort.
    - Message Ends: *All*
    - Message Occurrence Specifications: *All*
    - Destruction Occurrence Specifications: *All*

- - - Gates: MessageOccurrenceSpecifications ordering rules
  - 17.5 Occurrences
    - 17.5.3 Semantics
      - Action Execution Specifications: *All*
      - Behavior Execution Specifications: *All*
      - Execution Occurrence Specifications: *All*
  - 17.6 Fragments
    - 17.6.3 Semantics
      - Weak Sequencing: *All*
  - 17.8 Sequence Diagrams
    - Introduction: *All*
      - 17.8.1: Sequence Diagram Notation
        - Graphic Nodes: Frame, Lifeline, ExecutionSpecification, DestructionOccurrenceSpecification
        - Graphic Paths: Message

## CHAPTER 18: USECASES

- 18.1 UseCases
  - 18.1.1 Summary: *All Except* technical definition of instance
  - 18.1.3 Semantics
    - Use Cases and Actors: *All Except* UseCase as BehavioredClassifier; description through a Collaboration; being owned by a Classifier.
    - Extends: *All Except* ownership of the extend relationship; extensionLocation
    - Includes: *All Except* Include being a kind of NamedElement

## CHAPTER 21: PRIMITIVE TYPES

- 21.1 Summary: *All*
- 21.2 Semantics: *All*